Foundations and Trends® in Information Retrieval
Vol. XX, No. XX (2017) 1–163
© 2017 now Publishers Inc.
DOI: 10.1561/XXXXXXXXXX

Web Forum Retrieval and Text Analytics: a Survey

Doris Hoogeveen
University of Melbourne
dhoogeveen@student.unimelb.edu.au

Li Wang
Evernote, California
li@liwang.info

Timothy Baldwin
University of Melbourne
tb@ldwin.net

Karin M. Verspoor
University of Melbourne
karin.verspoor@unimelb.edu.au

Hoogeveen, Doris, Li Wang, Timothy Baldwin and Karin M. Verspoor (2018) Web Forum Retrieval and Text Analytics: a Survey, Foundations and Trends in Information Retrieval, Vol. 12, No. 1, pp 1-163.

Contents

Abstract

This survey presents an overview of information retrieval, natural language processing and machine learning research that makes use of forum data, including both discussion forums and community question-answering (cQA) archives. The focus is on automated analysis, with the goal of gaining a better understanding of the data and its users.

We discuss the different strategies used for both retrieval tasks (post retrieval, question retrieval, and answer retrieval) and classification tasks (post type classification, question classification, post quality assessment, subjectivity, and viewpoint classification) at the post level, as well as at the thread level (thread retrieval, solvedness and task orientation, discourse structure recovery and dialogue act tagging, QA-pair extraction, and thread summarisation). We also review work on forum users, including user satisfaction, expert finding, question recommendation and routing, and community analysis.

The survey includes a brief history of forums, an overview of the different kinds of forums, a summary of publicly available datasets for forum research, and a short discussion on the evaluation of retrieval tasks using forum data.

The aim is to give a broad overview of the different kinds of forum research, a summary of the methods that have been applied, some insights into successful strategies, and potential areas for future research.

now Publishers Inc.. *Web Forum Retrieval and Text Analytics: a Survey.* Foundations and Trends® in Information Retrieval, vol. XX, no. XX, pp. 1–163, 2017.

DOI: 10.1561/XXXXXXXXXX.

1

Introduction

In this survey we will give an overview of a broad range of forum-related research. Forum research can be divided into two streams: discussion forums and community question-answering (cQA) archives. Both of these are websites that promote interaction and information sharing by the community, but they differ in their purpose, and because of that they often differ in their specific setup as well.

Forum data has been used for a large range of tasks and subtasks in information retrieval and natural language processing. Most of the tasks have to do with improving access to the rich information in the data, like post, question, or answer retrieval, thread summarisation, and expert finding. Subtasks cover specific aspects of the data and can be used to improve the results of the main tasks. Examples include dialogue act tagging, question and post type classification, post quality assessment, subjectivity and viewpoint classification, solvedness detection, thread type identification, topic detection, and user analysis. Forum research can also be used to improve the organization of the data, for instance by identifying duplicate questions, or categorizing posts.

In the remaining sections, we will present an overview of the different types of forums (§1.1), briefly discuss their history (§1.2), outline the scope of the survey (§1.3), present a glossary (§1.4), and present an overview of existing datasets used for forum research (§1.5).

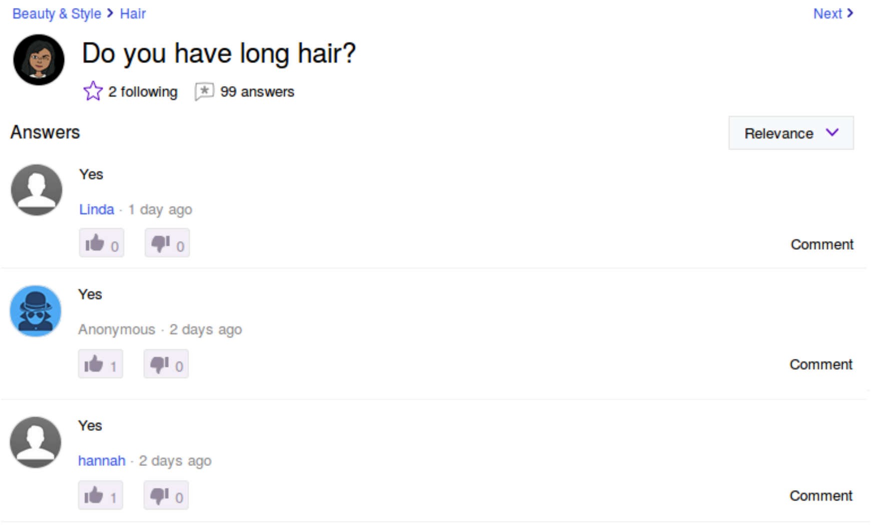
1.1 Types of forums

In this section we will look at the differences between discussion forums and community question-answering archives. Both of these promote community interaction. Community question-answering archives are meant to help people to solve their problems and answer their questions. As soon as someone posts a good answer to a new question, the interaction is considered to be finished. Discussion forums on the other hand, are meant as a platform for people to discuss things.

This difference is not always strictly observed however. Some cQA archives contain questions like `"Any1 from NY?"`, which do not express an information need, but rather a social need. Another example is requests for recommendations. Such questions do not have one correct answer and are therefore again more suited to discussion forums. Conversely, many factual questions and requests for help are posted on discussion forums, which might be more suitable for cQA archives.

Not much work has been published on the typology of forums. Choi et al. [2012] proposed a typology of online Q&A models consisting of four distinct types: community-based (e.g. Yahoo! Answers), collaborative (e.g. WikiAnswers), expert-based (e.g. the Internet Public Library (IPL) 'Ask a Librarian'-service), and social (e.g. Twitter, which we do not consider to be a forum). Shah et al. [2014] placed the four cQA forum types in a hierarchical structure of Q&A services, which also includes face-to-face Q&A, and automatic Q&A services. Discussion forums are not present in either of these taxonomies. Several dimensions along which we can classify internet communication tools (including forums) are presented in Long and Baecker [1997]. While slightly outdated, it includes aspects like *conversational style* and *audience membership*, which are still valid today. Similar relevant dimensions or aspects can be found in Davies et al. [2005] (e.g. degree of interaction, motivation/orientation, size, maintenance, etc.).

In this survey we argue that forums exist on a spectrum with dis-



Beauty & Style > Hair

Do you have long hair?

2 following 99 answers

Answers

Yes

Linda · 1 day ago

Yes

Anonymous · 2 days ago

Yes

hannah · 2 days ago

Figure 1.1: An example of a question on a cQA archive that may be intended to start a conversation. Source: Yahoo! Answers, `https://au.answers.yahoo.com/question/index?qid=20160921123000AAlwLIx`, accessed on 24th of September 2016.

cussion threads on the one hand, where users have a high degree of freedom in what they post, and strict question-answering threads on the other, with heavy moderation to ensure only good answers are posted and threads are closed as soon as the question has been answered in a satisfactory way. In some cases the distinction is blurred. Linux Questions (`http://www.linuxquestions.org/`) for instance, looks like a forum, and has subforums dedicated to discussing Linux related topics, but also focuses on answering questions. Yahoo! Answers (`https://answers.yahoo.com/`), a cQA archive, contains questions that look like they are intended to spark a conversation. An example can be found in Figure 1.1. This also illustrates the lack of moderation on Yahoo! Answers.

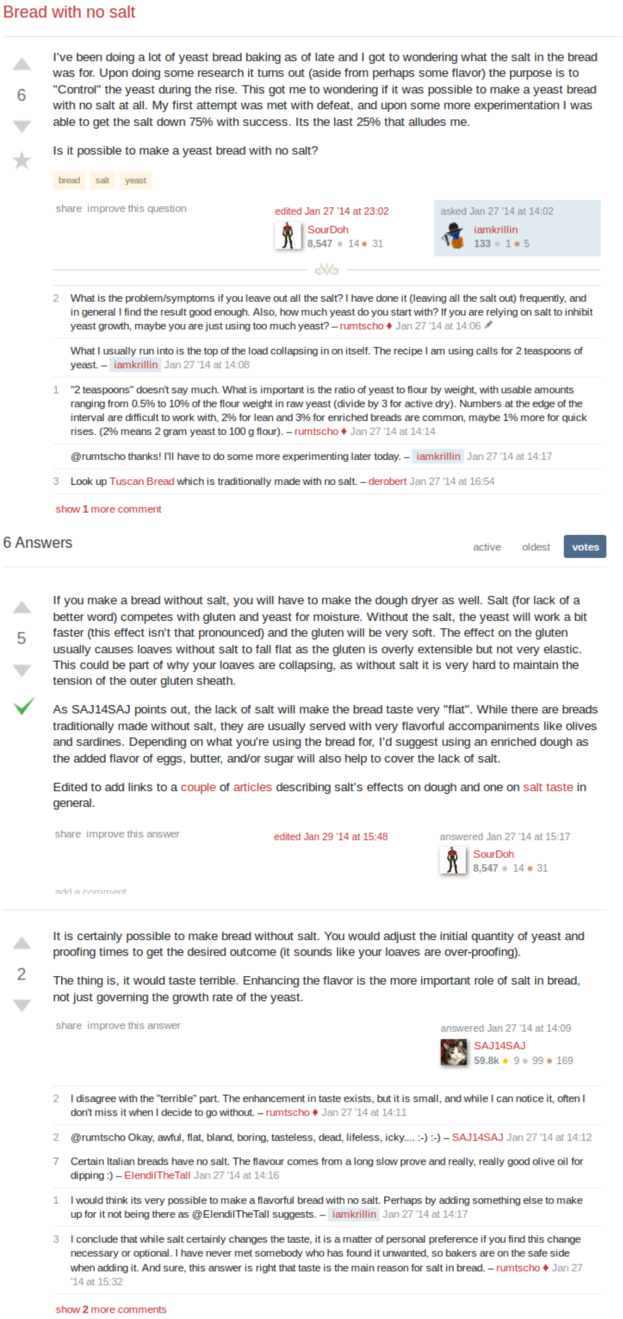
On the far end of the cQA side of the spectrum there are cQA sites with a high degree of moderation supplied by the community itself. On such websites there is often a reward system in place for users that ask good questions and provide good answers. StackExchange is a good example of this. Figure 1.2 shows an example of a thread from the StackExchange Cooking site.

As can be seen in the example, a distinction is made between answers and comments. Comments are used to ask for clarification, correct people, offer small suggestions, or make general remarks or even jokes. Answers are reserved for genuine answers. The number of reputation points and other rewards the users have obtained is shown next to their name. In this way, active contributors and experts can be distinguished from new users. This can be one way for users to consider which answer is the best one. Users can also look at the number of up votes and down votes an answer has received. These votes are cast by the community to indicate the quality of answers (and questions).

Another characteristic of most cQA archives, and something that discussion forums do not offer, is that question askers are encouraged to choose one of the answers as the best answer. That way other users know that the information need has been satisfied and they can focus their efforts on other questions. Repeated questions can be linked to archived ones, and an active effort is made by the community to keep the answers focused and not to stray away from the question. When it does happen, the question is usually closed. This is very different from discussion forums, where some threads can 'live' for very long and no one is bothered by it. A classic example of this is the famous `"i am lonely will anyone speak to me"` thread posted in the Moviecodec.com branch discussion forum, The Lounge Forums, in 2004.[1] It is still active today: more than twenty years since it was started.

Forums differ in how much access they offer to the outside world, but most of them make their content visible for everyone, while requiring people to sign up if they want to contribute. Some forums offer the option to sign up as an anonymous user. This makes the threshold to contribute lower. In some forums that is seen as a good thing, because

[1] `https://www.loungeforums.com/on-topic/i-am-lonely-will-anyone-speak-to-me-2420/`. It is more than 2000 pages long. Several magazines and newspapers have featured this thread. See for more information `https://en.wikipedia.org/wiki/I_am_lonely_will_anyone_speak_to_me`.



Bread with no salt

6

I've been doing a lot of yeast bread baking as of late and I got to wondering what the salt in the bread was for. Upon doing some research it turns out (aside from perhaps some flavor) the purpose is to "Control" the yeast during the rise. This got me to wondering if it was possible to make a yeast bread with no salt at all. My first attempt was met with defeat, and upon some more experimentation I was able to get the salt down 75% with success. Its the last 25% that alludes me.

Is it possible to make a yeast bread with no salt?

edited Jan 27 '14 at 23:02
SourDoh
8,547 ● 14 ● 31

asked Jan 27 '14 at 14:02
iamkrillin
133 ● 1 ● 5

2 What is the problem/symptoms if you leave out all the salt? I have done it (leaving all the salt out) frequently, and in general I find the result good enough. Also, how much yeast do you start with? If you are relying on salt to inhibit yeast growth, maybe you are just using too much yeast? – rumtscho ♦ Jan 27 '14 at 14:06

What I usually run into is the top of the load collapsing in on itself. The recipe I am using calls for 2 teaspoons of yeast. – iamkrillin Jan 27 '14 at 14:08

1 "2 teaspoons" doesn't say much. What is important is the ratio of yeast to flour by weight, with usable amounts ranging from 0.5% to 10% of the flour weight in raw yeast (divide by 3 for active dry). Numbers at the edge of the interval are difficult to work with, 2% for lean and 3% for enriched breads are common, maybe 1% more for quick rises. (2% means 2 gram yeast to 100 g flour). – rumtscho ♦ Jan 27 '14 at 14:14

@rumtscho thanks! I'll have to do some more experimenting later today. – iamkrillin Jan 27 '14 at 14:17

3 Look up Tuscan Bread which is traditionally made with no salt. – derobert Jan 27 '14 at 16:54

6 Answers

5

If you make a bread without salt, you will have to make the dough dryer as well. Salt (for lack of a better word) competes with gluten and yeast for moisture. Without the salt, the yeast will work a bit faster (this effect isn't that pronounced) and the gluten will be very soft. The effect on the gluten usually causes loaves without salt to fall flat as the gluten is overly extensible but not very elastic. This could be part of why your loaves are collapsing, as without salt it is very hard to maintain the tension of the outer gluten sheath.

As SAJ14SAJ points out, the lack of salt will make the bread taste very "flat". While there are breads traditionally made without salt, they are usually served with very flavorful accompaniments like olives and sardines. Depending on what you're using the bread for, I'd suggest using an enriched dough as the added flavor of eggs, butter, and/or sugar will also help to cover the lack of salt.

Edited to add links to a couple of articles describing salt's effects on dough and one on salt taste in general.

edited Jan 29 '14 at 15:48

answered Jan 27 '14 at 15:17
SourDoh
8,547 ● 14 ● 31

2

It is certainly possible to make bread without salt. You would adjust the initial quantity of yeast and proofing times to get the desired outcome (it sounds like your loaves are over-proofing).

The thing is, it would taste terrible. Enhancing the flavor is the more important role of salt in bread, not just governing the growth rate of the yeast.

answered Jan 27 '14 at 14:09
SAJ14SAJ
59.8k ● 9 ● 99 ● 169

2 I disagree with the "terrible" part. The enhancement in taste exists, but it is small, and while I can notice it, often I don't miss it when I decide to go without. – rumtscho ♦ Jan 27 '14 at 14:11

2 @rumtscho Okay, awful, flat, bland, boring, tasteless, dead, lifeless, icky.... :-) :-) – SAJ14SAJ Jan 27 '14 at 14:12

7 Certain Italian breads have no salt. The flavour comes from a long slow prove and really, really good olive oil for dipping :) – ElendilTheTall Jan 27 '14 at 14:16

1 I would think its very possible to make a flavorful bread with no salt. Perhaps by adding something else to make up for it not being there as @ElendilTheTall suggests. – iamkrillin Jan 27 '14 at 14:17

3 I conclude that while salt certainly changes the taste, it is a matter of personal preference if you find this change necessary or optional. I have never met somebody who has found it unwanted, so bakers are on the safe side when adding it. And sure, this answer is right that taste is the main reason for salt in bread. – rumtscho ♦ Jan 27 '14 at 15:32

Figure 1.2: An example of a cQA thread. Source: StackExchange Cooking, `http://cooking.stackexchange.com/questions/41501/bread-with-no-salt`. Modified slightly by removing some answers, for presentational purposes. Accessed on 24th of September 2016.

it lowers the bar of entry, but in forums that want to create a steady community of people that contribute regularly, these kinds of one-off contributions are discouraged. Having a system where people need to sign up before they can participate has the added benefit of making it difficult for bots to post spam, and it allows for personalisation of the forum. Some forums even offer member pages with all kinds of meta data such as when they became a member, how active they are, reputation points, question and answer history, and all the subforums they participate in, or topics they have expertise in. StackExchange[2] is once again a good example of this.

While many discussion forums explicitly show the discourse structure of the thread, i.e., which post is a reply to which earlier post, (see Figure 1.3 for an example), this is not always the case (see Figure 1.4). Quoted posts, allowed by some forums and illustrated in Figure 1.5, can be used to retrieve at least part of the discourse structure. We discuss this in §4.2.

CQA archives only have a simple two-part discourse structure, between a question and each of its answers. The original order of the answers is often not preserved. Instead, they are usually ordered based on the number of votes they have received from the community, with the answer that was accepted as the correct one by the question asker at the top.

1.2 A short history of forums

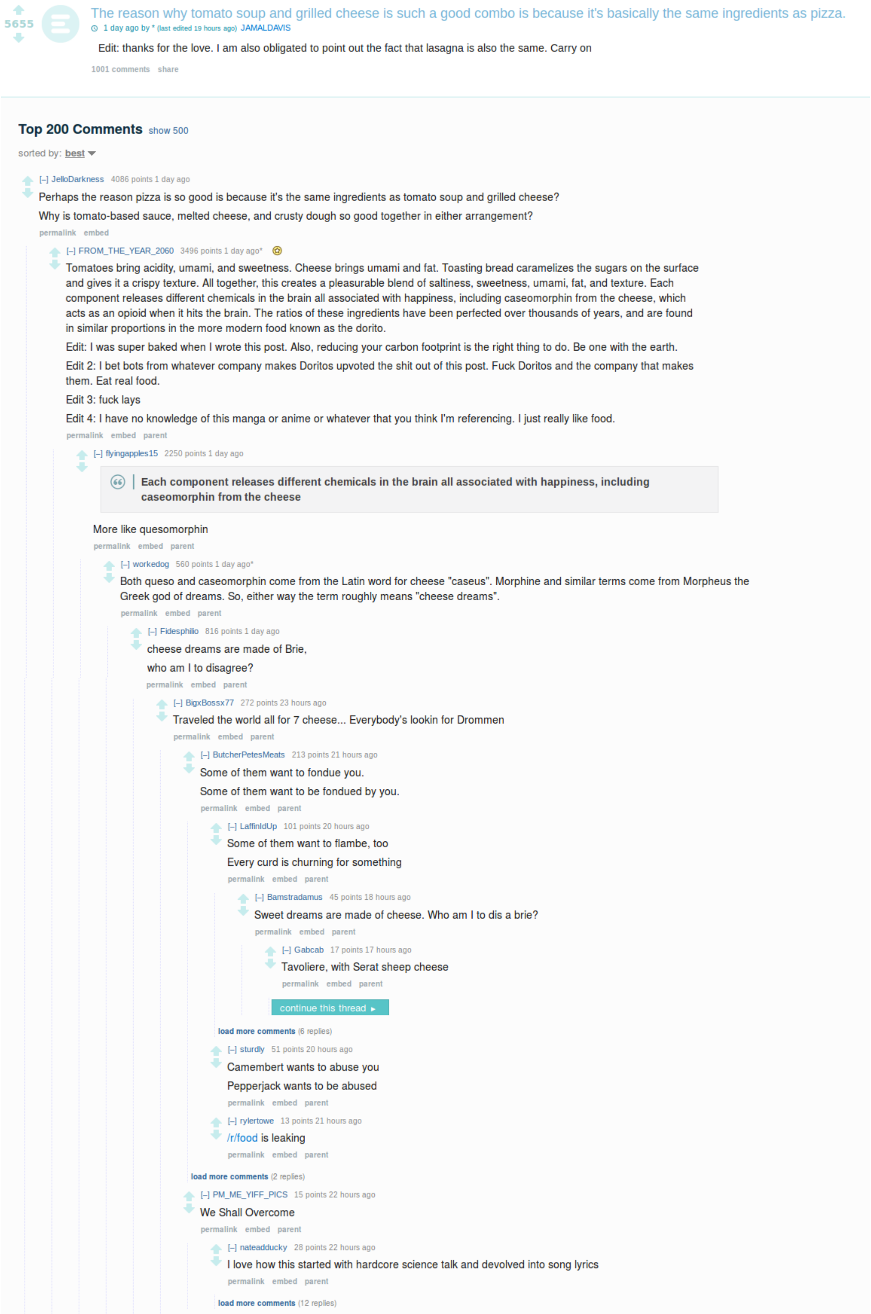
One of the earliest examples of a community question-answering service is The Straight Dope[3] founded in 1973. It started out as a column in several American newspapers, but these days it also has an online forum where people can ask questions and receive answers. The setup is closer to a discussion forum than a cQA archive however, with several subforums specifically created for discussion, such as the Elections subforum.

Another early example is the Internet Oracle[4] founded in 1989. It

[2] `http://stackexchange.com/`

[3] `http://www.straightdope.com/`

[4] `http://internetoracle.org/`



5655

The reason why tomato soup and grilled cheese is such a good combo is because it's basically the same ingredients as pizza.

1 day ago by * (last edited 19 hours ago) JAMALDAVIS

Edit: thanks for the love. I am also obligated to point out the fact that lasagna is also the same. Carry on

1001 comments share

Top 200 Comments show 500

sorted by: best

[–] JelloDarkness 4086 points 1 day ago

Perhaps the reason pizza is so good is because it's the same ingredients as tomato soup and grilled cheese?

Why is tomato-based sauce, melted cheese, and crusty dough so good together in either arrangement?

permalink embed

[–] FROM_THE_YEAR_2060 3496 points 1 day ago*

Tomatoes bring acidity, umami, and sweetness. Cheese brings umami and fat. Toasting bread caramelizes the sugars on the surface and gives it a crispy texture. All together, this creates a pleasurable blend of saltiness, sweetness, umami, fat, and texture. Each component releases different chemicals in the brain all associated with happiness, including caseomorphin from the cheese, which acts as an opioid when it hits the brain. The ratios of these ingredients have been perfected over thousands of years, and are found in similar proportions in the more modern food known as the dorito.

Edit: I was super baked when I wrote this post. Also, reducing your carbon footprint is the right thing to do. Be one with the earth.

Edit 2: I bet bots from whatever company makes Doritos upvoted the shit out of this post. Fuck Doritos and the company that makes them. Eat real food.

Edit 3: fuck lays

Edit 4: I have no knowledge of this manga or anime or whatever that you think I'm referencing. I just really like food.

permalink embed parent

[–] flyingapples15 2250 points 1 day ago

> **Each component releases different chemicals in the brain all associated with happiness, including caseomorphin from the cheese**

More like quesomorphin

permalink embed parent

[–] workedog 560 points 1 day ago*

Both queso and caseomorphin come from the Latin word for cheese "caseus". Morphine and similar terms come from Morpheus the Greek god of dreams. So, either way the term roughly means "cheese dreams".

permalink embed parent

[–] Fidesphilio 816 points 1 day ago

cheese dreams are made of Brie,

who am I to disagree?

permalink embed parent

[–] BigxBossx77 272 points 23 hours ago

Traveled the world all for 7 cheese... Everybody's lookin for Drommen

permalink embed parent

[–] ButcherPetesMeats 213 points 21 hours ago

Some of them want to fondue you.

Some of them want to be fondued by you.

permalink embed parent

[–] LaffinIdUp 101 points 20 hours ago

Some of them want to flambe, too

Every curd is churning for something

permalink embed parent

[–] Bamstradamus 45 points 18 hours ago

Sweet dreams are made of cheese. Who am I to dis a brie?

permalink embed parent

[–] Gabcab 17 points 17 hours ago

Tavoliere, with Serat sheep cheese

permalink embed parent

continue this thread ▸

load more comments (6 replies)

[–] sturdly 51 points 20 hours ago

Camembert wants to abuse you

Pepperjack wants to be abused

permalink embed parent

[–] rylertowe 13 points 21 hours ago

/r/food is leaking

permalink embed parent

load more comments (2 replies)

[–] PM_ME_YIFF_PICS 15 points 22 hours ago

We Shall Overcome

permalink embed parent

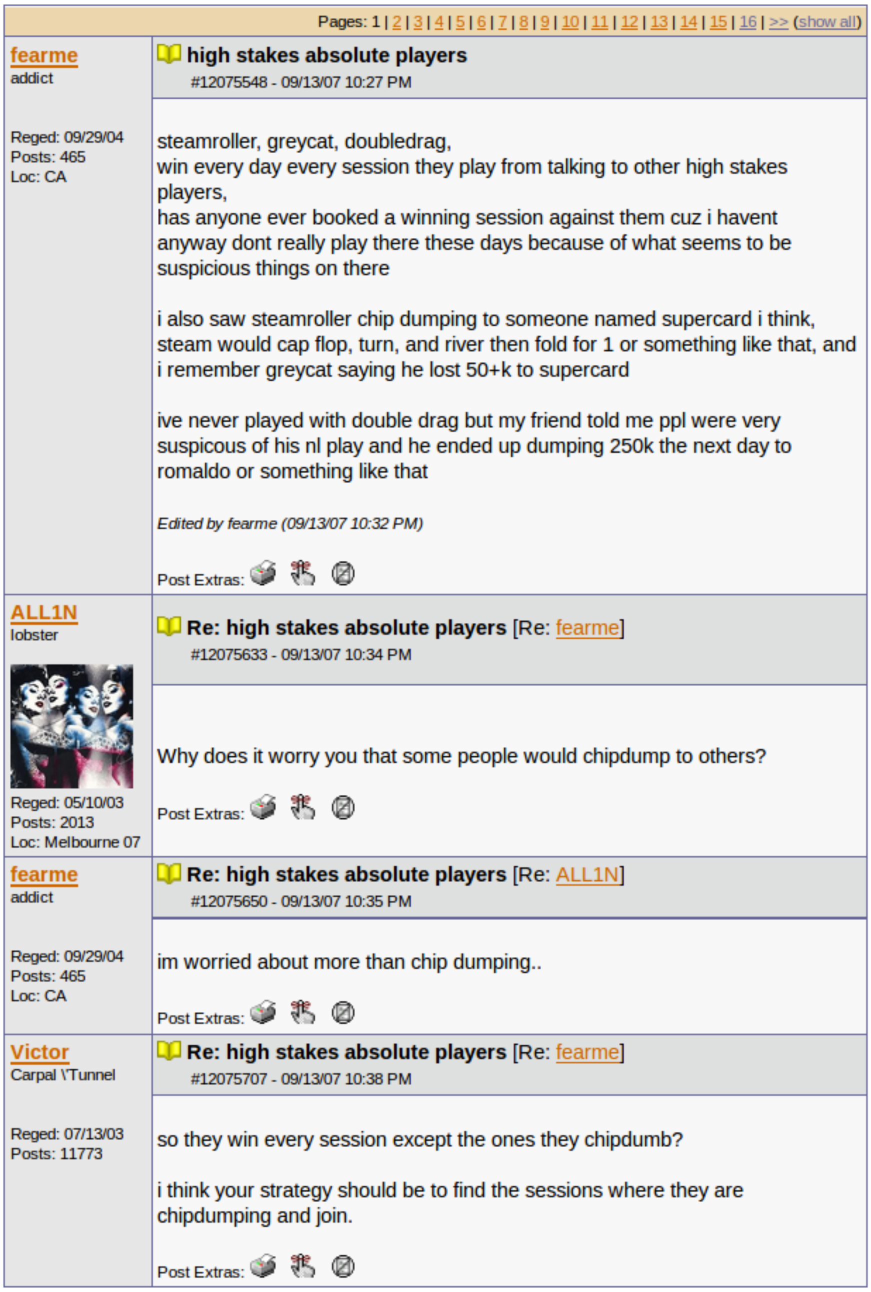
[–] nateadducky 28 points 22 hours ago

I love how this started with hardcore science talk and devolved into song lyrics

permalink embed parent

load more comments (12 replies)

Figure 1.3: An example of a discussion forum thread with explicit discourse structure. Source: Reddit, `https://www.reddit.com/r/Showerthoughts/comments/5403tk/the_reason_why_tomato_soup_and_grilled_cheese_is/`, accessed on 24th of September 2016.



Pages: 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | >> (show all)

fearme
addict

Reged: 09/29/04
Posts: 465
Loc: CA

high stakes absolute players
#12075548 - 09/13/07 10:27 PM

steamroller, greycat, doubledrag,
win every day every session they play from talking to other high stakes players,
has anyone ever booked a winning session against them cuz i havent
anyway dont really play there these days because of what seems to be suspicious things on there

i also saw steamroller chip dumping to someone named supercard i think, steam would cap flop, turn, and river then fold for 1 or something like that, and i remember greycat saying he lost 50+k to supercard

ive never played with double drag but my friend told me ppl were very suspicous of his nl play and he ended up dumping 250k the next day to romaldo or something like that

Edited by fearme (09/13/07 10:32 PM)

Post Extras:

ALL1N
lobster

Reged: 05/10/03
Posts: 2013
Loc: Melbourne 07

Re: high stakes absolute players [Re: fearme]
#12075633 - 09/13/07 10:34 PM

Why does it worry you that some people would chipdump to others?

Post Extras:

fearme
addict

Reged: 09/29/04
Posts: 465
Loc: CA

Re: high stakes absolute players [Re: ALL1N]
#12075650 - 09/13/07 10:35 PM

im worried about more than chip dumping..

Post Extras:

Victor
Carpal \'Tunnel

Reged: 07/13/03
Posts: 11773

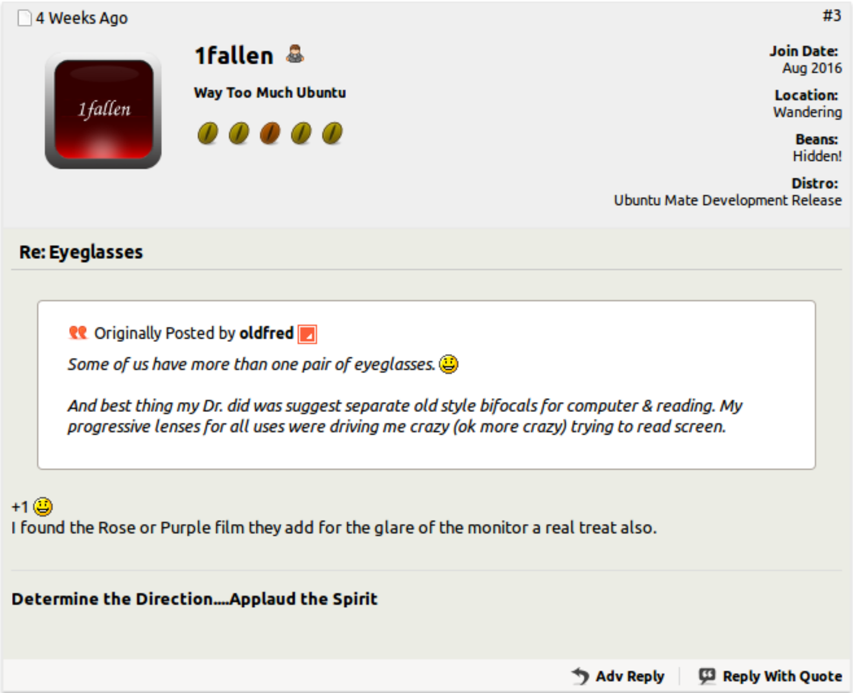
Re: high stakes absolute players [Re: fearme]
#12075707 - 09/13/07 10:38 PM

so they win every session except the ones they chipdumb?

i think your strategy should be to find the sessions where they are chipdumping and join.

Post Extras:

Figure 1.4: An example of a discussion forum thread without explicit discourse structure. Source: The Two Plus Two Forum, `http://archives1.twoplustwo.com/showflat.php?Cat=0&Number=12075548`, accessed on 21st of October 2016.



□ 4 Weeks Ago

#3

1fallen

Way Too Much Ubuntu

Join Date:
Aug 2016

Location:
Wandering

Beans:
Hidden!

Distro:
Ubuntu Mate Development Release

Re: Eyeglasses

> Originally Posted by **oldfred**
>
> *Some of us have more than one pair of eyeglasses.*
>
> *And best thing my Dr. did was suggest separate old style bifocals for computer & reading. My progressive lenses for all uses were driving me crazy (ok more crazy) trying to read screen.*

+1

I found the Rose or Purple film they add for the glare of the monitor a real treat also.

Determine the Direction.....Applaud the Spirit

Figure 1.5: An example of a post on a discussion forum, that quoted an earlier post, to make it clear what exactly is replied to. Source: Ubuntu Forums, `https://ubuntuforums.org/showthread.php?t=2337749`, accessed on 21st of October 2016.

specialises in humorous answers. Although it is a community question-answering service, questions and answers are submitted and distributed via e-mail.

Discussion forums also started to appear in the late 1980s. The Delphi Forums[5] was created in 1983 and is one of the earliest forums; it is still active today. Online discussion forums have their origins in bulletin boards and newsgroups such as Usenet, which has been around since 1980.

In the 1990s several cQA archives emerged. For instance:

- The Madsci Network:[6] It is heavily moderated and questions are all answered by scientists, rather than being open to anyone willing to contribute.
- Experts-Exchange[7]: This site is specifically for technology experts. It started out as purely community question-answering, but has expanded and now also offers help with code reviews, connecting freelancers to jobs, educating people, and live chat with an expert.
- 3form:[8] focuses on finding solutions to problems, rather than answers to questions. That is, questions are requests for information, either factual or not, while problems are questions for help in solving a particular issue.

Discussion forums also grew in popularity. In 1994 the W3C introduced WWW Interactive Talk (WIT),[9] a discussion forum that followed a set of design principles to display online discussions in such a way that it was easy to see which different topics were being discussed, and which points had been settled or not. Before WIT, many discussion forums suffered from the problem of people posting the same arguments over and over again, because there was no clear overview of a

[5] `http://www.delphiforums.com/`

[6] `http://www.madsci.org/` started in 1995 and still going.

[7] `https://www.experts-exchange.com/` started in 1996 and still going.

[8] `http://3form.org/` started in 1998 and still going.

[9] Official website: `https://www.w3.org/WIT/`, and more information can be found at `http://speed.eik.bme.hu/help/html/Special_Edition-Using_CGI/ch17.htm#WWWInteractiveTalk`.

full thread. Although this was a step forward, and many alternatives sprang from this, to a certain extent we are still struggling with similar issues today.

In the first decade of the 2000s a large number of new cQA archives appeared, many of which are still extremely popular today: Baidu Knows,[10] WikiAnswers/Answers.com,[11] Quora,[12] Naver Knowledge Search,[13] Yahoo! Answers,[14] and the StackExchange[15] website, especially StackOverflow.[16] The only notable exception is Google Answers[17] which was started in 2002 but discontinued in 2006. Many of these large cQA archives are in English, but not all of them: Naver is Korean, and Baidu Knows is Chinese.

One specific example of a space where forums have been used and found to be helpful is education. There are several online cQA archives dedicated to questions about topics taught in schools. An example of this is Brainly,[18] which has the slogan "For students. By students." The idea is that students help each other to learn. Other examples are Chegg,[19] and Piazza.[20] Lang-8[21] is a language learning platform that has many similarities to forums. Users write posts in a language they are learning. Native speakers of that language will then correct the post sentence by sentence and comment on it. The original poster can reply to the corrections, and other native speakers can join in the conversation too, to discuss linguistic constructs or explain semantic or syntactic points.

[10] `https://zhidao.baidu.com/` started in 2005.

[11] `http://www.answers.com/` started on 2006 and its predecessor FAQForm in 2002.

[12] `https://www.quora.com/` started in 2009.

[13] `http://kin.naver.com/index.nhn` started in 2002.

[14] `https://answers.yahoo.com/` started in 2005 and formally known as Yahoo! Q&A.

[15] `http://stackexchange.com/` started in 2008.

[16] `https://stackoverflow.com/`, the first cQA site of the StackExchange network.

[17] `http://answers.google.com/answers/`. It grew out of Google Questions and Answers which was started in 2001.

[18] `http://brainly.com/`

[19] `https://www.chegg.com/`

[20] `https://piazza.com/`

[21] `http://lang-8.com/`

Many learning management systems include a forum to enable students to start discussions online, or ask questions. This is considered to be a vital ingredient of MOOCs for instance, where the number of students is so large that it is not possible for them to individually get in touch with the professor or tutors, and forums offer an alternative to ask for help or discuss the subject matter. In such a setting, the forums are used both as a cQA platform and as a discussion forum. One MOOC platform, EdX,[22] has recognised this dual nature of MOOC forums and allows people to choose what kind of post they make: a discussion sparking post, or a question-answer post. Threads are then labeled accordingly, so that other people know what kind of content a thread contains. The idea is that this labeling enhances information access.[23]

1.3 Scope and outline

In this survey we will describe research into automated analysis of forum data. That includes data from both discussion forums (also called web user forums; see, for instance, [Wang et al., 2013b]) and community question-answering (cQA) archives. These two forum types share a number of characteristics (as discussed in §1.1), which are not shared with other (semi) threaded discourses, like chat discussions, email threads, product reviews, or frequently asked question (FAQ) pages. These are therefore outside the scope of this survey.

At the start of this section we mentioned several tasks and subtasks. Each of these will be discussed in the following sections, divided into post classification (§2), post retrieval (§3), thread level tasks (§4), and social forum analysis or user studies (§5).

Previously published survey articles include Shah et al. [2009], who present an overview of early research in the cQA field, Gazan [2011], Li [2014], and Srba and Bielikova [2016], who all present an overview of cQA related research. Srba and Bielikova [2016] is the most recent and most comprehensive survey, discussing 265 research papers published

[22] `https://www.edx.org/`

[23] `http://blog.edx.org/navigating-two-kinds-online-discussion`

before February 2015. They also show that the number of publications in this field has increased each year.

This survey covers 450 papers published until November 2016, and distinguishes itself from earlier survey papers by including discussion forums, instead of focusing on cQA archives only.

1.4 Glossary

The same or similar concepts sometimes appear in the literature under different names. We will try to use the same terminology for each concept throughout this survey. This section summarises the important terminology we will use.

Thread: we use the term "thread" to refer to forum discussion threads, or a question on a cQA forum together with all of its answers (and comments). In discussion forums this is the full thread, which may span multiple pages (see Page below).

Page: in discussion forums, threads can sometimes become very large. If this happens, instead of displaying the full thread, only a certain number of posts are displayed at a time. So threads are divided into smaller units for easier display. Such chunks are called "pages".

Post/message: the terms "post" and "message" are often used interchangeably in the research community to refer to each posting in a forum thread. In this survey we use "post" to denote forum thread post. The term "post" can also be used to refer to either the question post in a cQA archive, or an answer post. We use it as a general term when we want to refer to any text posted by a user, regardless of whether it is an initial post or question post, or an answer post. In situations where it matters we will distinguish clearly between the two, by calling them "initial post" (or "question post") and "answer post".

Initial post: this refers to the first post in a discussion forum thread, which starts a discussion. In the literature, it is sometimes also called the "root post/message" or "first post/message".

Question post: this refers to the first post in a cQA thread, in which a question is asked. All other posts in a cQA thread are answers to this post.

Answer post: this refers to any post in a cQA thread that is not the question post, but rather a response to a question post.

Word/term: in this survey, "word" and "term" are used interchangeably to indicate a word unit in a post.

Thread initiator: the user who starts a new discussion thread (in discussion forums), or who posts a question (in cQA archives). This is the person that writes the INITIAL POST or QUESTION POST. In a cQA context we will sometimes refer to this person as the "question asker".

Quoted text: in discussion forums a user may sometimes quote content from previous posts or email messages in his/her post. This quoted content is called "quoted text". In cQA archives, quoted material often comes from other threads or from technical documentation. An example from a discussion forum can be found in Figure 1.5.

Comment: in some cQA archives, users can write comments to posts, in addition to answers. These two kinds of posts (comments and answers) serve a slightly different purpose. Answers are supposed to directly answer the question, while comments can be used to correct someone, ask for clarification on a certain point, make a small addition to a post, or provide similar short contributions that are not standalone answers.

Thread structure: The structure of a discussion forum thread can be viewed as a tree, with the initial post at the top, and reply posts branching out below it. Each post is placed below the post it responds to. This structure can be explicit, like in Figure 1.3, or not, like in Figure 1.4.

As background information we would like to very briefly introduce some IR evaluation metrics here, which will be mentioned in different places throughout this survey. Many different evaluation metrics are used for IR tasks using forum data, i.e. post retrieval, and IR in general. For instance, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR) [Voorhees et al., 1999], Precision@n, nDCG [Järvelin and Kekäläinen, 2002], AUC (precision–recall or ROC), and Rank-Biased Precision [Moffat and Zobel, 2008]. Of these, MAP is the most widely used. It is the mean of the average precision at a given cut-off point, calculated over all the queries in a set. The average precision is shown in Equation 1.1, in which N is the cut-off point, P is the precision, and R is an indicator of whether the document retrieved at i is relevant or not.

$$\text{AP@}N = \frac{\sum_{i=1}^{N} P(i) \cdot R(i)}{\#\ \textit{of relevant documents}} \tag{1.1}$$

1.5 Existing data sets

The field of forum related research has long suffered from a lack of publicly available datasets, but this is slowly changing. Over the years, many researchers have constructed their own sets using web forum crawling techniques, for instance using methods described in Wang et al. [2008] or Yang et al. [2009a]. Recently, some forums have started making (part of) their data available to the research community, and many top-tier conferences (e.g. the AAAI International Conference on Web and Social Media) encourage their authors to share their data and provide data sharing services specifically for this purpose. An overview of a large number of public and private datasets used in forum research can be found in Hoogeveen et al. [2015]. In this section we will present only the most important ones, which are openly available for research purposes. They are summarised in Table 1.1.

The Yahoo! Webscope Dataset (L6) Surdeanu et al. [2008] `http://webscope.sandbox.yahoo.com/catalog.php?datatype=l`	4M question and answer pairs from a dump of Yahoo! Answers on 25/10/2007.
The WikiAnswers Corpus Fader et al. [2013] `http://knowitall.cs.washington.edu/oqa/data/wikianswers/`	30M clusters of questions from WikiAnswers,[24] tagged as paraphrases by users. Around 11% of them have an answer.
TREC 2015 LiveQA data Agichtein et al. [2015] `http://trec.nist.gov/data/qa/2015_LiveQA.html`	1000 Yahoo! Answers questions used as queries in the TREC 2015 LiveQA task, including answer strings from systems, with human judgements.
The SemEval Task 3 cQA Dataset Nakov et al. [2015] `http://alt.qcri.org/semeval2015/task3/index.php?id=data-and-tools`	2900 English questions and answers from the Qatar Living Forum,[25] and 1500 Arabic ones from the Fatwa forum on IslamWeb.[26]
StackExchange dump `https://stackoverflow.blog/2009/06/stack-overflow-creative-commons-data-dump/`	A periodical dump of all the data on StackExchange, in XML format.
CQADupStack Hoogeveen et al. [2015] `http://nlp.cis.unimelb.edu.au/resources/cqadupstack/`	All the data of twelve StackExchange forums, in JSON format.
MSR Challenge Dataset Bacchelli [2013] `http://2013.msrconf.org/challenge.php#challenge_data`	Stripped version of a StackOverflow dump, in XML and postgresql formats.
The NTCIR-8 cQA dataset Ishikawa et al. [2010] `http://research.nii.ac.jp/ntcir/permission/ntcir-8/perm-en-CQA.html`	1500 questions and answers from Yahoo! Chiebukuro, the Japanese version of Yahoo! Answers, between April 2004 and October 2005.
The Reddit Comment Corpus `https://www.reddit.com/r/datasets/comments/590re2/updated_reddit_comments_and_posts_updated_on/`	A periodical dump of all the comments. Some of it contains sentiment annotations.
The Quora Dataset `https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs`	400.000 question pairs, annotated for duplicates. Released on 25/01/2017.

Table 1.1: An overview of publicly available forum data sets.

2

Post classification

Discussion forum threads and cQA threads both consist of posts: individual messages written by different users (or by the same user at different times). Posts are the smallest structured unit of a forum. Significant research has been done on post retrieval (§3), both in discussion forums (§3.1) and in cQA archives, where a distinction can be made between question retrieval (§3.2) and answer retrieval (§3.3).

In this section, we discuss classification of posts, which can be used to improve question, answer, or discussion forum post retrieval. We will first look at post type classification in §2.1, considering automatic detection of whether a discussion forum post is an answer post or not.

Then, we will look at classifying questions as *how*-questions, yes/no-questions, etc. in §2.2. This gives us information about the types of answers we might expect, which can potentially be used to improve answer retrieval.

Post quality assessment is discussed in §2.3. In later sections we will mention work that uses this to improve retrieval results.

Finally, we will look at subjectivity and viewpoint classification in §2.4, which just as for the question type, can give us insights into the kinds of answers that would be suitable for the question at hand.

2.1 Post type classification

In this section we look at different ways of classifying discussion forum posts. cQA threads have a clear structure, with a question post at the start, followed by answer posts. Discussion forum threads, however, are more dynamic: new questions can be asked in the middle of threads, topics can shift, and even though the initial post may be a question, it is by no means certain that the following posts contain answers.

While most research in discussion forum post classification focuses on detecting answer posts, there are other types of post classification, for instance finding the post with the most important information [Feng et al., 2006b]. This is the post that contains the *focus* of the thread. Such information can potentially be used in thread summarisation (see §4.5). Answer post detection can also be used to improve forum data access, for instance by enhancing discussion forum post retrieval (see §3.1).

In answer post detection, some researchers have also looked at question post identification [Hong and Davison, 2009, Obasa et al., 2016], but these are often assumed to have already been identified [Huang et al., 2007], or the first post is assumed to be the question [Catherine et al., 2012, 2013]. Work on detecting question *sentences* and their context is discussed in §4.4.

To identify question posts, bag of word features were found not to be useful by themselves, but worked well together with simple rule features (the presence of question marks, and the presence of 5W1H words) and with forum metadata features. These two types of features complement each other [Obasa et al., 2016].

Some research has identified n-grams as the most effective single feature type for identifying question posts, but combinations of simpler features can achieve comparable or better performance, e.g. the authorship of the poster, the number of question marks, the number of 5W1H words and the number of posts in a thread [Hong and Davison, 2009].

To identify answer posts, some features that have been found to be useful are: post author is not question author, the position of the post in the thread, whether the post belonged to the first n posts or not, whether the post is replied to by the question asker, the authority of the poster, and whether a post contains a URL or not [Huang et al., 2007, Catherine et al., 2012, 2013, Hong and Davison, 2009].

All researchers report that using only structural features gives better results than using only content features, and that the best results are obtained by combining the two [Huang et al., 2007, Catherine et al., 2012, 2013, Hong and Davison, 2009]. In some experiments, adding content features affected mainly the precision, while the recall remained stable [Huang et al., 2007].

The lexical similarity between a question and its relevant answer posts is very similar to the lexical similarity between a question and its non-relevant answer posts. Because of this, features which measure the content similarity between a post and the question post are among the least effective features in answer post detection.

SVMs are the most commonly used models in answer post detection research [Catherine et al., 2012, Hong and Davison, 2009, Huang et al., 2007], but experiments have also been conducted using a semi-supervised co-training methodology [Blum and Mitchell, 1998, Catherine et al., 2013]. Specifically, the algorithm starts with a small number of training instances and continues for n iterations. At each iteration, two classifiers are formed by training an SVM learner over two independent feature sets (structural features and pattern features in this case). The two classifiers are then used to classify unlabelled instances, and the predictions with the highest confidence are moved to the current set of labelled instances for training in the next iteration.

The same framework was used to jointly identify acknowledgement posts. A positive acknowledgement post from the author of the question suggests that the problem is solved, while a negative one indicates that the proposed solutions do not work. This is important information for determining whether an answer is useful or not. Good results can be obtained using a limited amount of training data (3 threads) and especially by adding the acknowledgement information [Catherine et al., 2013].

Research has also been done in using the thread structure to identify the dialogue acts of posts. If one such dialogue act is “answer”, then this task is very similar to answer post identification. Such work is discussed further in §4.2.2.

2.2 Question classification

Question classification is about detecting the type of a question, based on the answer type that we would expect for it. For instance, if the question is a yes/no question, then the answer could potentially consist only of the word *yes* or *no*. If the question is asking for people's opinion on some topic, it may be relevant to retrieve several different answers for it, rather than only one correct one. For factual questions the opposite is true. Accurately determining the question type could therefore help when retrieving relevant answers. Question type identification can also help when summarising answers [Liu et al., 2008b]. This is discussed in §4.5.2.

There is no standard hierarchy or list of question types. Most researchers develop their own, with the number varying from 2 to around 150 types. The division in types differs not only in the number of types, but also in what exactly is meant by 'type'. Apart from the examples given in the previous paragraph, some researchers have instead used types that are closer to topics, based on the semantics of the question, rather than on the purpose or expected answer format [Metzler and Croft, 2005, Li and Roth, 2002, Suzuki et al., 2003]. Such question type taxonomies are much more fine-grained than taxonomies based on the format of the question or the types of answers they are expected to receive, including over 50 types.

Automatic question type classification has been researched extensively outside of the cQA and the discussion forum domain [Voorhees et al., 1999, Harabagiu et al., 2000, Hovy et al., 2001, Li and Roth, 2002, Suzuki et al., 2003, Silva et al., 2011, Zhang and Lee, 2003, Hermjakob, 2001, Huang et al., 2008, Metzler and Croft, 2005]. Lytinen and Tomuro for instance, have researched question classification as a way to improve the performance of their question-answering system FAQFinder [Tomuro and Lytinen, 2001, Tomuro, 2002, Lytinen and

Tomuro, 2002],[1] and in the TREC QA task[2] 1999-2004 many participants have used question types to improve the results of their systems [Ittycheriah et al., 2001, Harabagiu et al., 2000, Hovy et al., 2001].

In the forum research community, surprisingly little attention has been paid to incorporating question type information in answer retrieval models. One study uses a question type hierarchy based on Rose and Levinson's taxonomy of search engine queries [Rose and Levinson, 2004], to assign types to questions in cQA data, in order to identify the types of the answer posts [Liu et al., 2008b]. However, no question classification experiments are presented. Instead, the researchers assume these types can be assigned automatically, and focus on using answer types to help with answer summarisation (see §4.5).

In other work that mentions question type classification, the main focus generally lies in answering only certain types of questions. Question type classification is thus often treated only as a necessary preprocessing step, and simple approaches are taken, such as a supervised machine learning model with textual features [Pechsiri and Piriyakul, 2016], or a pattern matching system using regular expressions [He and Dai, 2011].

Finally, questions can also be classified based on their topic. This can help with automatically assigning questions to a particular category, or automatically choosing appropriate tags for it. Experiments in this space include both supervised and semi-supervised models, mainly using textual features [Li et al., 2016].

2.3 Post quality assessment

One of the most important directions in post-level analysis is automatically assessing post quality, to help users better access high quality information. Good access to high quality content has a high impact on user satisfaction [Agichtein et al., 2008] and is the best way to retain existing users and attract new ones [Le et al., 2016, Liu et al., 2008a]. More users means a larger community of potential answerers, which

[1] `http://faqfinder.mines.edu/`

[2] `http://trec.nist.gov/data/qa.html`

contributes heavily to a cQA archive's success [Harper et al., 2008].

Most forums have a user voting system in place to enable the community to distinguish high quality content from low quality content, but user generated votings for answers are not always reliable [Suryanto et al., 2009, Jeon et al., 2006], and are often missing [Burel et al., 2012, Chai et al., 2010, Sun et al., 2009].

Being able to automatically determine the answer quality and ranking answers accordingly is especially important for large cQA archives, because the more users a cQA archive has, the less reliable (that is accurate, complete and verifiable) the answers become [Shachaf, 2010].

Research on post quality has mainly focused on two task settings: (1) classification into either "good" or "bad" posts [Agichtein et al., 2008, Blooma et al., 2012, Joty et al., 2015, Le et al., 2016, Lui and Baldwin, 2009, Shah and Pomerantz, 2010, Weimer et al., 2007, Weimer and Gurevych, 2007]; and (2) identifying the best answer for a given question [Adamic et al., 2008, Burel et al., 2012, 2016, Gkotsis et al., 2014, Blooma et al., 2008, Shah and Pomerantz, 2010, Shah, 2015, Tian et al., 2013a, Dom and Paranjpe, 2008].

Automatic post quality assessment has largely been treated as a supervised classification task, with the main focus on feature engineering. Some early work used forum data [Weimer et al., 2007, Weimer and Gurevych, 2007, Wanas et al., 2008, Lui and Baldwin, 2009], but in more recent years the focus has shifted completely to cQA data [Burel et al., 2012, 2016, Chua and Banerjee, 2015a, Dalip et al., 2013, Shtok et al., 2012, Dror et al., 2013]. Post quality can also be incorporated in an answer retrieval function (see §3.3.2).

The best answer as selected by the asker is often used as the gold standard, and the asker's choice corresponds well with the choice of the other users, as measured by the number of upvotes and downvotes the community awarded certain answers [Burel et al., 2012, Burel, 2016]. However, Jeon et al. [2006] manually annotated a test set of questions to use as their gold standard. They used the best answer as selected by the asker as one of their features, and found that it did not have the highest correlation with the quality of the answer.[3] Community-

[3] See Kim et al. [2007] and Kim and Oh [2009] for research into the criteria that

generated answer scores or ratings on the other hand, *are* a good predictor of answer quality [Burel et al., 2012, Bian et al., 2008a], which shows how valuable community feedback is on cQA websites. There is a high level of agreement over what a good quality answer looks like [Shah and Pomerantz, 2010, Burel et al., 2012, Burel, 2016]. The only problem here is that many answers do not have any rating [Burel et al., 2012, Burel, 2016].

Another way to serve users high quality content is to amalgamate multiple answers to a question, in order to give the user the most complete answer. It has been found that the quality of an amalgamated response is indeed better than that of the *best answer* in terms of completeness and verifiability, but not in terms of accuracy [Shachaf, 2011].

2.3.1 Features for post quality classification

We found that more than 150 different features were used in the papers discussed in this section, including some interesting ones like the average number of syllables per word, readability measures, and n-grams of POS tags to capture grammaticality to some degree [Agichtein et al., 2008]. These features can be divided roughly into five categories: *user or community features*, *content features* (which can be subdivided into *lexical* and *syntactic*), *structural or thread features*, *forum-specific features*, and *usage features*, like how often an answer has been clicked on. *Forum-specific features* include a mixture of features that are all specific to a certain forum or cQA archive, and cannot easily be obtained for other forums. An example of this from the Brainly forum[4] is the answerer's school grade level. The specific experimental setting is important when choosing features, because their impact varies greatly depending on whether they are used to try to identify good answers in the full dataset (global search), or to distinguish the best answer from the others in a given thread (local search) [Burel et al., 2016].

question askers use to select the best answer.

[4] `http://brainly.com/`

Many different observations can be distilled from the answer quality literature, starting with the fact that good questions attract good answers [Jeon et al., 2006, Yao et al., 2013, Agichtein et al., 2008, Yao et al., 2015, Souza et al., 2016], and that there is a high correlation between the quality of answers and the knowledgeability of the users writing them [Jeon et al., 2006, Burel, 2016, Agichtein et al., 2008, Le et al., 2016, Shah and Pomerantz, 2010]. Users that focus more on answering questions in one category have their answers selected as best more often than broad posters [Adamic et al., 2008, Suryanto et al., 2009].

Several characteristics of good answers have been identified: answers with more diverse vocabulary are more likely to be best answers [Burel et al., 2016], answers that are often referenced by other answerers are more likely to be good [Wanas et al., 2008], and answers that receive more replies or comments from other users are more likely to be of high quality [Wanas et al., 2008, Tian et al., 2013a]. Besides this, answers that are posted earlier have a higher chance to be selected as the best answer [Tian et al., 2013a, Hong and Davison, 2009, Calefato et al., 2016], and the best answer is usually the most different from the others [Tian et al., 2013a].

The single feature that is most often found to be a good predictor of answer quality is *answer length* [Adamic et al., 2008, Weimer and Gurevych, 2007, Jeon et al., 2006, Agichtein et al., 2008, Le et al., 2016, Burel et al., 2016, Tian et al., 2013a, Gkotsis et al., 2014, Calefato et al., 2016]. Only occasionally do researchers report the opposite result [Burel et al., 2012]. Forum specific features are also often cited as good features [Burel et al., 2012, Burel, 2016, Jeon et al., 2006, Le et al., 2016], but they have the drawback of not being available for all datasets.

User features have been studied in depth and are generally found to be useful [Lui and Baldwin, 2009, Yang et al., 2011, Agichtein et al., 2008, Burel et al., 2012, 2016, Agichtein et al., 2008, Shah, 2015, Suryanto et al., 2009, Hong and Davison, 2009, Dom and Paranjpe, 2008, Adamic et al., 2008, Ponzanelli et al., 2014, Maleewong, 2016, Molino et al., 2016], and more robust than textual features [Le et al., 2016], although the opposite has also been noted: Blooma et al. [2008]

found that non-textual features were less predictive than textual features. However, the difference in usefulness between content features and user features may be more complex than this, because user reputation is not independent of textual content [Bian et al., 2009, Gkotsis et al., 2014] The answerer's question-answering history can also be used to estimate the probability of his or her answer being chosen as the best one [Dom and Paranjpe, 2008].

Some research has looked into how the effectiveness of answers is related to the question's type (see §2.2) and the reputation of the answerer [Chua and Banerjee, 2015a, Hong and Davison, 2009, Huang et al., 2007]. Authoritative users and novice users were found to contribute positively, but in different ways. While authoritative users submit more detailed and high-quality answers, novice users submit more readable answers [Chua and Banerjee, 2015a].

Thread-level features and interaction features, e.g. the number of distinct interactions between the asker and the answerer, the difference between the posting time of the question and the answer, or the proximity of the answer to an answer posted by the asker, were found to make a unique contribution, even though they were only available for a small number of instances [Shah, 2015, Barrón-Cedeno et al., 2015].

For content features it has been noted that lexical content features outperform syntactic content features [Weimer and Gurevych, 2007]. While some research has found that a high accuracy can be obtained without the use of content or language dependent features [Wanas et al., 2008], their usefulness can be greatly enhanced by taking relations between answers into account [Burel et al., 2016, Tian et al., 2013a, Gkotsis et al., 2014, Burel et al., 2012]. This can be done by normalising the features by the values of the other answers in the thread and in that way turning features into ratios. Different ways of doing this are explored by Burel et al. [2016].

When we look at style, grammaticality and readability have been identified as useful features for predicting the best answer [Maleewong, 2016, Molino et al., 2016]. Lexical similarity features have been shown to outperform distributional semantic features when used in isolation, but not when combined with other features (e.g. user features and text quality features) [Molino et al., 2016].

When answer quality is incorporated into an answer retrieval model, query feedback features based on the work of Zhou and Croft [2007] have been found to be very helpful [Shtok et al., 2012].

It is unclear whether human assessments of quality, like *accuracy*, *completeness* and *reliability*, are more predictive than automatically extracted user and content features [Zhu et al., 2009, Blooma et al., 2011]. Some researchers have found that they are [Blooma et al., 2012], while others have found the opposite [Shah and Pomerantz, 2010]. Many of such aspects (informativeness, novelty, etc.) are highly correlated [Shah and Pomerantz, 2010] and can be estimated automatically [Katerattanakul and Siau, 1999, Liu and Huang, 2005, Rieh and Belkin, 1998, Rieh, 2002, Strong et al., 1997, Wang and Strong, 1996]. The estimated versions have been found to be good predictors of answer quality [Blooma et al., 2008, Molino et al., 2016]. Good results have been obtained by adding many of the hand-crafted features we have mentioned to a deep convolutional model [Suggu et al., 2016].

Some research has shown that capturing the interaction between different types of features, for instance by incorporating weakly hierarchical lasso, results in better performance than simply concatenating the features [Tian and Li, 2016]. And finally, as always, the usefulness of different types of features depends heavily on the dataset used [Weimer and Gurevych, 2007, Burel et al., 2012, Burel, 2016].

One aspect that none of the research discussed in this section has incorporated is the possibility of all the answers to a question being of bad quality [Burel et al., 2012]. This problem of recognising when no good answer exists is very close to *solvedness prediction*, which we discuss in §4.1.

2.3.2 Completeness and answerability

So far we have focused mainly on the quality of *answers*, but since the quality of answers is heavily related to the quality of *questions* [Jeon et al., 2006, Yao et al., 2013, Agichtein et al., 2008, Yao et al., 2015, Souza et al., 2016], it is worth looking at question quality too.

According to one estimate, around 15% of incoming questions in

Yahoo! Answers remain unanswered [Shtok et al., 2012], and on average around 11% on the StackExchange sites [Convertino et al., 2017], usually because they have been asked before, or because they are of poor quality, for instance by being overly broad, by supplying excessive information, or because they are socially awkward. Incorrect tags are another reason why questions remain unanswered [Asaduzzaman et al., 2013]. One of the main reasons for questions not receiving answers is that they do not contain enough information for other users to be able to provide a good answer to it [Shah et al., 2012, Convertino et al., 2017, Zhou and Fong, 2016]. We call such questions *incomplete.*

Some work has been done on predicting the completeness or resolvability of questions. In discussion forums such information can be used to assess a thread's utility for troubleshooting purposes [Baldwin et al., 2007]. Most work in this space focuses on cQA archives, however, in which predicting the completeness or answerability of questions could be used to assist users in rephrasing their question in a way that makes them more likely to receive good answers [Kitzie et al., 2013, Baltadzhieva and Chrupała, 2015]. Both tasks are difficult, with low agreement among annotators [Baldwin et al., 2007, Yang et al., 2011], partially because of the "inherent randomness in the answering process" [Dror et al., 2013], although one study found that the question score as received by the community via votes is a good predictor of question quality [Ahn et al., 2013].

For cQA question answerability prediction, many contradictory results have been presented, which may be due to the specific forum being researched. For instance, one study found that expert users were more likely to receive answers than novice users ([Yang et al., 2011], Yahoo! Answers), while another found that both expert and novice users were more likely to receive answers than medium users ([Chua and Banerjee, 2015b], Stack Overflow), and another again found that user reputation was not a helpful feature [Zhou and Fong, 2016]. Some studies found that short questions were more likely to be answered ([Saha et al., 2013, Chua and Banerjee, 2015b, Zhou and Fong, 2016], Stack Overflow), while another found that both short and long questions were more likely to be answered than medium length ones ([Yang et al., 2011], Yahoo! Answers).

The time of posting was found to be a good predictor in some studies ([Chua and Banerjee, 2015b, Zhou and Fong, 2016], Stack Overflow, and [Dror et al., 2013], Yahoo! Answers), but not in another ([Yang et al., 2011], Yahoo! Answers), and it is unclear whether adding code snippets to a question attracts answers ([Treude et al., 2011], Stack Overflow) or not ([Chua and Banerjee, 2015b, Zhou and Fong, 2016], also Stack Overflow). Some researchers found that a question was less likely to receive answers if it contained linguistic errors ([Kitzie et al., 2013], Yahoo! Answers), while another found that this did not make a difference ([Chua and Banerjee, 2015b], Stack Overflow).

For other predictors, more consistent results were found. Complex questions and overly polite ones are both less likely to receive answers [Yang et al., 2011, Chua and Banerjee, 2015b, Zhou and Fong, 2016]. Subjectivity was found to be a good predictor [Yang et al., 2011, Chua and Banerjee, 2015b, Zhou and Fong, 2016], stylistic features were found to be helpful [Correa and Sureka, 2014], and adding information that shows a personal connection to the topic elicits more replies [Burke et al., 2007], both in subjective and in information questions [Suzuki et al., 2011]. Users prefer to answer questions that start with a question word [Zhou and Fong, 2016]. Even if an incomplete or otherwise low quality question does receive answers, the first one typically takes longer to arrive than for a high quality question [Li et al., 2012, Souza et al., 2016].

Instead of treating the problem as a binary classification task, it can be framed as a regression task in which the number of answers a question will receive is predicted [Dror et al., 2013]. Another variant of answerability prediction is to predict whether a question will be closed by the community due to it being of low quality [Madeti, 2016]. Related work has also looked at predicting the popularity of a question by making use of user ratings [Sun et al., 2009], and at analysing questions about code that are difficult to answer [LaToza and Myers, 2010].

Questions may be complete, but still not receive any answers. This could be simply because the potential answerer has not seen the question [Furlan et al., 2012]. Question recommendation and question routing, which we discuss in §5.3.1 can help in such situations.

2.4 Subjectivity and viewpoint classification

One aspect of posts we have not looked at so far is whether they ask for or express an opinion or not. Automatically determining this is called *subjectivity classification*, and falls under *intent detection*.

In discussion forums, identifying whether a thread is seeking opinions or looking for factual information can help improve forum search and help forum administrators monitor abusive conversations [Biyani et al., 2012]. After this, we can go one step further and automatically determine the viewpoint. This information can then be used for automatic summarisation of discussion forum threads for instance, which we discuss in §4.5.

In cQA archives, subjective questions spark different kinds of answers to objective or factual questions. The definition of what constitutes a good answer also differs for these two question types. A good objective answer often contains references, for instance, while a good subjective answer does not. Instead, it should contain different viewpoints on the topic of the question, with arguments for and against. Classifying subjectivity in questions allows us to anticipate the kinds of answers needed, and can therefore help to retrieve more appropriate answers. In Yahoo! Answers 66% of the questions are subjective [Li et al., 2008b], and so adding this distinction to an answer retrieval model can potentially increase the effectiveness of the system considerably. Similarly, it can help improve best answer selection [Kim et al., 2007, Adamic et al., 2008].

Much work on viewpoint classification and sentiment analysis focuses on consumer reviews (see for instance [Schouten and Frasincar, 2014], or [Medhat et al., 2014] for a survey on this topic). Ideas from those applications could potentially be useful in forum post viewpoint classification too, because consumer reviews share some similarities with forum posts (e.g. length or level of (in)formality).

2.4.1 Subjectivity and viewpoint classification in cQA archives

Most researchers working on subjectivity classification in community question-answering treat the problem as a binary classification task, where questions are classified as being either *subjective* or *objective* [Li et al., 2008b,a, Aikawa et al., 2011, Harper et al., 2009, Amiri et al., 2013, Zhou et al., 2012d, Gurevych et al., 2009].[5] However, a third kind of question can sometimes be distinguished: *social questions* [Chen et al., 2012].

Subjective and social questions differ mainly in their goal. Subjective questions, like objective questions, are information seeking questions, asking for personal opinions or general advice. Social questions such as `Any1 near Newyork city?`, on the other hand, are only asked with the goal of having social interactions. On some cQA sites, like StackExchange, the community takes an active role in deleting such questions. In Yahoo! Answers, on the other hand, such questions are perfectly fine, and in some subforums, like Family Relationships, they are the most common kind, and it may be sensible to recognise them as a separate class [Chen et al., 2012].

Early work on subjectivity classification of cQA questions made use of both the question and its answers [Li et al., 2008b,a]. While adding answers did improve the results, researchers have since argued that for new questions, there are no answers available and it is therefore better not to use the answer data when constructing a subjectivity classification model [Aikawa et al., 2011, Harper et al., 2009, Amiri et al., 2013, Zhou et al., 2012d, Gurevych et al., 2009, Chen et al., 2012].

Most researchers have treated the task as a supervised classification task and have manually annotated Yahoo! Answer data for this [Li et al., 2008b, Aikawa et al., 2011, Harper et al., 2009, Amiri et al., 2013]. Co-training [Blum and Mitchell, 1998], a semi-supervised model, has been explored as a method to reduce the manual annotations needed, and it was found that only 25-30% of the training examples were needed

[5] Different names are used to denote these two categories: subjective vs objective, subjective vs non-subjective, positive vs negative (subjectivity), opinion questions vs factual questions, informational questions vs conversational questions, etc.

to achieve similar results to a regular supervised method [Li et al., 2008a, Chen et al., 2012]. Another semi-supervised approach addressed the class imbalance problem by using an adaptive Extreme Learning Machine [Huang et al., 2006, Fu et al., 2016a].

The need for labelled data can also be alleviated by using heuristics as measures of subjectivity, and using these to generate more training data without human intervention. Examples of this include the number and distribution of answer likes and votes, the appearance of references in answers, the number of answers, and the appearance of polls or surveys [Zhou et al., 2012d].

There is no consensus over what constitute good features. Word n-grams, for instance, are reported to be an effective feature by some [Zhou et al., 2011b], while others found them to be redundant [Li et al., 2008b,a, Aikawa et al., 2011]. Some found that the proportion of subjective questions increased as question length increased [Zhou et al., 2011b], while others noted that question length was not a useful feature [Harper et al., 2009].

All studies used textual features. Character trigrams have been reported to be useful [Li et al., 2008b]. Dependency features have been explored, but have been found to be less predictive than word bigrams [Aikawa et al., 2011]. Adding metadata features, like the time a question was posted, or the topic of the subforum, is useful [Chen et al., 2012, Harper et al., 2009], but only if a dataset contains data from different subforums, because the distribution of subjective and objective questions varies across different subforums [Chen et al., 2012]. In one study, the topic of the subforum was found to be a more useful feature than a bag of words [Harper et al., 2009].

Some interesting observations have been made regarding subjective questions, which could be used to design new features: on average, subjective questions have a higher punctuation density, more grammatical modifiers, and more entities [Zhou et al., 2011b]; conversational users have more neighbours in a user network graph than informational users [Harper et al., 2009]; and the presence of the word *you* is a strong indicator that a question is subjective [Harper et al., 2009].

An interesting approach was taken by Gurevych et al. [2009]. They manually constructed a lexicon of words and multiword expressions, and a set of part-of-speech sequences. All of these were assigned a subjectivity weight. To calculate the subjectivity of a question, the weights of the words and part-of-speech sequences appearing in the question were summed. A threshold was set on the score, based on the number of sentences in a question. Using this simple technique, they obtained an F1-score of 0.86, which is higher than many more complex methods.

There are two difficult problems in subjectivity detection which have not received much attention. Firstly, questions about current events may look factual while they are not. An example of this can be found in Aikawa et al. [2011]: *Why was Mr. Fukushima (football player) fired?* seems like an objective question, but in reality the reason was not known at the time the question was posted, and so this question was meant to spark a discussion. It is currently not clear how such subjective questions could be identified.

Secondly, opinion words, as identified via opinion word lists, are strong indicators of subjectivity, but some subjective questions do not contain any opinion words from such lists. The currently available opinion word lists are incomplete for cQA data, especially in terms of slang and other informal internet language [Amiri et al., 2013].

2.4.2 Subjectivity and viewpoint classification in discussion forums

Compared to cQA, less interest has been shown in subjectivity detection in forum threads. Most work in this area has been done by Prakhar Biyani in his PhD thesis [Biyani, 2014]; he used the results of his subjectivity classification work to improve thread retrieval [Biyani et al., 2015] (see §4.3).

As in most subjectivity identification research using cQA data, the problem was treated as a classification task: complete threads were classified rather than individual posts. We will discuss this work here to contrast it with the cQA subjectivity classification work. Several kinds of features were explored: structural features, dialogue act features, subjectivity lexicon-based features, and sentiment features. Thread length was reported to be the most indicative feature [Biyani et al., 2012].

Most errors were made with short subjective threads, where the intended discussion did not happen, or objective threads that experienced topic drift and were therefore longer than usual [Biyani et al., 2014].

After classifying posts as subjective or not, we can go one step further and classify the subjective posts in various ways. A distinction can be made between emotional statements and evaluative opinions [Zhai et al., 2011], or between sentences that express an attitude towards the recipient and sentences that do not [Hassan et al., 2010]. Posts can be classified as either agreeing or disagreeing with the previous post [Fortuna et al., 2007], or they can be clustered based on the viewpoint expressed in them [Qiu and Jiang, 2013].

Alternatively, the threads in a discussion forum can be clustered based on the topics that are discussed in them. Using this information, users can be divided into sets based on the homogeneity of their opinions. Users that post in the same topic cluster can then be assumed to agree if they are in the same set, or disagree if they are in a different set [Georgiou et al., 2010].

2.5 Post classification summary

In this chapter we discussed several ways in which forum posts can be classified. We looked at post type classification, question type classification, quality assessment, and subjectivity and viewpoint classification. Each of these tasks can potentially be used to improve post retrieval. For instance, we may be interested in only retrieving posts of a certain type. Depending on the type of question that is asked, it may be better to retrieve objective rather than subjective posts, or vice versa, and we want to retrieve posts of high quality. In the next chapter we will move our focus from post classification to post retrieval.

3

Post retrieval

Forums contain a wealth of information that could be useful for a large number of people. However, searching through these archives presents specific challenges, due to the noisy nature of the data. When retrieving information from forums, there are two logical retrieval units to consider: posts and threads. In cQA archives, retrieving posts is a logical choice, because they are usually self contained. Threads in cQA archives generally speaking do not have a discourse structure. In discussion forums on the other hand, retrieving one single post may not give enough information to understand the topic under discussion, and hence retrieving threads may be a better way to resolve a user's information need. However, since the distinction between cQA archives and discussion forums is not clear cut, and many examples reside in the grey area in the middle, post retrieval and thread retrieval strategies can be applied in both contexts. Thread retrieval is discussed in §4.3. Here, we will focus on retrieval strategies at the post level.

3.1 Discussion forum post retrieval

In this section we will discuss post retrieval from discussion forums in response to a search query. Post retrieval from cQA archives will be discussed in §3.2 and §3.3. Retrieving posts from discussion forums is different from retrieving posts from cQA archives due to the different nature of these two related sources. The main difference is the internal structure of the threads, which can be used when retrieving posts from discussion forums.

With respect to the methodologies used for forum IR, ranking algorithms are commonly used. Early research explored ways to train supervised learn-to-rank functions, or adopted unsupervised semantic similarity measures for ranking. Later research has mainly adopted language model-based IR approaches, which are often smoothed based on thread structure. We review these next.

In learn-to-rank experiments it has been found that features based on the linking structure (e.g. the position of the post in the thread, or the number of descendent posts) are effective, while user based features (e.g. the average length of the user's posts, or the number of threads the user has initiated) are not [Xi et al., 2004].

In the application of language modelling to discussion forums, most research has focused on different ways of smoothing the models by making use of the thread structure. As a baseline, the language model for posts can be smoothed based on the whole collection. Two different smoothing approaches have been explored: one in which the collection smoothing is *replaced* by thread structure based smoothing [Duan and Zhai, 2011]; and one in which the collection smoothing is *extended* by adding a second smoothing step, based on the thread structure [Seo et al., 2009, 2011]. The latter work did not examine which smoothing step in this mixture model contributed most to the performance. Furthermore, these two approaches to thread structure smoothing have not been directly compared and at this point it is unclear which one works best.

Each post in a thread has a so-called context. This context is based on the thread structure that the post is part of. One such context is the entire dialogue leading to the post, i.e. the path from the initial

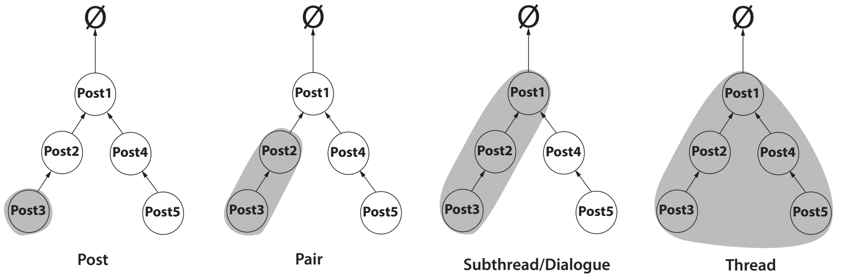


Figure 3.1: Different contexts in a thread structure, as used by Seo et al. [2009, 2011].

post to the relevant post in the thread tree. Another context is the set of post-reply pairs directly involving the post. These two structures are illustrated in Figure 3.1.

Adding a smoothing step based on pair and dialogue contexts significantly improves the retrieval results using only the posts or the posts smoothed by the whole thread, and pairs work even better than dialogues [Seo et al., 2009, 2011]. This model can also be used to validate results in automatic thread structure discovery [Wang et al., 2011a], which we discuss in §4.2.

Simpler ways to select posts to be used for smoothing have also been explored, e.g. using all posts preceding the target post. When using multiple posts in the smoothing, different weights can be applied to the posts based on, for instance, their distance to the target post [Duan and Zhai, 2011]. Fairly good performance can also be achieved by smoothing using only the first post of the thread [Duan and Zhai, 2011].

A slightly different, but related, task is that of ranking posts *within* one thread. Discussion forum threads can become very large, spanning many pages, with many new posts being added every hour. The quality and originality of the posts in such threads may vary considerably and large time savings can be made by ranking them according to the quality of their contribution, even at the cost of losing (part of) the discourse structure.

Discussion forum post ranking is similar to answer ranking in cQA archives (see §3.3). It differs not only in what is retrieved and ranked (posts within one thread vs. all relevant posts in the collection), but also in what the queries are. Rather than keyword queries, in this case the initial post can be used as the query. While answer ranking in cQA archives is a very active field of research, for discussion forum data much less research has been conducted, even though it has been shown that the original order of posts in a thread, which is chronological, makes it difficult for users to find high quality content [Han et al., 2014].

Within-thread post ranking has not received much attention from the research community yet. So far the only approach that has been explored is a supervised machine learning approach using a combination of topic model features, content surface features (e.g. length, capital word frequency), and forum-specific features (e.g. URL count, quote count) [Han et al., 2014].

3.2 CQA question retrieval

In this section we will discuss one type of post retrieval from cQA archives: question retrieval. One problem that many cQA archives suffer from is that users ask a question that has already been asked before. In some categories of Yahoo! Answers, for instance, as many as 25% of the new questions are repeated questions [Shtok et al., 2012]. This redundancy is good for IR, but causes annoyance among question answerers. It is bad for the optimisation of resources (the answerers) and therefore bad for the community.

Some cQA websites have a mechanism in place that allows the community to flag new questions as a duplicate of an archived question (e.g. StackExchange), but some do not offer any way to keep the question archive 'clean' (e.g. Yahoo! Answers). Ideally, there should be one canonical version of each question. If we can identify questions that have been asked before, we can kill three birds with one stone:

1. we reduce the redundancy in the archived data, resulting in potentially better data access
2. we increase user satisfaction by giving people an answer straight away, instead of them having to wait for the community to answer

Dead fly on hot oily pan
Fly lands on hot oily pan and duly dies. Kashrut implications of the pan not the fly :p

Does a pot that cooked food with a bug in it need to be kashered afterward?
If one were cooking a pot of pasta or a soup and later found a bug or many bugs in the pasta or soup, would the pot need to be re-kashered after it was thoroughly cleaned?

Figure 3.2: An example of two questions that are duplicates but have no lexical overlap besides a handful of function words. Source: `http://judaism.stackexchange.com`

their question

3. we save the community the manual effort of labelling repeated questions as duplicates of archived ones (and the annoyance caused by this)

Additionally this could help people to adjust their query if results are returned that are not quite what they are looking for. Manual duplicate detection is not perfect and it does happen that a new question is flagged as a duplicate, while it is actually not according to the question asker. This could be avoided if the question asker is given an idea of how his or her question might be interpreted by showing similar questions.

Most cQA forums already offer a search function, but the number of duplicate questions that are undetected indicates that there is substantial room for improvement here. One of the main problems to overcome is the so called *lexical gap* between different questions with similar meanings, as observed by Bernhard and Gurevych [2008], who earlier observed the same problem for questions and answers in FAQ pages [Berger et al., 2000].

Figure 3.2 shows an example of two questions that are duplicates, but which do not have any words in common, other than a few function words (*and*, *of*, *the*, and *not*).

Figure 3.3 shows an example of the opposite problem: two questions with a high lexical overlap, but which are not actually duplicates.

Does one need dry hands before washing?
I have always noticed people that make sure their hands are perfectly dry before washing. I.e. washing for eating bread. Why do people do so? Where does this apparent stringency come from?

Wash hands before eating or touching bread?
Is it important to wash one's hands before eating or is also the preparation of the meal important as well? Should one wash before handling bread but not make a brachah, or is washing only relevant to right before one begins to eat their meal?

Figure 3.3: An example of two questions that are not duplicates but have a relatively high degree of lexical overlap. Source: `http://judaism.stackexchange.com`

One question asks about whether you need to wash your hands before touching bread, and the other one asks whether your hands need to be dry before you wash them (for instance before touching bread).

These examples illustrate the fact that lexical overlap (or a lack thereof) by itself is not always enough to determine whether two questions are duplicates or not. It is exactly this problem that researchers have tried to tackle in their question retrieval experiments.

CQA questions consist of several different components: a title, a body or description, the answers, and optionally comments to the question and the answers. When computing the similarity between two questions, researchers need to choose which of these components to use. Including more provides more context, but can also introduce more noise, and so it depends on the method what the best unit for comparison is.

Several phases can be identified in the development of models for question retrieval: the use of statistical translation models, the use of topic models, and more recently, the use of neural networks. Each of these attempts to overcome the lexical gap problem by somehow learning relationships between words. In the following subsections we will discuss them in turn.

3.2.1 Question retrieval: statistical translation models

In statistical machine translate word alignment models, translation models are used to learn relations between words in different languages, in the form of translation probabilities between word pairs, or pairs of phrases [Brown et al., 1993, Och et al., 1999, Marcu and Wong, 2002, Koehn et al., 2003]. The same principle can be applied to monolingual parallel data, to discover synonyms and related words [Koehn and Knight, 2002, Brockett et al., 2006, Garera et al., 2009]. As such, the models can be used to alleviate the lexical gap problem.

The idea behind using a statistical translation model for duplicate question retrieval is to view the input question as a monolingual translation (i.e. a paraphrase) of an archived duplicate question. For each word or phrase in an input question, it is possible to calculate the probability of it being a translation of a certain word or phrase in an archived question, using the notion of word alignment from statistical machine translation [Brown et al., 1993]. Combining these probabilities, as shown in Equation 3.1 below, gives the translation probability of the full sentence or paragraph (a single question in this case).

In order to train the model a large set of question pairs, that are labeled as either duplicates or not, is required. During training, the translation model will learn to assign a high translation probability to synonyms or paraphrases. The translation probability of a potential duplicate question can therefore be used as a score for its semantic similarity to the input question.

Berger and Lafferty [1999] were the first to propose the use of a monolingual translation model for information retrieval tasks, and later for question-answering tasks [Berger et al., 2000]. The translation model that they used is IBM Model 1 [Brown et al., 1993]. This model learns translation probabilities from word co-occurrence statistics. When applied to an information retrieval problem, IBM Model 1 can be presented as follows:[1]

$$P(\mathbf{q}|D) = \prod_{w \in \mathbf{q}} P(w|D) \tag{3.1}$$

[1] Equations are taken from Xue et al. [2008]

$$P(w|D) = \frac{|D|}{|D|+1} P_{tr}(w|D) + \frac{1}{|D|+1} P(w|null) \tag{3.2}$$

$$P_{tr}(w|D) = \sum_{t \in D} P(w|t) P_{ml}(t|D) \tag{3.3}$$

where $\mathbf{q}$ is the query, D the document, and $|D|$ the length of the document; $P(w|t)$ is the probability of word w being a translation of word t; and $P_{ml}(t|D)$ is the maximum likelihood of word t appearing in document D. In Equation 3.1, add-one smoothing is applied to ensure non-zero probabilities for words that do not appear in the training data.

Jeon et al. [2005b] were the first to apply a translation model (IBM Model 1) to a question retrieval task using cQA data and outperformed several standard IR models: a query-likelihood language model [Ponte and Croft, 1998, Ponte, 1998], the BM25 Okapi model [Robertson et al., 1994], and simple cosine similarity. As training instances, they used questions with similar *answers*. In earlier research this had been shown to be an effective way of finding duplicate questions [Jeon et al., 2005a].[2]

Researchers have built on the statistical translation model in many ways. For instance, a phrase-based model can be used instead of a word based model [Zhou et al., 2011a], or external resources (like Wikipedia) can be used to recognise entities or multiword expressions in the questions and to use those as units in the translation model [Singh, 2012]. User intent, encoded in the question type (e.g. "yes/no-question", "recommendation question", "navigational question", etc.; see §2.2), can be incorporated to narrow down the search space [Wang et al., 2011b, Chen et al., 2016b].

There is an important issue that must be considered when using translation models on monolingual data: self-translation probabilities. When the source and target languages are the same, every word has a certain probability of being translated into itself. The question is

[2] Jeon et al. [2005a] show that it is possible to identify duplicate questions by comparing their answers. Unfortunately they do not compare their results against scores obtained by applying these methods to questions directly, instead of to their answers.

what these probabilities should be. Setting these probabilities to a high value means a high lexical overlap is favoured. This will result in a high precision, but a low recall, because it reduces the synonym detection power of the translation model. Setting these probabilities to a low value on the other hand also impairs performance because it does not value lexical overlap enough.

The problem of the self-translation probabilities can be addressed by combining translation probabilities generated by IBM Model 1 linearly with maximum likelihood estimations [Xue et al., 2008]. This new model is called a translation-based language model (TRLM), and has been extremely influential in the field of question/answer retrieval; it was regarded as the state of the art for several years. The new model is defined by the following equations, where $\mathbf{q}$ is the query. (q, a) is a QA-pair from the CQA-data. C is the full collection of QA-pairs in the CQA-data:

$$P(\mathbf{q}|(q,a)) = \prod_{w \in \mathbf{q}} P(w|(q,a)) \tag{3.4}$$

$$P(w|(q,a)) = \frac{|(q,a)|}{|(q,a)| + \lambda} P_{mx}(w|(q,a)) + \frac{\lambda}{|(q,a)| + \lambda} P_{ml}(w|C) \tag{3.5}$$

$$P_{mx}(w|(q,a)) = (1-\beta) P_{ml}(w|q) + \beta \sum_{t \in q} P(w|t) P_{ml}(t|q) \tag{3.6}$$

Three more changes to the translation model [Jeon et al., 2005b] were explored:

- Question-answer pairs were used as training data for the translation model, instead of question-question pairs.[3]

- Different ways to combine P(Q|A) and P(A|Q) when learning the translation probabilities were compared, and the most effective strategy was found to be *pooling*. In this strategy the QA and AQ-pairs are 'pooled' together into one set which is used as

[3] Question-answer pairs have been shown by to be better training data than duplicate question pairs [Lee et al., 2008], although this paper is not cited by Xue et al. [2008], so they may not have been aware of this.

training data. This gave better results than using only P(Q|A), only P(A|Q), or combining P(Q|A) and P(A|Q) linearly to obtain word-to-word translation probabilities.

- Query likelihood scores of the answer part were added to the translation-based language model scores of the question part of the candidate QA-pairs, changing $P_{mx}(w|(q,a))$ in Equation 3.6 to:

$$P_{mx}(w|(q,a)) = \alpha P_{ml}(w|q) + \beta \sum_{t \in q} P(w|t) P_{ml}(t|q) + \gamma P_{ml}(w|a) \tag{3.7}$$

Several studies report that using only the answers when trying to return results to an input question does not give good retrieval results [Jeon et al., 2005b, Burke et al., 1997].[4] However, incorporating the answer part to add extra weight to the potential duplicate questions can improve the retrieval results [Xue et al., 2008]. This supports the findings of Jeon et al. [2005a] who, as explained above, identified duplicate questions based on the similarity of their answers.

Translation models have also been used to recommend *related* questions, rather than duplicate ones [Li and Manandhar, 2011].

The quality of translation probabilities

The quality of the translation probabilities directly influences the performance of question retrieval models that incorporate translation models. Using such models to solve the lexical mismatch problem can only give good results if the learned translation probabilities are of high quality. It is therefore potentially worthwhile to refine them in different ways. Even so, this research topic has not been explored extensively. We are aware of only two very different studies that focus on this. One tries to improve the translation probabilities by eliminating unimportant words from the training data [Lee et al., 2008], while the other aims to enrich the training data by adding glosses from lexical semantic resources [Bernhard and Gurevych, 2009]. Another difference is that

[4] As we will see in §3.3, many researchers disagree with this conclusion and have been able to obtain good results when retrieving answers directly.

the former study applied the translation probabilities to a *question* retrieval task, while the latter tried to retrieve *answers* directly.

In these experiments, multiple aspects have been investigated: the type of training data, the measure applied to determine the importance of words, the threshold set on such measures, and the sources of information used to enrich the training data.

In experiments focused on improving the translation probabilities by eliminating unimportant words, the best results were obtained by using question-answer pairs as training data, instead of duplicate question pairs; by using TF-IDF scores to measure the importance of words, instead of TextRank [Mihalcea and Tarau, 2004]; and by removing a fixed proportion of the vocabulary, ordered from least important to most important, instead of removing all words with a score lower than average, or below a certain threshold [Lee et al., 2008].

In experiments focused on improving the translation probabilities by adding glosses from lexical semantic resources, the best results were obtained by linearly combining three different models: one trained on WikiAnswers questions; one trained on WikiAnswers QA-pairs[5]; and one trained on glosses from Wiktionary[6] and Wikipedia [Bernhard and Gurevych, 2009].[7]

Another aspect of translation probabilities that has been investigated is the length imbalance between translation pairs and the effect of translation direction [Zhou et al., 2013a]. These factors come into play when the training data consists of question-answer pairs, rather than question-question pairs.

IBM Model 1 assumes that translations are of a comparable length to the input, but answers tend to be longer than questions. There is often a considerable difference in length between the two, which may have a negative effect on the performance. The length of answers can be balanced by using only the most important words. This has been shown to improve the performance of translation models [Zhou et al., 2013a].

[5] Both taken from `http://wiki.answers.com`

[6] `http://en/wiktionary.org`

[7] The English Wikipedia (`http://en.wikipedia.org`) and the Simple English Wikipedia (`http://simple.wikipedia.org`).

The direction of the translations can be considered to go either from answers to questions, or from questions to answers. Results show that the answer → question direction is empirically superior. A model that combines the two linearly but gives more weight to the answer → question model obtains the best performance [Zhou et al., 2013a].

3.2.2 Question retrieval: topic models

The lexical gap between duplicate questions can be partially bridged by making use of topic models [Cai et al., 2011, Zhou et al., 2011b, Ji et al., 2012]. When training a topic model, groups of words are clustered into topics based on recurring patterns of co-occurence. The resulting topics capture meaning at a less fine-grained semantic level than translation models or word overlap-based similarity measures. If the clusters are large enough (and the number of topics low enough), it can happen that two questions belong to the same topic even though they do not share (m)any words. In this way, topic models can help alleviate the lexical gap problem.

Because topic models capture meaning at a different level from translation models, the two complement each other.[8] For this reason, [Cai et al., 2011], [Zhou et al., 2011b], and [Ji et al., 2012] all linearly combined a topic model with the translation-based language model (TRLM) [Xue et al., 2008]. Table 3.1 summarises the differences between these three papers.

As can be seen in Table 3.1, the research differs in the data used, the particular topic model, whether regularisation was applied or not, what part of the data the topic model was trained on, and whether the category information was used or not. All three papers report improved performance over the TRLM. Posterior regularisation can be added to improve the results even further [Ji et al., 2012], and adding category information was also found to be helpful [Cai et al., 2011].

The papers in Table 3.1 all trained their topic models over the full dataset. Their test questions therefore have all been assigned a topic

[8] Although it has been noted that topic based models often outperform translation-based models, which can be used as an argument against translation-based methods [Zhang et al., 2014b].

	Cai et al. [2011]	Zhou et al. [2011b]	Ji et al. [2012]
Data:	cQA	discussion forum	cQA
Topic model:	LDA	LDA	QATM
How combined:	linearly	linearly	linearly
Regularisation:	no	no	posterior
Trained on:	questions	questions	questions and answers
Category info:	included	not used	not used

Table 3.1: An overview of the differences between three different methods that all linearly combined a topic model with [Xue et al., 2008]'s translation-based language model. QATM is an adaptation of the PLSA topic model [Hofmann, 1999]; LDA was developed by [Blei et al., 2003]; more information about posterior regularisation can be found in [Ganchev et al., 2010].

distribution as part of the training of the topic model. In a real world setting this is not the case. New questions are asked, which do not have a topic distribution yet. Instead, this distribution will need to be inferred. When the model is trained on questions only, a distribution for new questions can be inferred easily. However, this means that the data in the answers cannot be used, and we disregard potentially useful information.

One way to get around this problem is to train two topic models: one over questions only (Q-model) and one over questions and their answers (QA-model), and learn a mapping between the two. The Q-model can be used to infer a distribution for new questions. Next, the answer data can be leveraged by making use of the mapping to translate the Q-distribution of the new question to a corresponding QA-distribution. This can then be used to find relevant archived questions [Chahuara et al., 2016].

Another novel idea in this work is the use of distributed representations of words instead of concrete ones when training the topic model [Chahuara et al., 2016]. We will see more models that make use of this in §3.2.3 on deep learning, where we will also discuss topic models that have been combined with a neural network [Das et al., 2016a].

Topic models have also been explored for the very similar task of answer retrieval, e.g. [Vasiljevic et al., 2016, Zolaktaf et al., 2011]. This will be discussed in §3.3.

A question retrieval method that is similar to topic models is (non-negative) matrix factorisation. Such models have also been explored for question retrieval and been found to produce better results than both translation-based models and topic models [Zhou et al., 2013b, 2014, 2016a].

Question retrieval: question quality

It is possible to rank archived questions based not only on their relevance, but also on their quality, for instance by training a classifier and combining it with the retrieval model. Doing this improves the retrieval results [Zhang et al., 2014b]. More information on automatically determining post quality can be found in §2.3.

Another method for incorporating post quality involves representing the cQA data as a graph, where each node represents a user, and directed edges are inserted from each asker to the answerers. Using this graph, topical expertise and the topical interests of users can be encoded in a topic model [Yang et al., 2013]. The topic model can be extended even further by using a variant of the PageRank algorithm [Page et al., 1999] that incorporates the information from the topic model. The result is a PageRank algorithm that takes users' expertise in certain topics into account [Yang et al., 2013]. This model can recommend expert users, find answers, and find similar questions to new questions.

3.2.3 Deep learning approaches to question retrieval

In recent years, deep learning approaches have been gaining popularity. Deep learning approaches to question retrieval map questions to a dense vector representation and then compare these vectors to determine the similarity. However, these approaches require large amounts of annotated data to produce good results. Nassif et al. [2016] for instance, trained a model of two stacked bidirectional long short-term memory (LSTM) neural networks [Hochreiter and Schmidhuber, 1997] with a multilayer perceptron output layer on a small dataset ([Nakov et al., 2016]) and did not outperform a simple TF-IDF baseline, although a larger dataset may have produced much better results. Apart from the small dataset, the low results were attributed to the fact that the test data had more out of vocabulary words than the development data, and the fact that the baselines, including TF-IDF, had been computed using external resources.

The need for large volumes of annotated data can be solved by pre-training on separate sets of unannotated data [Hinton et al., 2006, Bengio et al., 2007, Ranzato et al., 2007]. In question retrieval, such unannotated data would be cQA data [Zhou et al., 2015, dos Santos et al., 2015, Lei et al., 2016]. Pre-training ensures that the word vectors are initialised in a way that reflects data distributions. Pre-trained models therefore need less actual training data to obtain good results.

Another way to get around the annotation problem is to train an unsupervised model that maximises the self-similarity of a question, rather than the similarity of a question to an annotated similar question. In this way, each word in a training question needs to be similar to all other words in the same question and its answer(s) [Zhang et al., 2016].

A third way to by-pass the need for large sets of annotated data is to train a model on question-answer pairs instead of question-question pairs. A problem with this however, is that the model learns a relationship between question and answer terms, instead of between terms in similar questions. While this harms the textual similarity detection, a simple solution is to add regular retrieval scores (BM25 [Robertson et al., 1994]) to the model [Das et al., 2016b].

Table 3.2 shows an overview of recent work on question retrieval using deep learning approaches. Only dos Santos et al. [2015] and Das et al. [2016b] evaluated their model in a retrieval setting. All the others used a standard retrieval method (usually BM25) to retrieve a number of candidate questions for each query question, and then applied their neural network to these small sets of candidates to rerank them. This

	Zhou et al. [2015]	dos Santos et al. [2015]	Qiu and Huang [2015]	Zhang et al. [2016]	Lei et al. [2016]	Das et al. [2016a]	Nassif et al. [2016]	Das et al. [2016b]
NN model:	skip-gram + fisher vectors	CNN + BoW	CNN + tensor layer	CBOW + categories	recurrent gated CNN	Topic model + CNN	stacked LSTMs + MLP	siamese CNN + BM25
(Un)supervised:	supervised	supervised	supervised	unsupervised	supervised	supervised	supervised	supervised
BoW vs. sequence:	BoW	BoW	BoW	BoW	sequence	character trigrams	sequence	character trigrams
Test method:	reranking BM25	retrieval and ranking	reranking (VSM)	reranking BM25	reranking BM25	reranking (BM25)	reranking Google results	retrieval and ranking
Category info:	cat info included	cat info not used	cat info not used	cat info included	cat info not used	cat info not used	cat info not used	cat info not used
Training:	pre-training	pre-training	train on qa-pairs	Q self-similarity	pre-training	train on qa-pairs	annotated data	train on qa-pairs
Compared what:	title	title and title + body	title	title + body + answers	title + body	title	title + body	title
Dataset used:	Yahoo! Ans. & Baidu Zhidao	StackExch. Ubuntu & English	Yahoo! Ans. & Baidu Zhidao	Yahoo! Ans. & Baidu Zhidao	StackExch. Ubuntu	Yahoo! Webscope	Nakov et al. [2016]	Yahoo! Webscope

Table 3.2: An overview of the differences between eight different papers that used a neural network in their question retrieval experiments. The rows represent the following aspects: the NN model used, whether the model is supervised or unsupervised, the kind of representation that is used (embedded bag-of-words, sequence, or character n-grams), how the model was tested, whether category information was used or not, how they sourced large volumes of training data, what exactly was compared in the tests, and which datasets were used.

makes training the model more efficient [Lei et al., 2016].

While recurrent neural networks are a natural fit for tasks where sentences of different lengths are compared, simpler bag-of-words models have shown good results too. We will briefly discuss four such papers, which are also listed in Table 3.2: [dos Santos et al., 2015, Zhou et al., 2015, Zhang et al., 2016, Qiu and Huang, 2015], before looking at recurrent neural models.

dos Santos et al. [2015] used a very straightforward approach in which they combined a convolutional neural network (CNN) with a bag-of-words representation. Each question followed two parallel paths through the model: one path computed a weighted bag-of-words representation and the other path computed a distributed representation by means of a CNN. For each query question, these two representations were then compared to the representations of the candidate archived questions, using the cosine similarity metric, and these two cosine similarity scores were linearly combined to produce the final similarity score. The authors concluded that the two path model worked well for long questions, while for short questions they obtained better results when leaving out the bag-of-words path and only using the CNN. Unfortunately the model was not compared against any state of the art methods, only simple baselines.

Qiu and Huang [2015] took a very similar approach, but added a tensor layer at the top of the model, following the Neural Tensor Network (NTN) [Socher et al., 2013], to model the relations between similar questions more accurately than by using the cosine similarity.

Zhou et al. [2015] used Mikolov et al. [2013]'s `word2vec` to learn word embeddings. They then generated fisher vectors [Perronnin and Dance, 2007] to solve the problem of different questions having different lengths. Questions in cQA archives are often divided into categories to enable users to browse questions per domain. As we will see in §3.2.4, making use of this category information is generally a good idea.

Zhou et al. [2015] included `word2vec` features at testing time as a weight on the similarity scores between words in the embedding space. Words that appeared in the same category received a similarity weight of 1, while words that did not appear in the same category received a similarity weight of 0. In other words, two words were only considered to be similar if they appeared in the same category. The final similarity of two questions was calculated by taking the dot product of the fisher vectors of the questions. With this model they outperformed the phrase-based translation model [Zhou et al., 2011a], and several topic models [Ji et al., 2012, Zhang et al., 2014b].

Zhang et al. [2016] also used `word2vec`. Apart from the words in the questions, they used the words in the answers when training their word embeddings. Instead of comparing the vectors of questions directly, they used the embedded word similarities (calculated using the cosine similarity metric) to replace the translation probabilities in the translation-based language model (TRLM) [Xue et al., 2008]. The similarity score was converted into a probability using the softmax function. The score of this new version of the TRLM was then linearly combined with a similarity score based on the category of the two questions being compared. Like Zhou et al. [2015]'s model above, this model outperforms several topic models [Ji et al., 2012, Zhang et al., 2014b], and also a language model using category information [Cao et al., 2009].

Deep learning for question retrieval: using character n-grams

Another approach to represent questions without sequence information is to use character n-grams instead of words. This is a good way to reduce the dimensionality of the vector representations and at the same time alleviates the out-of-vocabulary (OOV) problem that word-level models encounter [Das et al., 2016b,a].

A Siamese convolutional neural network (CNN) ([Bromley et al., 1993]) using character trigrams combined with BM25 [Robertson et al., 1994] retrieval scores, outperformed the phrase-based translation model [Zhou et al., 2011a], several topic models [Ji et al., 2012, Zhang et al., 2014b], and a regular CNN model [LeCun et al., 1998, Das et al., 2016b]. The difference between this model and the other neural models we have discussed so far, is that in a Siamese model the parameters of the two CNNs (of the two input sentences/questions) are shared.

A model that combined a topic model with a CNN model using character trigrams also outperformed the phrase-based translation model [Zhou et al., 2011a] and several topic models [Ji et al., 2012, Zhang et al., 2014b, Das et al., 2016a]. These two models cannot be compared directly however, because one of them was used in a full retrieval setting, while the other was only used in a reranking setting.

Deep learning for question retrieval: recurrent models

As mentioned before, recurrent models are a natural fit for comparing sentences of different lengths, but not many researchers have used them in question retrieval yet. In recurrent neural networks, each question is treated as a sequence of words. This is very different from the bag-of-words approaches above, in which sequence information is only taken from the immediate context of words. Recurrent models are particularly good for learning long distance dependencies between words.

The best results to date have been obtained using a recurrent and gated convolutional model [Lei et al., 2016].[9] This outperforms several related models (LSTM [Hochreiter and Schmidhuber, 1997] in the work of Nassif et al. [2016], GRU [Cho et al., 2014, Chung et al., 2014], and CNN [LeCun et al., 1998]), but only when pre-training is applied. Unfortunately, the model is not compared to any state of the art question retrieval models that do *not* make use of deep learning.

In SemEval 2016 Task 3 Subtask B on cQA question-question similarity,[10] several systems were submitted that made use of deep learning approaches. For instance, Hsu et al. [2016] used a recurrent neural network and extended it with an attention mechanism to better handle long distance relationships. More information on the task and the participating systems can be found in Nakov et al. [2016].[11]

[9] The model is inspired by the work of LeCun et al. [1998] and Lei et al. [2015].

[10] `http://alt.qcri.org/semeval2016/task3/`

[11] The same dataset was used for the ECML/PKDD 2016 Discovery Challenge: `http://alt.qcri.org/ecml2016/`.

3.2.4 Question retrieval: using category information

One type of meta data that many researchers have made use of in their models is *category information*. Many cQA forums organise the questions people ask into different categories. Often these are pre-defined, but on Quora for instance, users can create as many categories as they like. On StackExchange, the questions are not only split up into different categories, but these categories each have completely separate forums. Category information can be leveraged best when the set is fixed and there are a large number of categories, like on Yahoo! Answers. Yahoo also organises its categories into a hierarchy.

When the set of categories is fine-grained, like on Quora for instance, they are essentially the same as tags, which are often used in social media to group items of similar content together. Some forums make a clear distinction between the two however. StackExchange for instance, has high-level categories (e.g. Physics, Linguistics, Bitcoin, Poker) and questions within those categories can receive tags to specify their content further (e.g. quantum-mechanics, notation, optics, mathematical-physics for the Physics category[12]).

Category information can be used to limit the number of archived questions to search through [Cao et al., 2012, Zhou et al., 2013a], or to improve the relevance scores of retrieved questions [Cao et al., 2009, 2010, Chan et al., 2014]. The general consensus is that adding category information to the retrieval model in some way will boost the performance.

Category similarity

In Yahoo! Answers some question categories are similar, and so questions relevant to a particular query should be searched for in multiple categories. This way the retrieval results could improve over searching in one category only, while at the same time being more efficient than searching for relevant questions in all categories [Cao et al., 2012, Zhou et al., 2013a].

One way to make use of this idea is to use a classifier to estimate

[12] The categories are called *sites* on StackExchange.

the relevance of each category to the query question. This information can then be used in two ways: to prune the search space by only considering questions from a category with a probability higher than a given threshold, and later to rerank the returned archived questions, by giving more weight to questions from a category with a high probability [Cao et al., 2012].

Alternatively, the category information can be used for pruning only, and to find related categories by training a topic model over them. Related questions can then be searched for only in the category of the query question, and categories that are topically similar to it [Zhou et al., 2013a].

Both of these methods can be added to any retrieval model to improve the performance and the efficiency. The best results are reported for the category-enhanced TRLM [Xue et al., 2008], with running time improvements above 85% over the same model without category information [Cao et al., 2012, Zhou et al., 2013a].[13] Category similarity can also be exploited by linearly combining it with the question similarity scores [Chan et al., 2014].

Within-category word importance

Another idea that has been investigated is that category-specific frequent words are not informative when comparing questions within that category, but carry much more importance when comparing a query question to archived questions from a *different* category [Cao et al., 2009, 2010, 2012, Ming et al., 2010].

This notion can be incorporated into a retrieval model by weighting terms depending on their category-specificity [Ming et al., 2010], or by using a two-level smoothing setup: a category model smoothed with the whole question collection, and a question model smoothed with the category model [Cao et al., 2009]. This setup outperforms the translation model [Jeon et al., 2005b] and can be improved even further (both in terms of effectiveness and efficiency) by adding query classification

[13] Cao et al. [2012] only tested the efficiency difference on the language model, but because computation for the TRLM is more expensive, improvements are likely to be even greater for the TRLM than for the language model.

to limit the number of potential duplicates, assuming a question and its duplicate will have the same category [Cao et al., 2009].

A better way of incorporating this idea is by means of the notions of *global relevance* and *local relevance* [Cao et al., 2010]. The local relevance is a basic relevance score. It is computed between a query question and an archived question. The global relevance is computed between a query question and the *category* of an archived question. Categories are represented by the words that appear in all the archived questions associated with that category. Different models can be used to compute the global and local relevance scores. Experimental results show that the best results are obtained by using a vector space model for the global relevance and a translation-based language model [Xue et al., 2008] for the local relevance [Cao et al., 2010].

Although more complex retrieval models have been developed to address the *lexical gap* issue, traditional models have proven to still be useful, especially when enhanced with category information. For instance, a vector space model applied to questions enriched with synonyms, associative relations and category information constructed from Wikipedia concepts outperformed many sophisticated retrieval strategies like the phrase-based translation model [Zhou et al., 2011a], the translation-based language model [Xue et al., 2008], and a matrix factorisation model [Zhou et al., 2013b,c]. This is an interesting result, because it shows that improving the representation of questions can outperform better retrieval models.

Category information has also been used in deep learning models [Zhou et al., 2015, Zhang et al., 2016] and non-negative matrix factorisation models [Zhou et al., 2014] too. This work is discussed in §3.2.2 and §3.2.3.

3.2.5 Other question retrieval methods

Question retrieval is an active field of research. In the previous sections we have touched upon three large streams within the field: translation models, topic models and deep learning approaches. However, many studies have been conducted that do not fit into those streams. For instance, a model has been proposed in which questions are translated into a different language and the translations are added to the bag-of-words representation of the original question, thus potentially creating more lexical overlap between semantically similar questions [Zhou et al., 2012b]. This is a form of query expansion.

Information can be derived from the internal structure of questions. Some work has looked at using question patterns to find related questions [Hao and Agichtein, 2012a,b]. The success of this method is limited. Others have used question structure to determine their topic and focus, and incorporated this information into a language model [Duan et al., 2008]. Here, topic and focus are linguistic terms. *Topic* refers to what is is talked about (but is not derived using probabilistic topic modelling methods). This is often the subject of a sentence, but not always. *Focus* refers to what is said about the topic [Gundel and Fretheim, 2004].

Questions can also be represented as graphs, capturing either the syntactic structure [Wang et al., 2009d] or dependency structure [Zhang et al., 2012]. Syntactic graphs can be used to calculate how many tree fragments two questions have in common in order to determine how similar they are [Wang et al., 2009d]. To obtain good performance with such a model, it needs to be extended with a lexical similarity method [Wang et al., 2009d]. Dependency graphs can be used to estimate the closeness of query question terms and to adjust their weights accordingly [Zhang et al., 2012].

One downside to using syntactic or dependency graphs for question retrieval is that such methods tend to work better for short questions, while some cQA archives typically have quite long questions (e.g. StackExchange). For such models, the retrieval can be improved by segmenting multi-sentence questions and removing the context sentences [Wang et al., 2010c,b].

Syntactic information and dependency relations can also be used in a learn-to-rank framework. In such a setup, syntactic features have been shown to be more effective than dependency features [Carmel et al., 2014], although it should be noted that queries in this research were web queries, not questions. Syntactic features can be complemented by part-of-speech features [Carmel et al., 2014].

Other features investigated for learn-to-rank models include the matching of main verbs between two questions, the inclusion and proximity of locations in the questions, the matching of focus words, and the cosine similarity. These last two were found to be particularly helpful [Bunescu and Huang, 2010a,b].

External resources, like Wikipedia concepts, can be used to find lexical relations between words. This information can be added to a model, for instance by linearly interpolating a question similarity score with a Wikipedia concept similarity score [Zhou et al., 2013c], or it can be used as a feature in a classifier (e.g. WordNet similarity) [Ahasanuzzaman et al., 2016].

Finally, ranking models can be improved by taking certain aspects of a question into account, like *utility* [Song et al., 2008], a subjective notion of a question's usefulness, objectively defined as the likelihood that a question is asked by multiple users, or subjectivity [Gurevych et al., 2009], which we discuss in §2.4.

Other methods that have shown promising results are multidimensional scaling [Borg and Groenen, 2005, Xiang et al., 2016], using tree kernels [Da San Martino et al., 2016] (which has also produced good results for answer retrieval. See §3.3), and representing questions and answers as a quadripartite graph with nodes consisting of askers, answerers, questions, and answers, and using a hierarchical clustering algorithm to identify similar questions [John et al., 2016]. Some research has looked at cross-lingual question retrieval, in which the question is written in one language, while the retrieved results are written in another one [Chen et al., 2016a].

Until now we have looked at full questions as queries, but in many situations users find cQA answers via a search engine. Queries posted to a search engine are often short, instead of full sentences. Researchers have worked on automatically generating questions from web search queries, by learning from search engine logs [Zhao et al., 2011]. Such queries can be analysed to improve the cQA search results, for instance by using dependency parsing or dividing the query into syntactic units [Pinter et al., 2016], by classifying queries into different categories [Figueroa and Neumann, 2016], or by weighting unmatched query terms by “mirroring” features of matched terms in similar features of unmatched terms [Petersil et al., 2016]. They can also be expanded, in particular to allow for exploratory search within the cQA results [Gao et al., 2016].

The related task of finding questions that are related to the query question in some way, to allow users to explore additional or alternative aspects to their problem, has received little interest from the research community. Some experiments have been conducted using translation models [Li and Manandhar, 2011], and graph or tree matching models [Cong et al., 2008].

Automatically grouping questions together based on topical similarity is another related task. This is similar to automatic tagging of new questions, and can be used for organising the content of a forum automatically. Experiments in this space include clustering questions [Deepak, 2016], and classification approaches [Qu et al., 2012]. In classification, simple bag-of-word features have been shown to perform better than n-grams, features extracted from question titles are more informative than those extracted from the body, the best answer, or the asker, and hierarchical classification gives better results than flat classification [Qu et al., 2012].

3.3 CQA answer retrieval

In this section we will discuss the second type of post retrieval from cQA archives: answer retrieval. This includes both retrieval of answers from the complete archive, and retrieval or ranking of answers within one thread. Ranking answers within one thread according to their relevance to the query is highly related to ranking answers according to their quality, because it is often the low quality answers that are regarded as not very relevant. Post quality assessment, which includes best answer prediction and ranking answers based on their quality is discussed in §2.3.

In answer retrieval from the complete archive many strategies have been explored that have also been used for question retrieval. For instance, a monolingual translation model has been applied, to learn a

Someone who forms their own opinion
Is there a single word for someone who forms their own opinion based entirely on their personal experience, without having been influenced by any outside source?

Answer:
I'd probably use: free-thinker.

Figure 3.4: An example of a question and an answer that have no lexical overlap. Source: `http://english.stackexchange.com`

translation from questions to answers [Bernhard and Gurevych, 2009]. Although most models can be used both for question retrieval and answer retrieval, very little research has looked at which model gives better results for a particular task.

Is it generally assumed that the *lexical gap* problem, which we introduced in §3.2, is even greater between a question and its answer, than between two duplicate questions [Shtok et al., 2012]. On the other hand, there is much more training data available for question-answer pairs, than for duplicate question pairs.

Another thing to note is that the lexical gap between question-question pairs and between question-answer pairs may be different. In question-question pairs the words in the questions are assumed to be semantically *similar* if the questions are duplicates, while in question-answer pairs the words may only be semantically *related*. An example of this can be found in Figure 3.4, in which the question contains the terms *opinion*, *without*, and *influenced*, which are all related to the term *free-thinker* in the answer, without being semantically similar. This difference in the type of lexical gap may influence how well certain techniques work for question retrieval and answer retrieval, respectively.

3.3.1 Answer retrieval: topic models

Topic models, which have been investigated extensively for question retrieval (see §3.2.2), have seen relatively less use for answer retrieval. In one study, a topic model was trained on a set of answers and then used to infer a topic distribution for new questions [Vasiljevic et al., 2016]. The inferred topic distribution was compared to the distributions of the archived answers to find the most similar one(s). The model performed slightly better than a tf-idf baseline, but even so, the study showed that using only a topic model is not enough to capture the level of semantic similarity needed to determine if an answer is truly relevant [Vasiljevic et al., 2016]. As with question retrieval (see §3.2), the best results can be expected in combination with other approaches like the TRLM [Xue et al., 2008], or by adding category information to it.

Most researchers include all questions when training the topic model, because that way they can get good topic distributions for their query questions. In a real world setting, however, this is not possible: the topic distributions of new questions need to be inferred *after* training the model. Vasiljevic et al. [2016] and Chahuara et al. [2016] are the only researchers that recognise this and do not include query questions during training.

It has been argued that topic models are insufficient for answer retrieval because askers and answerers have a different level of knowledge, which shows in their posts [Zolaktaf et al., 2011]. Answerers will use more technical terms than askers, and may introduce related concepts as well. Because of this, the topic distribution of questions and answers is different, and a regular topic model like LDA [Blei et al., 2003] will not capture the distinction. To remedy this problem, questions and answers can be modelled separately, while still depending on each other, because topics in answers are influenced by the topics in the corresponding question. This can be achieved by conditioning the answer topics on the topics in the corresponding questions [Zolaktaf et al., 2011].

3.3.2 Answer retrieval: incorporating answer quality

Answer retrieval can be improved by incorporating the quality of the answers [Jeon et al., 2006, Bian et al., 2008a, Suryanto et al., 2009, Omari et al., 2016, Zhang et al., 2014b]. Different ways to determine answer quality are discussed in §2.3, with common features being the answer content, grammaticality or readability of the answer. User expertise (see §5.3) can also be added to improve answer retrieval [Suryanto et al., 2009].

A learn-to-rank model with features that capture the answer quality has been shown to outperform two simple baselines: chronologically ordered answers and answers ordered by the number of community votes, in decreasing order [Bian et al., 2008a]. The second baseline may seem surprising, but it has been shown that good answers are sometimes not among the best answers as voted for by the community, especially in the context of Yahoo! Answers [Suryanto et al., 2009, Jeon et al., 2006].

Apart from encoding it in features for a learn to rank model, quality information can be added as a prior of the query likelihood model [Ponte and Croft, 1998, Ponte, 1998, Jeon et al., 2006], or it can be incorporated in a mixture model of a query likelihood model and a topic model [Zhang et al., 2014b].[14]

It has been argued that user votes become less trustworthy as a quality measure of content when a forum gains popularity [Bian et al., 2008b]. Retrieval models that make use of these votes should therefore be robust enough to be able to deal with poor quality votes, or *vote spam*. A learn to rank model with textual features and user interaction features can be effectively trained to learn to recognise different kinds of vote spam in cQA archives [Bian et al., 2008b].

In some cQA archives, for example Quora,[15] users can follow other users. This means that they will get to see all the content the other user posts, all the questions they follow, and all the answers they vote for. Because of this, people are more likely to receive upvotes on their answers from followers than from other people, and the more followers someone has, the more votes they will receive. Answer ranking can be improved by taking this voting bias into account in two ways: by working with percentages instead of raw numbers; and by making a distinction between votes from followers and votes from other people [Geerthik et al., 2016].

In all the answer retrieval research incorporating answer quality

[14] Zhang et al. [2014b] used their method for both answer ranking and question ranking. More details on this work and other research on question retrieval that incorporates quality information can be found in §3.2.

[15] `https://www.quora.com/`

which we have discussed so far, the quality of each answer was determined separately from the other answers. Some aspects of good answers however, can only be determined by looking at the other answers to a given question. *Novelty* for instance — the number of new answer components that are introduced in an answer compared to the other answers — is a good example of this. In web search, novelty and diversification have long been recognised as important when presenting results [Agrawal et al., 2009, Clarke et al., 2011, 2008, Rafiei et al., 2010, Vallet and Castells, 2012], but in answer ranking few researchers have taken them into consideration.

In one model, answers are segmented into propositions, which are grouped together based on their semantic similarity. A group of propositions represents one aspect or idea. For each of these ideas, its importance is determined by looking at how often it appears in different answers. Answers that contain many propositions (diverse answers), including both common ones (important ones) and uncommon ones (novel ones), are ranked high in the result list. A hierarchical clustering method is applied to obtain the ranking [Omari et al., 2016].

Another way of looking at the within-thread answer ranking problem is by predicting or estimating the rating an answer will receive from the community, rather than ranking answers based on their relevance (and possibly their quality) [Dalip et al., 2013]. Because this is highly related to answer quality, we will discuss this method in §2.3. We mention it here because it produces a ranking of the answers.

3.3.3 Answer retrieval: adding user information

Community question-answering sites are open to anyone, and most of them do not have an active moderation system. Because of this, many answers are provided by people who are by no means experts. In fact, about 20% of the users contribute nearly half of the best answers on Yahoo! Answers [Zhou et al., 2012e]. It has also been shown that user information can be used to help predict the quality of answers [Lui and Baldwin, 2009, Yang et al., 2011, Agichtein et al., 2008, Burel et al., 2012, Shah, 2015].[16] This provides motivation for making use of user

[16] More information on this can be found in §2.3.

information when ranking answers. If an answer is written by a user that has many answers voted as being the best, there is a reasonable chance that this new answer is also good, and should be placed high in the answer ranking of a particular thread.

This idea was explored by looking at three different user related aspects: engagement, authority, and level [Zhou et al., 2012e]. The user features were added to an existing learn-to-rank model for answer retrieval which used similarity features, translation features, density or frequency features (e.g. the number of question terms matched in a single sentence in the answer) and web correlation features (i.e. features that measure the correlation between qa-pairs and large external collections) [Surdeanu et al., 2011]. Authority and engagement related features were found to be useful, while level related features were not. The existence of a profile picture was found to be one of the most informative features. However, the authors did not obtain large improvements over a baseline system, and more research is needed to find out how exactly user information can best be used in answer retrieval [Zhou et al., 2012e].

Instead of using user information as features, it can be leveraged in answer retrieval by co-ranking questions, answers and users. These three are highly related: we already know that good questions attract good answers [Jeon et al., 2006, Yao et al., 2013, Agichtein et al., 2008], and that knowledgeable users provide good answers [Lui and Baldwin, 2009, Yang et al., 2011, Agichtein et al., 2008, Burel et al., 2012, Agichtein et al., 2008, Shah, 2015]. A co-ranking model can be used to exploit the interrelationship between questions, answers and users [Zhang et al., 2014a].

3.3.4 Machine learning approaches to answer retrieval

While this section is about answer *retrieval*, we now move on to discuss several machine learning approaches that have been used to classify answers, and rank them according to their classification scores.

In experiments with a ranking Perceptron over *how*-questions from Yahoo! Answers, Shen and Joshi [2005] experimented with the following four types of features: question-answer similarity features, question-answer transformation features (e.g. translation probabilities), keyword density and frequency, and features that capture the correlation between the question-answer pair and other collections. Of these, translation features were found to be the most helpful, but an additional increase in performance was achieved by adding frequency/density features and correlation features. Furthermore, both the semantic and syntactic features from the question-answer similarity feature group contribute to increase the results even further [Surdeanu et al., 2008].

The same Yahoo! Answers dataset was used by several other researchers, who showed that the answer discourse structure can complement lexical similarity features [Jansen et al., 2014],[17] that higher order lexical models can leverage indirect evidence [Fried et al., 2015], and that inexpensive and cross-domain alignment models can be trained on discourse structures [Sharp et al., 2015]. The dataset has also been used in deep learning experiments [Bogdanova and Foster, 2016], which we discuss in the next section, but the best performance is still held by Surdeanu et al. [2008].

Answer posts can also be ranked using an unsupervised graph-based propagation model [Cong et al., 2008]. In this model, for each question, an answer graph is built using a language model to determine which answers to place an edge between. Edges are weighted based on a linear interpolation of the language model score, the distance of the destination answer post from the question, and the authority score of the author of the destination answer post. The graph is then used to propagate the initial ranking scores from an SVM [Cong et al., 2008].

Instead of using one graph for all threads, the data can be represented as one fully-connected graph per thread, in which the answers are the nodes. Each node and edge is represented by a vector of features, and the goal is to predict the label of each node as either GOOD or BAD based on its relevance to the question, for instance by using a collective classification model [Shafiq Joty et al., 2016].

[17] Yahoo! Answers is a cQA archive with very little moderation. Because of this, the threads of answers sometimes resemble discussion forum threads, and information useful for answer ranking can be derived from this structure.

Deep learning for answer retrieval

In recent years, deep learning approaches have been gaining popularity for answer retrieval. In document retrieval, the difference in length between the queries and the documents to be retrieved is a challenge for deep learning models. In community question-answering, this length difference is generally much smaller. This makes answer retrieval a suitable candidate for deep learning experiments. A similar point can be made for question retrieval. Recall that we discussed deep learning models for question retrieval in §3.2.3.

All published deep learning work on answer retrieval so far has focused on within-thread answer ranking. While the length difference between questions and answers is smaller than queries and documents, it is still there, and needs to be addressed when designing deep models for answer ranking. Several strategies have been applied, including creating a fixed size matrix to capture the co-occurrence of words in questions and answers [Shen et al., 2015a,b], with the downside that only questions and answers shorter than the matrix size could be processed. A deep convolutional network can beapplied to the matrix, to which a k-max pooling layer can be added to obtain vectors of equal length [Tymoshenko et al., 2016].

Another way to solve the length difference problem is by representing questions and answers as binary vectors of only the most frequent words in the training data [Wang et al., 2010a]. In this work, the researchers used a network of three restricted Bolzmann machines (RBM) [Hinton, 2002]. During training, question vectors were reconstructed from answer vectors, via the RBM layers. During testing, question vectors were compared against answer vectors, to determine their relevance.

Denoising Auto-Encoders (DAE) [Vincent et al., 2008] can be used to learn how to map both questions and answers to low-dimensional representations of fixed size [Zhou et al., 2016b]. In experiments with this setup, the two representations were compared using cosine similarity, and the resulting score was used as a feature in a learn-to-rank setup, together with a set of statistical-based features [Zhou et al., 2016b].

Document-level representation learning methods such as doc2vec [Le and Mikolov, 2014] can also be applied to the task, to generate a fixed length distributed representation of each question and answer. This is another way to solve the length difference problem [Bogdanova and Foster, 2016, Lau and Baldwin, 2016]. A simple feed forward neural network can then be used for classification.

By transforming the input to a layer of fixed length, we lose information. The idea is that we lose only irrelevant information, but for answers this may not be the case. They are typically much longer than questions, which means that we lose more (potentially useful) information in the transformation.

One solution for this problem is to allow questions and answers to be mapped to a lower dimensional layer of variable size, and to use a 3-way tensor transformation to combine the layers and produce one output layer [Bao and Wu, 2016].

Recurrent neural networks (RNNs) are a natural fit for input texts of different length. The current state of the art RNN is the Long Short-Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997], which was designed to handle long distance dependencies. Nassif et al. [2016] used stacked bidirectional LSTMs with a multilayer perceptron on top, and the addition of a number of extra features, to classify and rerank answers. The model was also used for question retrieval (see §3.2). Although the model produced good results for the answer reranking task, a classification model using only text and vector based features [Belinkov et al., 2015] produced similar results, while being much simpler.

Outside of cQA, in regular question answering, other deep models have been applied successfully in answer extraction and answer sentence selection: convolutional neural networks (CNNs) [Feng et al., 2015, Severyn and Moschitti, 2015], long short-term memories (LSTMs) [Wang and Nyberg, 2015a], or a combination of these [Tan et al., 2016]. More exploration of these models for cQA can be expected.

3.3.5 Other answer retrieval methods

One approach to bridge the lexical gap that has been investigated for (within-thread) answer retrieval, but is unsuitable for question retrieval, is the application of *analogical reasoning.* In this approach, questions and their answers are regarded as being connected via semantic links in a graph, which can be either positive or negative, depending on the quality of the answer and its relevance to the question [Tu et al., 2009, Wang et al., 2009e].

In experiments, for each query question, similar questions were retrieved using a standard retrieval technique. Next, the links between these related questions and their high quality answers were modelled using a Bayesian network. The question-answer links of the query question were compared to the retrieved ones, to find analogous links. The more analogous a link, the higher the answer should be ranked [Tu et al., 2009, Wang et al., 2009e].

The downside to this approach is that answers to questions can only be ranked if there are similar questions in the archive. Some cQA archives contain many duplicate questions (like Yahoo! Answers), but even so, the majority of the questions in the archive do not have any similar ones. Other cQA archives (like StackExchange) have a mechanism to link repeated questions to archived ones. After that, no answers can be posted to the new question, only to the archived one. In such a setup, the analogical reasoning approach cannot be applied.

Lee and Cheah [2015] developed a semantic relatedness measure for answer retrieval, based on the analysis of the depth of terms in WordNet.[18] Their results are slightly lower than can be obtained using machine learning methods, with the advantage of not relying on training data.

The type of a question (e.g. yes/no-question, 5W1H-question, etc.) can provide clues for the type of answer to expect. In turn, question classification can be used to improve answer retrieval [Pechsiri and Piriyakul, 2016], although not much work has been done on this yet. Some more information on this can be found in §2.2.

[18] `https://wordnet.princeton.edu/`

Some interesting work has also been done on using question retrieval methods to improve answer retrieval and vice versa, by mutually promoting the two tasks based on the results of each other [Lan et al., 2016]. Both question retrieval and answer retrieval benefit from such a setup.

A final piece of work relevant in the context of answer retrieval is that of Sondhi and Zhai [2014]. They tried to answer questions posted to cQA archives by making use of external online semi-structured knowledge bases as a source of answers. The work focuses on how to transform cQA questions into SQL queries. Outside of cQA, there is a large volume of literature on translating natural language sentences into database queries, known as NLIDB research [Popescu et al., 2003, Katz et al., 2002, Bernstein and Kaufmann, 2006, Lopez et al., 2005].

3.3.6 Shared tasks on answer retrieval

(Moved Section)

At SemEval 2015 a new shared task on answer selection over cQA data was introduced [Nakov et al., 2015],[19] and continued in a slightly different form in 2016 [Nakov et al., 2016][20] and 2017 [Nakov et al., 2017].[21] The answer ranking task was framed as a classification task with three classes: GOOD, POTENTIAL, and BAD.

Datasets were provided for each of Arabic and English, and separate system rankings were produced for the two languages. For the 2015 ranking, the systems that produced the best results on the Arabic set were not the same systems that produced the best results on the English set. For Arabic, a system using lexical similarity features only outputformed systems that used semantic similarity features as well [Belinkov et al., 2015], while for English, a system that combined topic models with word vector representations and semantic similarity features produced the best results [Tran et al., 2015]. More information on

[19] `http://alt.qcri.org/semeval2015/task3/`

[20] `http://alt.qcri.org/semeval2016/task3/`. Answer ranking for Arabic was retained, while English answer ranking was supplanted by English comment ranking.

[21] `http://alt.qcri.org/semeval2017/task3/`. The answer ranking tasks are the same as in 2016.

the winning systems and a comparison of the different approaches can be found in Nakov et al. [2015] and in the 2015 workshop proceedings.[22]

In 2016, the best scoring systems made use of tree kernels combined with similarity features (e.g. Filice et al. [2016]). This idea was later refined by Tymoshenko et al. [2016], who substituted the shallow parse tree for a constituency tree and added authorship information.

Most teams made use of SVMs or neural networks for the classification. The neural networks often did not outperform the more traditional classification models. More details on the participating systems can be found in Nakov et al. [2016] and the 2016 workshop proceedings.[23] The task also included two subtasks on comment ranking, a related challenge.

Another question-answering task that made use of cQA data was introduced in 2015 and repeated in 2016: TREC LiveQA.[24] In this task, participants were given questions from Yahoo! Answers, which their system needed to answer within a minute. While the questions came from a cQA archive, no restrictions were placed on the source of the answers. Even so, many participating systems sourced them from the Yahoo! Answers archive.

An interesting difference between the top scoring systems is that some of them turned the input question into a web search query by selecting the most informative words, thereby making the query shorter than the original question [Wang and Nyberg, 2015b, Nie et al., 2015], while others did the opposite and instead expanded the original question with synonyms and hypernyms, thereby making the query longer than the original question [Wu and Lan, 2015]. For the retrieval of candidate answers, all systems used existing search engines, and then extracted passages from these which were then re-ranked. A comparison of the participating teams and their results can be found in Agichtein et al. [2015].

[22] `http://alt.qcri.org/semeval2015/cdrom/index.html`

[23] `https://aclweb.org/anthology/S/S16/S16-1000.pdf`

[24] `https://sites.google.com/site/trecliveqa2016/`

3.4 Post retrieval evaluation

In §1.4 we briefly introduced some widely used evaluation metrics in IR research. When applied to forums however, there are some forum specific problems that need to be addressed. For instance, one problem with post retrieval evaluation that is often ignored but which occurs often in forum settings is the following: queries for which there are no relevant results. For such queries, the correct result is an empty list, which should be counted as the ideal response in the evaluation, but most IR evaluation metrics either count it as wrong, or the result is undefined in this case.

The problem is usually circumvented by only using queries that do have relevant results in the index, or by evaluating empty result queries differently from the other queries. However, neither scenario is ideal.

One metric that has been proposed to solve the evaluation of empty result queries is c@1 [Peñas and Rodrigo, 2011]. In this metric, c stands for *correctness*. It is assumed that some of the empty result lists are desired and thus correct, but we do not know how many. The accuracy of the non-empty result queries is therefore taken as an estimate of the accuracy of the empty result queries. However, c@1 can only be applied in situations where each query has at most one correct answer.

A related topic to zero result questions is *result list truncation*. In post retrieval, a truncated result list is usually a more desired result than an exhaustive ranked list. By truncating the result list, we can show users only the relevant results and nothing below it. This is another scenario which IR evaluation metrics for post retrieval should take into account. Handling truncated lists implies also handling empty result lists, because truncating result lists can lead to some queries ending up with an empty result list.

A recently proposed strategy to handle truncated result lists, and empty ones, is to insert a dummy result ("NIL") at the point where the list needs to be truncated, and calculate the gain for that result differently depending on how many relevant results there are in the index [Liu et al., 2016]: see Equation 3.8, where r_t is the gain of the terminal document (the NIL result), d is the number of documents in the returned list, and R is the total number of relevant documents in the index.

$$r_t = \begin{cases} 1 & \text{if } R = 0 \\ \sum_{i=1}^{d} r_i / R & \text{if } R > 0 \end{cases} \tag{3.8}$$

This idea can be incorporated in existing retrieval metrics, like MRR, MAP, nDCG [Järvelin and Kekäläinen, 2002], or RBP [Moffat and Zobel, 2008], simply by adding one extra result to the list (the NIL) and by applying Equation 3.8 to calculate the gain [Liu et al., 2016], although for some metrics (e.g. MAP) this might not be a good idea because the scores will be dominated by the NIL result queries.

3.5 Post retrieval summary

In this chapter we discussed forum post retrieval, which can be subdivided into cQA question retrieval, cQA answer retrieval, and discussion forum post retrieval. We looked at a large number of different approaches applied to post retrieval, including language model-based approaches, translation model-based approaches, topic models, and deep learning models, and we discussed some open problems with the evaluation of post retrieval models. In the next chapter we move away from separate posts and instead focus on complete threads.

4

Thread level tasks

A thread is a unit that consists of an initial post, and all other posts that it sparks. In cQA archives this is the question post, all the answer posts, and all the comments (if the answer/comment distinction exists in the particular archive). In discussion forums, it is one stream of posts associated with an initial post. This stream can span multiple pages.

In this section, we discuss retrieval and classification tasks at the thread level.

4.1 Task orientation and solvedness

Classifying threads in different ways can help to improve thread retrieval, just as it did for posts (see §2.1). Less research has been done on thread classification than on post classification however, with the main focus at the thread level being on the goal of the thread, and whether that goal has been achieved or not. Two examples of this are: task orientation and solvedness.

Task orientation is about determining the coarse-grained intent of the thread. One example of this is whether the thread is a question-answer thread, or a discussion thread. This task only makes sense for forums in the middle of the cQA-discussion forum spectrum, which receive both discussion questions and more cQA-type questions. It is very similar to subjectivity detection (see §2.4).

Automatically detecting the task orientation of forum threads has not received much attention from the research community yet. It is, however, an important task, that can improve thread retrieval by allowing a model to ignore either discussion threads or question-answer threads, depending on the query. The task is very challenging; it is difficult to outperform a majority class baseline [Baldwin et al., 2007].

A second example of task orientation prediction is, in the context of discussion forums associated with massive online open courses (“MOOCs”), whether a given thread is a general discussion, is targeted at organising a student group meetup, or relates specifically to a component of the MOOC such as the lectures or assignments. For example, Rossi and Gnawali [2014] proposed a language-independent supervised classification approach to the problem, and found that metadata features such as popularity and numbers of views are the most predictive of task orientation.

Solvedness is about whether the information need of the question asker has been met or not. It is highly related to post quality prediction (see §2.3), and also to completeness (see §2.3.2). It has received somewhat more attention in the literature. A question is solved if it has received a satisfactory answer. In most CQA archives there is a system in place to indicate this. Users that ask a question can usually choose one answer as the correct, or best, answer. The thread will usually be automatically closed when this happens. In discussion forums such a mechanism does not exist, because for many threads it is not relevant. Discussion threads do not have one ‘best’ answer, because a specific answer is not sought.

Researchers have experimented with thread discourse structure features [Wang et al., 2012], and with lexical and contextual features from four subparts of the thread: the initial post, the first response, the last post from the thread initiator, and the full set of responses. A combination of all subparts gave the best results [Baldwin et al., 2007]. Like task orientation, solvedness is a difficult task. Adding discourse features helps, and simulations suggest that improving the thread discourse structure parsing will also improve the solvedness classification [Wang et al., 2012].

Features taken from the asker (e.g. asker reputation) have been found to be more predictive than activity and QA quality features (e.g. number of page views, number of votes), community process features (e.g. average answerer reputation), and temporal process features (e.g. average time between answers) [Anderson et al., 2013]. The same study also looked at predicting the long-lasting value of QA-pairs [Anderson et al., 2013].

4.2 Thread discourse structure

One defining aspect of discussion forum threads is that they have a discourse structure. This is in contrast to cQA threads, where the only discourse is between question-answer pairs, rather than all the answers in the thread. Using information from the thread's discourse structure can help improve many thread level tasks, like thread retrieval [Wang et al., 2013b, Bhatia et al., 2016] (see §4.3), solvedness detection [Wang et al., 2012] (see §4.1), and thread summarisation [Klaas, 2005, Farrell et al., 2001] (see §4.5), and also post level tasks, like post retrieval [Duan and Zhai, 2011] (see §3.1).

In the next sections we will look at methods to recover the thread linking structure (§4.2.1), dialogue act tagging (§4.2.2), and how to partition threads at a point where topic shift occurs (§4.2.3).

4.2.1 Thread linking structure recovery

Discussion forum threads can be presented as tree diagrams, with branches between posts and their answer posts. This information, capturing which posts are a reaction to which older post, is the thread linking structure. Some forums make this explicit (an example can be found in Figure 1.3); others do not (for an example, see Figure 1.4).

Knowing the structure of a thread can help in tasks such as dialogue act tagging (see §4.2.2), or can be used to extract ⟨thread-title, reply⟩ pairs to be used in a chatbot [Huang et al., 2007]. When the structure is not explicit, we can automatically derive it by using a discourse parser that is trained on annotated data, to produce a representation of the thread discourse structure in the form of a rooted directed acyclic graph (DAG) [Wang et al., 2013b].

This problem has also been treated as a ranking task, with child posts as queries, and older posts as candidate parent posts [Seo et al., 2009, 2011]. Since each post generally only has one parent, only the top ranked result is relevant. A system using a combination of intrinsic (e.g. similarity of the quoted text in a post and on the original content) and extrinsic (e.g. the authors of the posts, or the time gap between two posts) features has been shown to be effective on this task [Seo et al., 2009, 2011].

Experiments have also been done to simultaneously model the structure and the semantics (including topic shifts) of forum threads by using topic models to infer the latent topics in a thread and using the topic distribution of posts to find reply relations [Lin et al., 2009], or to model both dialogue acts and the links between posts by presenting a hierarchical dialogue act label set and using structural features in the dialogue act classification [Kim et al., 2010c]. We will discuss dialogue act tagging in §4.2.2.

A threaded discussion is essentially a discourse between multiple participants. Inspiration for thread structure recovery models can therefore be derived from general research on discourse structures [Wolf and Gibson, 2005, Grosz and Sidner, 1986, Rosé et al., 1995, Lemon et al., 2002] and from discourse structure research on related data, like chat box conversations [Elsner and Charniak, 2008], or news article comments [Schuth et al., 2007].

4.2.2 Dialogue act tagging

Dialogue Acts (DAs), which were proposed based on the original work on speech acts [Austin, 1962, Searle, 1969], represent the meaning of discourse units at the level of illocutionary force, “the particular dimension of meaning along which statement, directive and question are distinguished” [Huddleston, 1988, p.129]. The identification of DAs in human interactions is often regarded as an important step to recover the discourse structure in the interaction. In the context of discussion forum threads this can potentially help in tasks like post-level retrieval [Bhatia et al., 2012] (see §3.1), thread-level retrieval [Wang et al., 2013b, Bhatia et al., 2016] (see §4.3), discussion summarisation [Zhou and Hovy, 2006] (see §4.5), user profiling [Kim et al., 2006], and thread visualisation [Wang et al., 2013b] (see §4.2.1).

When identifying dialogue acts in discussion forum data, a basic discourse unit can be a sentence, a paragraph or a post. While heuristic methods can reliably segment a discourse into sentences and paragraphs, automatic utterance segmentation is an open research question. For this reason, most researchers working on DA tagging in discussion forum threads work at the post level [Kim et al., 2006, Xi et al., 2004, Kim et al., 2010c, Fortuna et al., 2007, Bhatia et al., 2012].

Some researchers have used dialogue acts to annotate each individual discourse unit (e.g. Bhatia et al. [2012]), while others have treated a dialogue act as a relation *between* two discourse units (e.g. Kim et al. [2010c], Xi et al. [2004]). Classifying the DAs of posts or sentences can be done without taking the link structure into account [Bhatia et al., 2012, Jeong et al., 2009], but it is also possible to parse both the dialogue acts and the links among them at the same time [Kim et al., 2010c].

The dialogue act sets used by research in the field of discussion forums are often devised based on the requirements of specific tasks and use cases, and there is no commonly adopted dialogue act set to the best our knowledge. Table 4.1 shows an overview of the different dialogue act tag sets used in forum research. Most of them make some distinction between agreement/confirmation/support and disagreement/objection, distinguish requests for more information (clarification/elaborate/further details), and have a tag for purely social posts (polite mechanism/acknowledge and appreciate/courtesy and else other/junk/don't know).

There is also research focusing on particular types of DAs in forum threads, such as QUESTION-ANSWER pairs [Cong et al., 2008], and QUESTION-CONTEXT-ANSWER triples [Ding et al., 2008]. This research is discussed in §4.4.

	Dialogue Act Tags
[Fortuna et al., 2007]	Question, Answer, Agreement, Disagreement, Insult, Off-topic, Don't know
[Xi et al., 2004]	Question, Answer, Agreement/Amendment, Disagreement/Argument, Courtesy
[Kim et al., 2010c]	Question-question, Question-add, Question-confirmation, Question-correction, Answer-answer, Answer-add, Answer-confirmation, Answer-correction, Answer-objection, Resolution, Reproduction, Other
[Jeong et al., 2009]	Wh-Question, Yes-no Question, Rhetorical Question, Open-ended Question, Or/or-clause Question, Accept Response, Acknowledge and Appreciate, Action Motivator, Reject Response, Uncertain Response, Statement, Polite Mechanism
[Bhatia et al., 2012]	Question, Repeat Question, Clarification, Further Details, Solution, Positive Feedback, Negative Feedback, Junk
[Kim et al., 2006]	Question, Simple Answer, Complex Answer, Announcement, Suggest, Elaborate, Correct, Object, Criticize, Support, Acknowledge, Complement
[Gottipati et al., 2011]	Question, Clarifying Question, Answer, Clarifying Answer, Positive Feedback, Junk

Table 4.1: An overview of some of the different DA-tag sets used in forum research

Regarding DA classification, a range of methods have been used, including maximum entropy models [Kim et al., 2010b], SVMs [Fortuna et al., 2007, Kim et al., 2010b,c, Wang et al., 2007, Gottipati et al., 2011], rule induction methods [Cong et al., 2008], CRFs [Ding et al., 2008, Kim et al., 2010a,b,c], and Naive Bayes [Kim et al., 2010a,c]. Most research has found that Markov-like models (e.g. polygrams/n-gram language models and CRFs) with lower orders (e.g. unigram and bigram) lead to very good results [Gottipati et al., 2011].

It is interesting to note that although most research has focused on supervised methods involving only DAs, some research has approached the task via unsupervised [Cong et al., 2008], or semi-supervised [Jeong et al., 2009] methods. For example, Jeong et al. [2009] explored subtree features and semi-supervised methods to classify DAs of unlabelled discussion forum and email sentences. By comparing with a Maximum Entropy classifier (trained on unigram, bigram and trigram lexical features), they demonstrated that the subtree features could lead to similar or better results with smaller feature numbers. They also argued that semi-supervised methods (i.e. bootstrapping and boosting) with subtree features (structural features) could improve DA recognition.

A range of different features have been explored in DA classification, including lexical features such as bag-of-words [Ding et al., 2008, Kim et al., 2010b, Wang et al., 2007], structural features such as relative post position [Ding et al., 2008, Kim et al., 2010b, Wang et al., 2007], context features such as DA predictions of preceding posts [Kim et al., 2010b, Wang et al., 2007], semantic features such as similarity scores [Ding et al., 2008, Kim et al., 2010b], and graph-based features such as reply-to networks in forum threads [Fortuna et al., 2007, Jeong et al., 2009]. In general, lexical features are less effective than other features. It should also be noted that although context features were considered explicitly in some research (and found to be very important [Wang et al., 2007, Bhatia et al., 2012]), Markov-based methods are often able to capture these features inherently. Sentiment based features were found to be ineffective [Bhatia et al., 2012].

4.2.3 Thread partitioning

Thread partitioning is about identifying posts that lead to topic divergence. In cQA archives this is generally actively discouraged, because the answer posts are heavily focused on the question post. Off-topic answers are down-voted by the community, or even deleted. In discussion forums this is often less of a problem, because the focus lies more on interaction and discussion, rather than solving someone's problem and moving on to a different thread.

Identifying where topic divergence happens is important for understanding the discussion forum thread. In a retrieval setting, such threads are often only partially useful results (relevant only up to the topic shift, or only after the topic shift). Knowing if and where a shift happens can help to segment threads into coherent units, to improve retrieval results. It can also be useful for automatic forum thread summarisation, or to improve information access for visually impaired people by segmenting threads into coherent units instead of presenting them with the full thread [Kim et al., 2005].

The task is highly related to topic detection and tracking (TDT), which has received much attention outside of forums, especially in the analysis of news stories [Allan et al., 1998, Brants et al., 2003, Kumaran and Allan, 2004, Makkonen et al., 2004, Stokes and Carthy, 2001, Yang et al., 1998, Zhang et al., 2007b].

Post-level topic modelling is one way to find shifts in a conversation [Lin et al., 2009]. In such a model, it is assumed that threads have several topics, which are reflected in their posts. Post are therefore topically related to threads, but they are also related to their previous posts. This is where shifts can be detected [Lin et al., 2009].

Different types of topic shifts can be identified: shifts to a new topic, shifts to a more general topic, and shifts to a more specialised topic [Kim et al., 2005]. New topics can be distinguished by comparing the keywords of a post with the keywords in its parent post, taking quoted text into account. More general and more specific topics can be recognised by looking at the common base of a post and its parent post [Kim et al., 2005].

Experiments have also been done to track topics *across* threads, by first filtering out uninformative posts, and then using both the similarity of the content, and the similarity of the user activity to determine if two threads belong to the same topic [Zhu et al., 2008]. And at a higher level still, some research has looked at shifts in topics at the forum level by looking at the tags of questions [Gruetze et al., 2016].

4.3 Discussion forum thread retrieval

To facilitate easy access to information in discussion forums, many such websites offer a way to search through all the archived threads to find relevant content. Discussion forum thread retrieval is similar to cQA question retrieval when full threads, including answers, are retrieved. However, there is an important difference between the two: discussion forum threads have a complex discourse structure, while in cQA threads the only discourse is between the question and each answer.[1] More information on this structure and how it can be automatically derived is discussed in §4.2.

Similarly, discussion forum thread retrieval is related to discussion forum *post* retrieval (§3.1), but they differ in what exactly is retrieved. In post retrieval individual posts are returned, while in thread retrieval whole threads are returned. Discussion forum threads can be very long, and so sites may also choose to index (and return) pages instead of threads, to pinpoint the relevant information more. However, it has been argued that the full thread is the appropriate retrieval unit to use because otherwise the context of the discussion in the retrieved thread may not be clear, and this may also make it difficult to assess whether a page is relevant to a query or not [Bhatia and Mitra, 2010].

The simplest way to represent a thread is by concatenating all the posts in it to form one flat document. Standard IR models can then be applied.[2] This approach has been shown not to give good results [Albaham and Salim, 2012, 2013, Bhatia and Mitra, 2010] due to the noisy nature of forum threads [Elsas and Carbonell, 2009, Seo et al., 2009, 2011, Cho et al., 2014], or because of topic shifts that happen

[1] There is more discourse in the comments.

[2] Most researchers have used language models with Dirichlet smoothing as their baseline.

within a thread [Bhatia and Mitra, 2010]. It is also not practical because forums typically index the user generated data at the post level instead of at the thread level [Albaham and Salim, 2013].

By making use of the internal structure of the threads, the retrieval results can be improved substantially [Seo et al., 2009, 2011, Wang et al., 2013b, Elsas and Carbonell, 2009, Bhatia and Mitra, 2010]. One way of doing this is to compute a retrieval score for each post in a thread and combine these in some way (for instance linearly [Elsas and Carbonell, 2009, Seo et al., 2009, 2011]). However, this method does not offer any way to filter out the noise, junk posts or other irrelevant posts. Models that selectively include only certain posts to consider in the retrieval models have been shown to outperform models that take all posts into account, for instance by applying pseudo-cluster selection (PCS) [Seo and Croft, 2008, Elsas and Carbonell, 2009, Seo et al., 2009, 2011].

In addition to selecting only a subset of the posts in a thread to calculate its relevance, the selected posts can also be weighted according to certain properties, such as their dialogue act tags inferred through discourse analysis [Wang et al., 2013b].

Combining information from individual posts to get a thread score can also be done without making use of the thread structure. One way of doing this is to treat the thread retrieval task as an aggregate ranking task [Macdonald and Ounis, 2008a,b, 2011], and applying the Voting Model [Macdonald and Ounis, 2011, Albaham and Salim, 2012]. The idea behind the technique is to first rank the posts in the thread, then to fuse these post scores, and finally to rank threads based on the fused scores. It is unclear what the best way to aggregate the results is: score-based or rank-based [Albaham and Salim, 2012]. This approach can be seen as complementary to methods that use PCS [Elsas and Carbonell, 2009, Seo et al., 2009, 2011].

The Voting Model can be extended by adding post quality features [Albaham and Salim, 2013, Albaham et al., 2014, Heydari et al., 2016]. It has been shown that content quality features can improve thread retrieval, but this idea had previously only been applied to a simple concatenation of forum posts as the thread representation [Jiao, 2013, Wang et al., 2009c, Fan, 2009].

A feature that has been reported to be useful in this context is the ratio of users to posts [Heydari et al., 2016]. However, the usefulness of most other features varies considerably based on the dataset used. In general, quality features are more helpful for subjective data (e.g. about travel), than for objective data (e.g. technical questions) [Heydari et al., 2016]. See §2.3 for more information on post quality detection.

An interesting observation made by Albaham and Salim [2013] is that high quality posts are not necessarily relevant to a query, which explains why simply summing up the quality features for the separate posts does not necessarily give good results. Averaging them, taking the median or using the maximum value all produce better results. These aggregation methods summarise the overall quality of threads, while the summing method places too much emphasis on the quality of all the posts, overshadowing the thread relevance.

To improve thread retrieval further, user information, like reputation, can be incorporated [Faisal et al., 2016, Bhatia and Mitra, 2010]. Such features are complementary to post quality features. Experiments have shown that threads with more participants and posts tend to be of higher quality [Faisal et al., 2016].

A very different approach from the above work was taken by Bhatia and Mitra [2010], who explored forum thread-level retrieval by using language model-based inference networks to combine different structural units of threads, as well as query-independent priors. They divided a forum thread into three structural units: the thread title, the initial post and all the reply posts. The following formula was proposed to calculate ranking scores for a candidate thread T given a query Q:

$$P(T|Q) = P(T) \prod_{i=1}^{n} \Big\{ \sum_{j=1}^{m} \alpha_j P(q_i|S_{j_T}) \Big\}$$

where $P(T)$ is a query-independent prior for T, α_j is the weight for the structural unit j of T (i.e. S_{j_T}), and $P(q_i|S_{j_T})$ captures the probability of S_{j_T} generating the query term q_i. This probability was estimated using a language model with Dirichlet smoothing. Three different thread priors were explored (i.e. $P(T)$): thread length, user authority, and linking information (based on the links to other threads).

They found that a method which combines the three structural units with proper weights outperforms a method which treats the whole thread as a document. As for the priors, the “linking information” prior was found to be the most effective. In other work, “whether the thread is subjective or not” [Biyani et al., 2015], and dialogue act information [Bhatia et al., 2016] were found to be good priors. Another prior that could be used is Albaham and Salim [2012]’s voting scores, which we discussed earlier in this section.

The methods discussed above are all based on keyword search. Alternatively, a thread or unresolved post can be used as a query to retrieve similar threads [Cho et al., 2014, Singh et al., 2012]. This is similar to Question Retrieval (see §3.2).

One way of doing this is to compare two threads and to determine how well they are mutually contained within each other [Singh et al., 2012]. While using whole threads as queries can be motivated by increasing the information access on forums in general (i.e. by providing users with links to related threads), using unresolved posts as queries has the added benefit of improving the user experience by potentially resolving those posts.

In thread ranking experiments using unresolved posts as queries, it has been found that earlier posts in the archived threads are more useful than later posts [Cho et al., 2014]. This means that it is helpful to weight individual posts according to their position in the thread, to determine how much each post contributes to the retrieval rank of the thread.

Incorporating category information has been found to boost the performance of thread retrieval models [Cho et al., 2014], as it was for question retrieval (see §3.2). Experiments have also tried to make use of domain specific knowledge, by boosting posts if they contained domain specific entities (e.g. medical terms), or sentences, but only in the latter case (boosting based on sentences) was it found to be helpful [Cho et al., 2014].

The last approach to thread retrieval we would like to mention includes methods inspired by PageRank [Page et al., 1999], which have also been used effectively for question retrieval [Yang et al., 2013] (see

§3.2). PageRank is an algorithm developed for website retrieval, which assigns a probability to each page, denoting its importance. Importance is measured by the number of hyperlinks that point to a page and the importance of each of those pages.

Instead of hyperlinks, links from pages to other pages in the same forum can be used [Xu and Ma, 2006], or links that denote the overlap in the users that provided answers to a particular initial post [Chen et al., 2008]. In this last setup, two initial posts of threads are linked if they have answers by the same users. The bigger the overlap in these users, the stronger the link. An implicit assumption in this setup is that initial posts with many replies (i.e. long threads) are more important. Long threads often display topic drift. To tackle this, a decay factor can be introduced, as well as a threshold on the number of answers taken into consideration when calculating the link between two initial posts [Chen et al., 2008].

Because people are more likely to move to pages that are from the same topic, or similar ones, than to pages from a completely different topic, the threads can be clustered based on their topic(s), and a bias can be added to the model to reflect this intuition [Xu and Ma, 2006]. When only initial posts are compared instead of full threads, this approach is not suitable, because posts can be very short, which makes the clustering less reliable [Chen et al., 2008].

4.4 QA-pair extraction

Rather than trying to parse the general structure of threads, another line of research has focused on identifying and extracting specific relations between posts, or sentences in posts. The most explored task in this direction is to extract question-answer pairs, where each question-answer pair consists of a question and a corresponding answer sentence from the same discussion thread. This task may help enrich the knowledge base of cQA services [Cong et al., 2008, Ding et al., 2008, Yang et al., 2009b], improve information/answer access over forum threads [Cong et al., 2008], improve thread summarisation [Ding et al., 2008], or enhance search [Hong and Davison, 2009]. Similar work has been done for email conversations [Shrestha and McKeown, 2004].

Researchers have approached this task from different angles. For example, while some research has tried to address question extraction and answer identification at the same time [Cong et al., 2008, Wang et al., 2009b], other research has focused on extracting both question contexts (i.e. sentences which provide background information and impose constraints regarding a question) and answer sentences [Ding et al., 2008, Yang et al., 2009b, Cao et al., 2011] by assuming that questions are pre-identified.

It is generally accepted that simple heuristics, such as the presence of question marks or 5W1H words[3] are not enough to identify questions in forum posts [Cong et al., 2008]. Instead, using labeled sequential patterns (LSPs) is a good way to find question sentences [Cong et al., 2008], at least for English. For some languages, like Chinese, a sequential rule-based system is more suitable [Wang et al., 2009b]. A supervised sequence labelling approach using CRFs [Lafferty et al., 2001] to identify problem sentences, resolution sentences, and non-informative sentences in discussion forum threads has also produced good results [Raghavan et al., 2010].

A similar approach has been used to detect question context and answer sentences [Ding et al., 2008]. Context sentences are important because they can provide important background information that makes a question more meaningful or easier to understand. The CRF approach to context and answer sentence detection can be improved by modelling the interactions between sentences and making use of the thread structure [Yang et al., 2009b, Cao et al., 2011].

Sentences in posts can also be represented in a graph, and a propagation method can then be applied to distinguish question sentences and context sentences [Wang et al., 2010b]. This is very similar to the graph-based propagation method that Cong et al. [2008] used for answer ranking (see §3.3).

Other related work has looked at applying supervised classification models to identify problem sources and solution types in troubleshoot-

[3] 5W1H words are the six typical question words in English: *what*, *why*, *when*, *who*, *where*, and *how*.

ing discussions. Problem sources are things like ‘operating system’, ‘software’, ‘hardware’ or ‘network’, and solution types are ‘documentation’, ‘install’, ‘search’ and ‘support’. The aim of this work is to help users tag the general nature of their problem, and improve information access in troubleshooting-oriented technical user forums [Wang et al., 2010c].

4.5 Thread summarisation

One forum related task that has received little attention over the years is summarisation. In discussion forums this is thread summarisation, while in cQA archives this means answer summarisation. We will look at discussion forum thread summarisation first. Automatically generated summaries of discussion forum threads can provide new participants with a quick overview of the content. This can be very time efficient if a thread spans many pages [Bhatia et al., 2014]. For returning users, it can be a way to catch up on what has happened since their last visit, without having to read all the messages [Farrell et al., 2001].

CQA answer summarisation can be applied to improve the answer quality by presenting the user with one complete aggregated answer. The idea is that this improves the information access and user satisfaction.

There is related work on email summarisation [Carenini et al., 2007, Zajic et al., 2008, Lampert et al., 2008, Duboue, 2012, Ulrich, 2008, Wang et al., 2009a, Wan and McKeown, 2004, Hashem, 2014, Lam, 2002, Nenkova and Bagga, 2003, Rambow et al., 2004], concentrating on extracting key overview sentences; and chat summarisation [Zhou and Hovy, 2005, 2006, Newman, 2002, Newman and Blitzer, 2003]. We do not discuss these further.

4.5.1 Summarising discussion forum threads

Forum thread summarisation is different from traditional document summarisation in several ways. Forum threads have a significant internal structure that is very different from the structure found in other documents: some threads contain a lot of irrelevant information, and multiple authors are involved [Klaas, 2005]. Traditional single- or multi-document summarisation methods work poorly for threads [Tigelaar et al., 2010], and treating a thread as one document without regard for the internal structure also does not produce good results [Klaas, 2005, Farrell et al., 2001]. The structure of a discussion thread is important for understanding the discourse structure [Newman, 2002], which in turn is important for obtaining coherent and consistent summaries [Klaas, 2005, Farrell et al., 2001].

A discussion forum thread summary can be constructed by selecting only those posts that contain valuable contributions [Bhatia et al., 2014, Grozin et al., 2015]. This is a classification task. However, most researchers go one step further and try to identify relevant content for the summaries at the sentence level. This can be done either bottom-up (identifying relevant sentences in each post first and then deciding which ones to keep) [Farrell et al., 2001], or top-down (selecting relevant posts first and then identifying relevant sentences in those posts) [Klaas, 2005, Tigelaar et al., 2010].

The top-down approach consists of three steps: (1) identify all the posts in a thread that contain important and relevant information; (2) in those posts, identify all important sentences; and (3) combine those identified sentences in such a way that the result is a coherent summary.

The first step in the top-down approach is important to filter out junk posts and to make the summaries more concise. Forum posts tend to be fairly short. To achieve a high compression rate, it is therefore necessary to select only some posts to be included in the summary [Klaas, 2005].

To determine which posts to include when constructing the summary, the discourse structure of the thread can be used (see §4.2), which can be retrieved based on the quoted (parts of) other posts, for instance [Tigelaar et al., 2010]. The post position and the number of replies a post has received are good indicators of its importance in the thread, and readability metrics and formatting features can be used to filter out posts of very low quality [Tigelaar et al., 2010, Weimer et al., 2007]. Anaphora need to be resolved, so that it becomes possible to extract separate posts and sentences. Co-references can cross post boundaries [Tigelaar et al., 2010].

Author information can also help in distinguishing important posts from less important ones. Posts written by the initiating author are informative [Tigelaar et al., 2010], and in general the author reputation can be used to determine how important a post is [Klaas, 2005]. Authors that post more both in terms of frequency (*participation*) and number of words (*talkativity*) tend to have a bigger role in the discussion [Tigelaar et al., 2010, Klaas, 2005, Rienks, 2007]. Another way to rate authors is to look at how positive the reactions to his or her posts are [Feng et al., 2006b].

The second step in the top-down approach is to find relevant sentences in posts. This is also the first step in the bottom-up approach. One of the first things to consider here is how many sentences should be extracted per post. This is generally decided based on each post's weight or relevance score [Klaas, 2005, Tigelaar et al., 2010].

Although posts are often unstructured, the opening sentence sometimes contains a concise summary of what follows. This makes it a good candidate to be included in the thread summary. Apart from this, the sentence length, term salience, and whether a sentence ends with a quotation mark or not can be used to select relevant sentences [Klaas, 2005, Tigelaar et al., 2010]. Using term salience to select important sentences produces more on-topic, but less cohesive summaries [Klaas, 2005].

Term salience has also been used to identify relevant sentences in posts in a bottom-up approach [Farrell et al., 2001]. In this approach, the n most salient sentences from each posting were selected first. These were combined into paragraphs and then the m most salient sentences were recursively selected from the resulting set. This process was repeated as many times as necessary to obtain a summary of the desired length.

The third step in the top-down approach is easy: all researchers output the selected sentences in the original order, to aid coherence [Klaas, 2005, Tigelaar et al., 2010, Farrell et al., 2001, Bhatia et al., 2014].

Related research has looked at the relationship between individual posts in threads, and used a Roccio-style classifier to effectively use this information to summarise discussion forum threads into only a handful of words, corresponding to the topic of the thread [Feng et al., 2006a].

Evaluating thread summaries

Evaluation is one of the most difficult parts in automatic summarisation. Agreement between expert human abstractors can be below 50% [Firmin and Chrzanowski, 1999], making it difficult to assess summaries, whether automatically generated or human-generated, in a meaningful way. Different summaries of the same text can be equally good, even though they use different words. It has been shown that the evaluation metric used has a large effect on the score assigned to a particular summary, which can be countered by using multiple evaluation metrics. This has resulted in the development of the *pyramid method* [Nenkova et al., 2007], which has not been applied to forum summaries yet.

The two main aspects to be evaluated are *coverage* and *coherence.*

In one study, the system's post selection and sentence selection was compared to human annotations by computing the information ordering method Kendall's Tau-b [Lapata, 2006] and the well known summarisation evaluation metric ROUGE [Lin, 2004] respectively, and it was found that for the post selection humans agreed more with the machine's choices than with each other. For the sentence selection there was more agreement among the annotators [Tigelaar et al., 2010].

The annotators were also asked several questions to find out their opinion on the summaries. They were generally found to be useful, and the participants rated them on average 6.36 out of 10. The coherence was found to be better than the coverage. Note however, that only two threads were used in the evaluation.

In another study, a very different approach was taken to evaluation: instead of comparing the automatically generated summaries to human summaries, two different automatically generated summaries of the same thread were shown to the human annotators, who were asked to judge which one was better [Klaas, 2005]. No statistically significant findings were drawn from these annotations, which could possibly be explained by the low number of threads used. Once again, only three threads were annotated.

Experiments were done to try to evaluate whether selecting different lengths for the summaries made a difference. Two summaries of different lengths of two threads were presented to the annotators, and again no conclusions could be drawn, except for the fact that users wanted to have more than one sentence from the main post in the summaries.

The final task they gave to their annotators was to see if they could identify the correct thread subject line by looking only at the generated summaries. 35 threads were tested, and for 87% of these the correct subject line was identified, even though the summaries were only 5% of the length of the original threads. This suggests that the coverage was reasonable [Klaas, 2005].

General questions about the summaries produced mixed results. Users suggested that the system could be used to highlight important posts, rather than extract them, which is exactly how Farrell et al. [2001] implemented their system. However, they left the evaluation of their system for future work.

As for efficiency, most systems are fast, and could therefore be time saving. It took humans on average around 15 minutes to produce a summary consisting of sentences from the thread, while these could be automatically generated in a few seconds [Tigelaar et al., 2010, Klaas, 2005].

Of the papers described in this section, the only one that included a meaningful quantitative evaluation is Bhatia et al. [2014]. They only did post selection, which makes their system easier to evaluate. Even so, they used an interesting approach. Two human evaluators were asked to write summaries for a set of 200 forum threads, in their own words. These summaries were then compared against all the posts in the threads. The cosine similarity was calculated, and the top k posts were taken as the gold standard for that thread. In this way, two gold standards were created per thread. Of course, one could question cosine similarity as the basis for this. However, the results showed consistent trends over different datasets: incorporating dialogue act information and textual features together increased performance [Bhatia et al., 2014].

One interesting finding was that humans did not stick to the chronological order of the sentences. However, none of the systems discussed in this section have made an attempt to change the order of the selected sentences to improve coherence [Tigelaar et al., 2010].

4.5.2 Summarising cQA answers

While summarising discussion forum threads can be motivated by the length of many such threads and the time savings automatically generated summaries can provide, answer summarisation in cQA archives performs a different function. Only 48% of cQA questions have a unique best answer [Liu et al., 2008b]. For the other 52%, multiple answers complement each other and combining them may result in a more complete answer to a question. There is a clear relation with answer quality here, as discussed in §2.3.

The focus in cQA answer summarisation is on how to present a question asker with the best possible answer, by going beyond choosing one and instead aggregating multiple answers to provide a richer one. This can be viewed as query-focussed summarisation, where the question is the query, and the rest of the thread (the answers) are summarised to address the query.

Researchers working in this space have framed the task in several different ways. Some have worked on open questions [Liu et al., 2008b, Tomasoni and Huang, 2010, Tomasoni, 2003, Ren et al., 2016], while others looked at complex multi-sentence questions [Chan et al., 2012] and others again only took yes/no-questions into account [He and Dai, 2011]. For most threads, answer summarisation is more than simply concatenating several good answers, because 43–64% of the sentences in cQA answers are irrelevant [He and Dai, 2011]. A good aggregated answer only consists of relevant sentences. Therefore answer summarisation systems need to include a way to identify these relevant sentences. Table 4.2 lists the papers we will discuss in this section and what differentiates them.

As with discussion forum thread summarisation, two different approaches can be taken: top-down or bottom-up. The top-down approach

Paper	**Approach**	**Novelty**
Liu et al. [2008b]	top-down and bottom-up	question-type oriented answer summarisation
Tomasoni and Huang [2010]	bottom-up	scored basic elements for four aspects
Chan et al. [2012]	bottom-up	focused on complex multi-sentence questions
Wei et al. [2016]	top-down	created summary of answers of multiple relevant questions
Ren et al. [2016]	bottom-up	treated answer summarisation as an optimisation problem in sparse coding

Table 4.2: An overview of cQA answer summarisation research

is very similar to the top-down discussion forum thread summarisation approach. It consists of first selecting relevant answers, or clustering answers, and then selecting relevant sentences to include in the summary. The difference with discussion forum threads is that the first post is not included in the summary, and there is no discourse structure that can be used. While this seems to be the preferred strategy for discussion forum threads, for cQA answer summarisation most researchers opted for a bottom-up approach.

The bottom-up approach for cQA answer summarisation is quite different from the one we discussed above for discussion forum threads. It consists of first scoring linguistic units (words, n-grams, entities, etc.) and then combining those scores to select relevant sentences, for instance by using a maximum coverage model [Tomasoni and Huang, 2010].

A linguistic unit that has shown good results is a *basic element* (BE), scored based on its *quality*, *coverage*, *relevance*, and *novelty* [Tomasoni and Huang, 2010]. A BE is a ⟨head, modifier, relation⟩ triple.

Just as for discussion forum thread summarisation, it was found to be helpful to determine the number of sentences to be extracted per post, rather than globally [Tomasoni and Huang, 2010, Klaas, 2005]. The longer and more trustworthy the answer, the higher the number of sentences extracted.

Instead of summarising any type of question, some work has focused on complex multi-sentence questions, because these are most likely to suffer from incomplete *best answers* [Chan et al., 2012]. Such complex questions were divided into several sub-questions, and then a CRF classifier was applied, using both textual and non-textual features. This is a sequential labelling process. Every answer sentence in a thread was classified as either being a summary sentence or not. The summary sentences were then concatenated to form the final summary. This is therefore another example of a bottom-up approach, where all the answers are divided into smaller pieces (sentences in this case), scored or classified, and then combined to form the summary. Another model that works directly at the sentence level is the sparse-coding-based model proposed by Ren et al. [2016].

Some researchers realised that for different types of questions, different types of answers are expected. For many factual questions for instance, there is only one correct answer, possibly enriched by some extra information from other answers. For opinion or recommendation questions on the other hand, an overview of various opinions or recommendations ranked by popularity might be more appropriate [Liu et al., 2008b]. To investigate this, they constructed a question type hierarchy and an answer type hierarchy. They found the two to be highly correlated, with certain question types inviting certain answer types. This information was then used to summarise answers in different ways, depending on the question type.

Opinion questions were subdivided into two types: sentiment-oriented questions and list-oriented questions. For the list-oriented questions (like asking for a good sci-fi movie), the answers were divided into sentences. These were clustered and for each cluster, the key sentence was extracted and added to the summary, again in a bottom-up approach. For sentiment-oriented questions a voting strategy was applied, based on the number of opinion words in each answer. A summary for such questions would be an overview of how many answerers support the question statement, are neutral, or are against it.

For open questions on the other hand, a straightforward top-down multi-document summarisation (MDS) technique was used [Hovy and Lin, 1998, Lin and Hovy, 2002], where each answer was treated as a document. Answers were clustered based on their topic. From each topic, the most important answer was extracted and added to the summary. There was no sentence selection. While standard multi-document summarisation techniques have been shown not to produce good results for discussion forum threads [Tigelaar et al., 2010], the authors found in their evaluation that users were happy with the information in the summaries, although there was room for improvement in the readability [Liu et al., 2008b].

An interesting consequence of this summarisation method is that the selected summary sentences are not always output in chronological order. The key sentences extracted from the clusters for opinion questions were ordered based on cluster size, not based on their time stamp. This contrasts with the chronological ordering of sentences in discussion forum thread summarisations. This is because of the discourse structure present in discussion forum threads, which is much less present in a cQA setting. Cluster size makes more sense for cQA answer summarisation; it can be seen as an indication of how many people have a particular opinion. The more people share an opinion, the higher it should end up in the generated summary.

All the work on cQA answer summarisation that we have discussed so far focused on summarising the answers within one cQA-thread. It is however possible that other archived questions that are similar to the question at hand, also contain relevant information in their answers. If so, a summary of such answers can be used to answer new questions. One way to achieve this is to first retrieve all questions that are similar to a new question, then identify the relevant answers in the retrieved results, and summarise those by extracting relevant sentences [Wei et al., 2016]. This is a clear example of a top-down approach.

Due to a lack of comprehensive evaluation comparing the different methods, it is unclear which approach works best in general.

4.6 Thread level tasks summary

In this chapter we discussed classification and retrieval approaches at the thread level. We looked at solvedness and task orientation, and at research into identifying a thread's discourse structure, including automatic dialogue act tagging, and identifying posts that lead to topic divergence. We then examined thread retrieval strategies, which are different from post retrieval strategies because we can make use of the complex discourse structure of threads. And finally, we looked at QA-pair extraction and automatic thread summarisation. In the next chapter we focus on an aspect of forums that we have not paid much attention to until now, despite it being the backbone of every forum: the users.

5

Social forum analysis

Up until this point we have talked about the content of forums: posts and threads. In this section, we will discuss research that focuses on the people that produce this content: the users. We will first have a look at user satisfaction in §5.1, and other types of user and community analysis in §5.2. After that we will look at expert finding (§5.3) and the related tasks of question recommendation and question routing (§5.3.1), in which we try to link questions and potential answerers, based on the content of the question and the expertise of the answerer.

5.1 User satisfaction

User satisfaction is the notion of how satisfied a user is with the answers posted to his or her question. This is difficult to predict, because users have different expectations and information needs. That is, it is inherently subjective. The task is an interesting one however, because it gives us insight into people's information seeking behaviour. This could potentially help with (personalised) answer ranking, or completeness prediction (see §2.3.2) [Liu et al., 2008a, Liu and Agichtein, 2008a, Agichtein et al., 2009]. It is also an important topic to research because it is directly linked to the health of a cQA archive. A cQA community will only grow if its users are generally satisfied with the answers they get.

Results from user surveys to investigate the expectations and motivations of cQA users reveal that people mainly ask questions to fulfil a cognitive need, and they expect to receive quick responses containing accurate, complete, additional and alternative information, from trustworthy sources [Choi et al., 2014, Choi, 2013, Choi and Shah, 2016]. These expectations will influence how satisfied they are with the received answers. Furthermore, the longer it takes for a question to receive an answer, the higher the likelihood that the user is not satisfied with it [Anderson et al., 2012].

One way of measuring user satisfaction is to look at whether a user has chosen an answer as the correct one. If so, we can assume that the answer met the information need of the user, and therefore he or she was satisfied with it. If no answer was chosen on the other hand, the situation is uncertain. There may not be a satisfying answer, or the user may not know that they are supposed to choose one, or they simply do not bother [Liu et al., 2008a, Liu and Agichtein, 2008a, Agichtein et al., 2009].

The task has been treated as a classification task in which the goal is to predict whether a user will choose one answer as the best one or not. The focus lies on the positive class, for the reasons mentioned above. More than 70 different features have been investigated, categorised into six types: question features (e.g. the question title length, the posting time, or the number of answers), question-answer relationship features (e.g. the elapsed time between the question and the highest voted answer), asker history features (e.g. ratio of answers to questions posted), answerer history, textual features, and category features, which contain statistics for a given category, like the average votes given by voters from that category [Liu et al., 2008a, Agichtein et al., 2009].

The category features were found to be useful as there is high variability in the statistics per category. The asker history features were also found to have a high predictive power, possibly because recently satisfied users are likely to return. Answerer history and reputation on the other hand, was not found to be helpful [Liu et al., 2008a]. Good results could be obtained with small numbers of training examples: an F1-score of 0.75 with only 2000 examples, and an F1-score of 0.70 with only 500 examples [Liu et al., 2008a]. These results are better than the human judgements they were compared to. The task is difficult for humans because of the subjective nature.

The model can be improved by increasing the influence of the asker history. This is also interesting for training personalised user satisfaction models [Liu and Agichtein, 2008a, Agichtein et al., 2009]. The asker history can be incorporated more fully by training one model per user, or one model per group of users, based on the number of questions they have posted. Experimental results show that for users with a rich history (that have posted more than 30 questions), the individual model performs very well, but when the user history drops, the group model gives better results, because it has more data to learn from [Liu and Agichtein, 2008a, Agichtein et al., 2009].

Many search engines these days show pages of cQA questions in their result lists when a user query seems to match a cQA question. Determining the satisfaction of such web users with the answers to the returned cQA questions is one step removed from the scenario described above, and more difficult, for several reasons. First of all, the user whose satisfaction is predicted is not the same as the user that asked the question and may have a different information need and different expectations of the answers. Secondly, there is an added step from the query to the question. These two may not match, even when they are superficially similar [Liu et al., 2011].

The problem can be split into three parts: query clarity, query-question match, and answer quality. The idea behind this is that if the query is not very clear, the match between the query and the question is weak, or the quality of the answer is low, then a web user is less likely to be satisfied with the answer, and vice versa [Liu et al., 2011].

Prior work can be used to determine the query clarity [Cronen-Townsend et al., 2002, Teevan et al., 2008, Wang and Agichtein, 2010] and answer quality [Agichtein et al., 2008], and a final satisfaction score can be obtained by using the scores of the three parts in a classifier

[Liu et al., 2011]. Answer quality was found to be important. This is in line with earlier findings, which noted that the quality of the received answers has a significant impact on user satisfaction [Su et al., 2007, Wang et al., 2009e]. The composite approach described above was found to work better than a direct model which simply used all the features of the separate parts in one classifier. This model has the added benefit of being able to achieve better results when better models are developed for the separate parts [Liu et al., 2011].

In related work, researchers have also looked at how unsatisfied web searchers become cQA question askers [Liu et al., 2012].

5.2 User and community analysis

Users form the core of every forum; without users there are no discussions, no questions, and no answers. User participation has been identified as a key element of a healthy forum [Ludford et al., 2004], and making sure users are engaged is therefore very important for forums. Many studies have looked to gain an understanding of how users behave in forums, what attracts them, and what motivates them to contribute [Ludford et al., 2004, Nonnecke and Preece, 2000, Girgensohn and Lee, 2002].

Many users come to a forum for the social interaction [Harper et al., 2008, Raban, 2008], but their behaviour differs considerably. Wang and Zhang [2016] identified four kinds of cQA users based on their behavioural differences: *starters*, who ask many questions, but answer few, and are not well connected to other users; *technical editors*, who are knowledgeable users but their contributions are mainly technical edits instead of complete answers; *followers*, who do not contribute much content, but follow many topics and users; and *answerers*, who prefer to answer questions instead of ask them, and who receive the most likes and votes.

Answerers are the kind of users who most enjoy the “game” aspect of a cQA website. Gamification, giving users the option to earn votes, reputation points, badges, or similar rewards based on their contribution to the forum, has been shown to incentivise users to contribute more [Cavusoglu et al., 2015, Mamykina et al., 2011, Anderson et al., 2013, Raban, 2009], although it affects mainly the quantity, not the quality of the contributions [Lou et al., 2013, Welser et al., 2007], and the success heavily depends on how the system is implemented [Srba, 2011].

The community reinforces user reputation [Gyongyi et al., 2007]: users with a high reputation are likely to receive more votes, either because they put more effort into writing their answers, or because their reputation makes other users trust them more.

Furtado et al. [2013] identified ten different user profiles based on the quality and quantity of users' contributions, and studied transitions between them. They found that the distribution of the profiles was similar in different forums, and that it stayed mostly stable over time, even though users did transition from one profile to another over time.

Other motivations for participation in a forum, and especially a cQA archive, include wanting to learn, wanting to help others, believing they can provide knowledge that is valuable to other users, and simply having fun [Lou et al., 2011, 2013, Choi, 2013]. As for expectations, users are mainly looking for additional, alternative, accurate and complete information, and quick responses [Choi et al., 2014, Choi, 2013, Shah and Kitzie, 2012]. Users that only read posts, but do not contribute content themselves are known as "lurkers". The percentage of lurkers varies widely for different forums, but can in some instances be in the high ninety percents. There has been some research into understanding why people choose to be a lurker instead of participating actively [Nonnecke and Preece, 2000].

Some research has looked at the evolution and success of specific cQA archives. Yahoo! Answers for instance seems to be moving away from factoid questions and is becoming more effective for opinion questions [Liu and Agichtein, 2008b]. This is reflected in the motivations of their users [Choi et al., 2014, Choi, 2013]. Some factors that have been identified as reasons for the success of cQA archive StackOverflow,[1] a cQA archive that strongly favours non-opinion questions, are the tight engagement between the founders and the community, and the continuous development based on ongoing user feedback, supplied via a

[1] http://www.stackoverflow.com/

meta forum [Mamykina et al., 2011]. Even though subjective questions tend to get closed, they are found to be very popular among users of StackOverflow [Correa and Sureka, 2013].

One forum type where particular focus has been placed on user analysis is the discussion forums associated with massive open online courses ("MOOCs"), in large part because forums provide potential insights into how individual students are fairing in a subject, and how the overall subject is tracking. As such, the task of determining whether a given student is at risk of dropping out of a MOOC has received particular attention. For instance, Wong et al. [2015]analysed the relative impact that active vs. passive users have on MOOC forums, and concluded that active users have a more positive impact on the student community. Wen et al. [2014]analysed whether the overall sentiment in a student's body of posts (or in threads the student has participated in) is indicative of their likelihood to drop out, and found that the results varied across different MOOC courses, and that domain-specific understanding of what positive and negative sentiment signifies for a particular MOOC is vital to dropout prediction. Arguello and Shaffer [2015]used automatic dialogue act tagging in MOOC forum threads to help identify students in need of assistance. Onah et al. [2014]analysed the impact of posts from peers vs. experts (i.e. tutors or lecturers) on learning, and found that students tend to gain more from tutors, but that overall participation levels in discussion forums are low. Coetzee et al. [2014]found that higher forum participation levels tended to correlate with better performance and lower dropout rates on MOOCs, but that the addition of reputation systems had little impact on learning outcomes. Robinson [2015]presents a fascinating analysis of how students discuss maps in a MOOC on cartography, combining topic models, named entity recognition and geocoding to visualise the topics and places discussed in the course.

Data access and research reproducibility is a core issue with MOOC forums, as forum data is often subject to privacy constraints and accessible only to affiliates of the organisation the MOOC is offered by. A rare instance of a large-scale dataset of MOOC forum data is that of Rossi and Gnawali [2014], as part of their analysis of thread types (see

§4.1).

5.3 Expert finding

One of the main problems when working with forum data is that the quality of the posts varies considerably because, generally speaking, forums are open to anyone who would like to participate, whether they are knowledgeable or not, good communicators or not, and willing to contribute quality content or not.

In §2.3 we discussed methods to distinguish high quality content from low quality content. In this section we look at a related task: distinguishing knowledgeable from less knowledgeable users. High quality posts are often written by knowledgeable users, or *experts* [Jeon et al., 2006, Burel et al., 2016, Agichtein et al., 2008, Le et al., 2016, Shah and Pomerantz, 2010, Bian et al., 2009, Gkotsis et al., 2014, Niemann, 2015], which is why we have seen that user features are found to be helpful for post quality assessment [Lui and Baldwin, 2009, Yang et al., 2011, Agichtein et al., 2008, Burel et al., 2012, 2016, Agichtein et al., 2008, Shah, 2015, Suryanto et al., 2009, Hong and Davison, 2009, Le et al., 2016]. Developing ways of identifying experts on forums can therefore help us to identify high quality content (and vice versa) [Dom and Paranjpe, 2008].[2]

Instead of making use of the quality of questions, researchers have also looked at modelling the *difficulty* of them. The expertise of the users can then be estimated based on the difficulty of the question they have answered [Hanrahan et al., 2012].

There is a general consensus amongst researchers that expert users tend to answer many more questions than they ask [Movshovitz-Attias et al., 2013, Zolaktaf et al., 2011]. This observation has inspired several researchers to make use of graph-based methods to identify expert users [Jurczyk and Agichtein, 2007a,b, Suryanto et al., 2009, Zhou et al., 2012a, Bouguessa et al., 2008, Wang et al., 2013a, Zhao et al., 2015]. In such models, users are nodes, and edges are drawn from askers to

[2] Much work has been done on finding experts outside of forums, see for instance the survey paper by Balog et al. [2012], but we limit ourselves to finding experts in forums.

answerers. Other underlying assumptions for using such a graph are that users that ask high quality questions will receive many answers and will therefore have a high outdegree of edges, expert users tend to answer good questions, and many of them, so they will have a high in-degree of edges.

In this setup, askers can be seen as *hubs* and answerers as *authorities*, and the HITS algorithm [Kleinberg, 1999] can be applied [Jurczyk and Agichtein, 2007a,b, Guo and Hu, 2013]. Alternatively, PageRank [Page et al., 1999] can be used [Bouguessa et al., 2008, Zhou et al., 2012a, Wang et al., 2013a, Zhang et al., 2007a]. When user B answers a question of user A, and user C answers a question of user B, PageRank assumes that user C is more knowledgeable than user B, but this conclusion is only valid if the questions fall within the same category or topic [Bouguessa et al., 2008]. Experts are also easier to identify within a given domain [Jurczyk and Agichtein, 2007a, Niemann, 2015]. For these reasons, the performance of PageRank (or other graph based models) in an expert finding task can be improved by extending the model with latent topics [Haveliwala, 2002, Nie et al., 2006, Zhou et al., 2012a, Guo and Hu, 2013, Zhou et al., 2012c, Zhu et al., 2011].

Adding multiple edges between two users for multiple interactions, or weights on the edges based on the number of interactions improves results [Jurczyk and Agichtein, 2007a, Wang et al., 2013a].

Instead of placing edges from askers to all answerers, they can be placed from askers to only the users that provided a best answer [Bouguessa et al., 2008]. In classification experiments the number of answers voted as the best has been shown to be a more informative feature than the total number of answers [Sahu et al., 2016c], and so it can be expected that the in-degree of a node in this new graph is a better measure of authority than in the graphs above, where the in-degree measures the total number of answers, instead of the number of best answers.

The in-degree can be normalised within a topic or category and modelled as a mixture of two gamma distributions, where one of the distributions corresponds to experts, and one to non-experts [Bouguessa et al., 2008]. This idea can be extended by using feature vectors instead of only the in-degree, and applying a multivariate beta mixture model [Bouguessa and Romdhane, 2015].

A similar approach has been used for question routing, which we discuss in §5.3.1. The model was extended with a topic model, to capture the topical match between authoritative users and new questions [Sahu et al., 2016a].

Graphs of users can be extended by representing questions and answers as nodes too. Using such a graph, the relationship between high quality content and expert users can be exploited by estimating both at the same time using a semi-supervised classification model [Bian et al., 2009].

Graph-based approaches suffer from data sparsity. They contain only the asker–answerer interactions that actually happened, while the interactions that *could have happened* based on the expertise of both parties are left out. A complete graph would lead to better expert identification. For this reason, researchers have looked at ways to complete the graph, for instance by exploiting user similarity [Xie et al., 2016].

Several studies have looked at using temporal information (e.g. time gaps between postings of a user) and the evolution of users (e.g. how those time gaps change over time) to identify experts, future experts, or long-term contributors (who are often also experts) [Fu et al., 2016b, Movshovitz-Attias et al., 2013, Pal et al., 2012a, Fu et al., 2016b]. Three kinds of experts can be identified when analysing the changes in behavioural patterns of users: those of consistent activity, those of decreasing activity, and those of increasing activity [Pal et al., 2012a], although other research has found that expert user behaviour differs from non-expert user behaviour right from the start [Movshovitz-Attias et al., 2013, Fu et al., 2016b], and other research again has found that experts post less answers over time, while non-experts post more answers over time [Yang et al., 2014b]. Temporal information has also been used in the related task of churn prediction [Pudipeddi et al., 2014].

Experts prefer to contribute valuable answers and will therefore try to choose questions which have not received valuable answers by other users yet [Pal and Konstan, 2010, Pal et al., 2012b, Dearman and Truong, 2010]. This question selection bias is stable over time, and is a good predictor for expert identification [Pal and Konstan, 2010, Pal et al., 2012b]. Most research on expert identification tries to rank authors, or classify them as either experts of not. Alternatively users can be grouped together into several clusters based on their behaviour and performance on the forum [Pelleg and Moore, 2000, Anusha et al., 2015], for instance by using the X-means algorithm [Pelleg and Moore, 2000].

Apart from the level of expertise, users can be classified based on other aspects of their participation, like the clarity of their posts, the amount of effort they put into writing their posts, and the positivity of their contribution [Lui and Baldwin, 2010].

Deep learning techniques have so far received surprisingly little attention in the expert finding task. The only study we have been able to find makes use of a convolutional neural network (CNN). In this work, users are represented as vector representations of all the words in the questions to which they have given the best answer. Two convolutional layers and a max-pooling layer are applied to transform this rich representation into one value. This is done for each user. All the user values are then input to a fully connected softmax layer for the final decision on which users are experts and which are not [Wang et al., 2016].

5.3.1 Question recommendation and question routing

We will now look at two tasks that are highly related to expert finding: question recommendation and question routing. Question routing is about finding the right match between questions and potential answerers, by computing the semantic similarity between the question and the answerer's interests and areas of expertise. Question routing systems take a new question as input and return either a set of users or a ranked list of users that are deemed suitable to answer it.

Question recommendation is very similar to question routing, but the focus is different. While in question routing, the needs of both the asker and the answerer are taken into account, in question recommendation the focus lies on the answerers only. The goal is to present answerers with questions they might be interested in, regardless of their level of expertise, and so question recommendation systems take a user as input and return a set of questions the user might be interested in.

Because the expertise is not taken into account when recommending questions, the task boils down to computing the semantic similarity between new questions and the posting history of the user. The posting history can be taken as the questions and answers posted by the user, or as the questions the user has answered, potentially supplemented by the actual answers themselves.

To calculate the semantic similarity between users and new questions, simple methods like language models produce reasonable results [Xianfeng and Pengfei, 2016]. For a higher level semantic similarity, topic models [Qu et al., 2009] or matrix factorisation methods [Yang et al., 2014a] can be used. Topic model-based question recommendation systems can be extended by making a distinction between users' short-term interests and long-term interests [Wu et al., 2008]. This can be achieved by adding a weight on the conditional probability of a topic given a question, which can be shifted up or down based on user feedback [Wu et al., 2008].

For question routing, most work has tried to determine users' topical expertise to find the most suitable answerers for a new question, but researchers have also looked at estimating answerers' *availability* based on their past activity patterns [Li and King, 2010, Tian et al., 2013b, Dong et al., 2015] or temporal answering behaviour trends [Liu and Agichtein, 2011], at estimating the likelihood that an answerer will accept a recommended question and will answer it in a timely manner, and at understanding the reasons for choosing to answer a particular question [Dearman and Truong, 2010, Liu and Jansen, 2016].

The general idea in question routing is that a suitable answerer for a given question is someone who has answered similar questions in the past. Such users can be identified by comparing a new question to the questions a particular user has answered in the past, for instance by using a language model [Liu et al., 2005], optionally enhanced with category information [Li et al., 2011].

One study extracted representative words from user profiles and new questions, represented them as distributed representations and computed the cosine similarity between them to determine their semantic relatedness [Dong et al., 2015].

In classification experiments, textual features (e.g. lemmtised terms, POS tags), category features, and social features (e.g. voting information) have been shown to complement each other [Dror et al., 2011]. Users that participate in a lower number of categories receive higher answer rankings, but only for categories of a more factual nature, as opposed to categories that spark discussions, *parenting* for instance [Adamic et al., 2008].

The completeness of a user's personal profile (profile picture, education, work experience, website, etc.) can also be used to identify experts, because is it highly correlated with the number of reputation points earned. While most users do not have a complete profile, those who do produce higher quality content [Adaji and Vassileva, 2016].

As in expert finding, graph-based models have been explored extensively to identify knowledgeable answerers and enhanced in several ways to link answerers to questions in the right domain, for instance in combination with the language model approach mentioned above [Zhou et al., 2009], by taking into account the question's category and the categories the user is active in [Kao et al., 2010, Schall and Skopik, 2011], by incorporating the relevance of the previously answered questions to the new one [Suryanto et al., 2009], or by adding a user's descriptive ability and latent topic information to the model [Yang and Manandhar, 2014].

Topic models can be used to create topic distributions over user profiles, which can be compared to the topic distribution of a new question [Tian et al., 2013b, Sahu et al., 2016b, Guo et al., 2008]. These user topic distributions can be generated from the questions the user has answered [Sahu et al., 2016b], or from those questions *and* the answers given by the user [Tian et al., 2013b], or even including the questions the user has asked himself/herself [Guo et al., 2008].

The performance of topic models for question routing can be improved by taking the two different roles of each user (*asker* and *answerer*) into account [Xu et al., 2012], by complementing it with a term-model (BM25F [Robertson et al., 2004]) and incorporating the categories of the questions [Guo et al., 2008], by taking the tags of the questions into account [Sahu et al., 2016b, Xu et al., 2016], or by using a Segmented Topic Model (STM) [Du et al., 2010] that can assign each question of a user a separate topic distribution instead of grouping them together and creating one distribution per user [Riahi et al., 2012].

A further extension can be made by encoding two separate things in the latent topic distribution of a user: his or her topical *expertise*, and his or her topical *interest* [Yang et al., 2013, Tian et al., 2013b]. Tags or the textual content of the postings can be used to capture a user's interests, and voting information can be taken to indicate a user's expertise. These two concepts often go hand in hand, but not always, and separating them allows us to distinguish users with a high interest in an area they do not know much about (yet), from the actual knowledgeable users [Yang et al., 2013, Tian et al., 2013b].

Bayes' Theorem, shown in Equation 5.1, is often applied to make actual recommendations of users for a given question [Sahu et al., 2016b, Riahi et al., 2012, Tian et al., 2013b, Dong et al., 2015]. Here, $P(q)$ is usually assumed to be uniform across all questions, and therefore ignored. $P(u)$ can be used to encode a user's availability [Dong et al., 2015, Tian et al., 2013b], level of expertise, or authority [Dong et al., 2015], and $P(q|u)$ is the semantic similarity between a question and a user, sometimes including the expertise [Riahi et al., 2012].

$$P(u|q) \propto \frac{P(u)P(q|u)}{P(q)} \tag{5.1}$$

Finally, question routing can be cast as an item recommendation problem, where recommending new questions to suitable answerers is similar to recommending items to users in an online store. In such a model, questions are the items, answerers are the users, and forum user votes can be used as item rating scores. When looked at the problem like this, collaborative filtering methods can be used [Xu et al., 2016].

5.4 Social forum analysis summary

In this chapter we discussed research into the social aspect of forums, investigating types of users and communities. Forums tend to thrive when they have happy users with a strong motivation to use the forum on a regular basis. We reviewed ways of automatically determining user satisfaction, identifying expert users, and how we can recommend suitable questions to users and route questions to potential answerers.

6

Conclusion

In this survey, we presented an overview of research that focuses on automated analysis of forum data, including both discussion forums and cQA archives. As a general trend, we can see that much of the forum research is moving away from discussion forums, and instead is focusing more on cQA forums.

The four tasks that have received the most attention are question retrieval, answer retrieval, expert finding, and post quality assessment. Much of the other research we have discussed — for instance post type classification — can be used to enhance these tasks.

6.0.1 Standardization and comparison of methods

For some tasks it is currently difficult to compare existing work because of a lack of standardised lists of target classes or tags. For instance, in question type classification, widely varying lists of question types are used; in dialogue act tagging there is a large variation in the specific list of tags used; and in subjectivity and viewpoint classification there is no consensus over which opinion word lists to use. These fields would benefit from standardisation of theoretical and experimental grounding.

Due to the varying nature of different forums, some methods work better on certain forums than on others. In the Introduction to this survey we mentioned a spectrum across which forums exist, based on their level of moderation and acceptance of discussion threads. It would be useful to gain insight into how the degree of “discussion forum-ness” or “cQA-ness”, i.e. the specific structural characteristics of the forums, affects the effectiveness of the different methods.

At a lower level, the field would benefit from a comparison of retrieval techniques used for discussion forum posts, cQA questions, and cQA answers. For each of these, different models have been developed, but it is currently unclear how, for instance, question retrieval models perform on answer retrieval tasks, and vice versa. Knowing which models work best for which task, and understanding why, could result in valuable insights into the performance of these models, and ideas for how to improve them.

6.0.2 Challenges

In post retrieval we see the rise of deep learning methods [Zhou et al., 2015, dos Santos et al., 2015, Lei et al., 2016]. It is expected that in the future, these will also be used extensively for other forum related tasks. One challenge to overcome here is to make the models fast enough to be usable in real world settings, particularly in interactive contexts. Related to this is the fact that current retrieval models that make use of deep learning for post retrieval, usually use a standard retrieval algorithm first (often BM25), to retrieve a base set of candidates, and then only use the deep learning model to rerank these results. Not all relevant results are likely to be in this set of candidates however, and so at some point, this setup will need to be improved to be applicable in a retrieval setting. More traditional approaches can still be useful here, as it has been shown that improving the representation of questions can be effective in improving existing models [Zhou et al., 2013c].

Areas in which little work has been done include within-thread post ranking, finding related questions (as opposed to duplicate ones), using question type information to improve answer retrieval, topic shift detection, and thread summarisation. For thread summarisation one of the main challenges is the evaluation, both in forum thread summarisation and cQA answer summarisation. Current studies have only evaluated one part of a full summarisation pipeline, or have evaluated their systems on only a handful of threads. Some methods to overcome the inherent problems of summary evaluation have been developed for document summarisation (see for instance Nenkova et al. [2007]), and such methods could be used in thread summarisation too. It would also be worthwhile to look into how the order of sentences in a summary could be changed to improve coherence. Humans do this, but none of the systems we discussed looks into this.

6.0.3 Open research questions

There are still many open research question in forum research. For instance, there is the question of what constitutes a suitable gold standard, as briefly touched upon in §2.3. In post quality assessment, post retrieval, and expert finding, ratings supplied by the community, or the judgement of the question asker, are usually taken as the gold standard. However, the asker's judgement has been shown to not always be reliable [Jeon et al., 2006]. As for the community ratings, the findings are divided: some research has found that community voting is a good predictor of answer quality [Burel et al., 2012, Bian et al., 2008a], but other research has found that there is a bias in the votes [Sakai et al., 2011]. Votes can still be used as the gold standard, but only if we adjust the evaluation to take this bias into account. Sakai et al. [2011] present three different graded-relevance information retrieval metrics to do this.

Even with these known limitations, many current studies rely heavily on community ratings to evaluate their systems. This has one obvious benefit: it eliminates the need for annotation. However, it is currently unclear how we should deal with missing community ratings [Burel et al., 2012, 2016], or bad quality community ratings [Sakai et al., 2011, Jeon et al., 2006]. In answer ranking, an aspect that is often overlooked is that for some questions, all answers are of bad quality, and so answer ranking will always fail [Burel et al., 2012]. In subjectivity and viewpoint classification, one unsolved problem is that some discussion questions on current affairs look like factual ones, but they are not (at least not at the time of posting, which is reflected in their answers). An example of this can be found in §2.4. In question retrieval and duplicate question detection, two problems that have not received any attention yet are how to decide from a set of duplicate questions which one is the canonical version, and how we can recognise questions that have erroneously been flagged as a duplicate by the community. Completeness and answerability detection are tasks that are far from solved, with several studies presenting contradictory results. Ideally, we would like to go one step further and when a question is classified as incomplete, tell the user what exactly is wrong with it, e.g. it is too short, it contains an error message and should therefore also contain the code that caused it, or an example should be added.

And finally, an often ignored problem in post retrieval is what to do with questions for which there is no relevant question in the archive to be retrieved, and how to evaluate truncated lists. While some solutions for these important evaluation problems have been proposed [Peñas and Rodrigo, 2011, Liu et al., 2016], they have not yet been widely accepted by the community.

As can be seen from the extensive work surveyed in this article, research on automated analysis of forum data has received substantial and growing attention. Forums represent a core part of online interaction, as users seek to benefit from the knowledge and experiences of others. However, as the number of participants in these forums increases and the available information grows, there are increasing challenges in terms of finding relevant information, as well as in terms of potentially overloading those who contribute their knowledge and experience. The tasks and methods that we have surveyed here represent the efforts of a large community of information retrieval and natural language processing researchers to better understand the nature of information needs in forums, and to build tools that will benefit their participants. While we have outlined a number of directions for improvement, and a number of open questions, it is clear that important progress has been made.

Acknowledgements

We would like to thank the three anonymous reviewers for their insightful comments, which contributed significantly to the final shaping of this survey. We would also like to thank Mark Sanderson for his support throughout the writing/review process.

References

Ifeoma Adaji and Julita Vassileva. Towards Understanding User Participation in Stack Overflow Using Profile Data. In *Proceedings of the 8th International Conference on Social Informatics (SocInfo)*, volume Proceedings Part II, pages 3–13. Springer, 2016.

Lada A Adamic, Jun Zhang, Eytan Bakshy, and Mark S Ackerman. Knowledge Sharing and Yahoo Answers: Everyone Knows Something. In *Proceedings of the 17th International World Wide Web Conference*, pages 665–674. ACM, 2008.

Eugene Agichtein, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. Finding High-quality Content in Social Media. In *Proceedings of the 1st ACM International Conference on Web Search and Data Mining (WSDM)*, pages 183–194. ACM, 2008.

Eugene Agichtein, Yandong Liu, and Jiang Bian. Modeling Information-Seeker Satisfaction in Community Question Answering. *TKDD*, 3(2):10:1–10:27, 2009.

Eugene Agichtein, David Carmel, Donna Harman, Dan Pelleg, and Yuval Pinter. Overview of the TREC 2015 LiveQA Track. In *Proceedings of the 24th Text REtrieval Conference (TREC) (LiveQA Track)*, pages 1–9. NIST, 2015.

Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying Search Results. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining (WSDM)*, pages 5–14. ACM, 2009.

Muhammad Ahasanuzzaman, Muhammad Asaduzzaman, Chanchal K. Roy, and Kevin A. Schneider. Mining Duplicate Questions in Stack Overflow. In *Proceedings of the 13th International Conference on Mining Software Repositories (MRS)*, pages 402–412. ACM, 2016.

June Ahn, Brian S Butler, Cindy Weng, and Sarah Webster. Learning to be a Better Q'er in Social Q&A Sites: Social Norms and Information Artifacts. *JASIST*, 50(1):1–10, 2013.

Naoyoshi Aikawa, Tetsuya Sakai, and Hayato Yamana. Community QA Question Classification: Is the Asker Looking for Subjective Answers or Not? *IPSJ Online Transactions*, 4:160–168, 2011.

Ameer Tawfik Albaham and Naomie Salim. Adapting Voting Techniques for Online Forum Thread Retrieval. In *Proceedings of the 1st International Conference on Advanced Machine Learning Technologies and Applications (AMLTA)*, pages 439–448. Springer, 2012.

Ameer Tawfik Albaham and Naomie Salim. Quality Biased Thread Retrieval Using the Voting Model. In *Proceedings of the 18th Australasian Document Computing Symposium (ADCS)*, pages 97–100. ACM, 2013.

Ameer Tawfik Albaham, Naomie Salim, and Obasa Isiaka Adekunle. Leveraging Post Level Quality Indicators in Online Forum Thread Retrieval. In *Proceedings of the 1st International Conference on Advanced Data and Information Engineering (DaEng)*, pages 417–425. Springer, 2014.

James Allan, Jaime G Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. Topic Detection and Tracking Pilot Study Final Report. In *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*. NIST, 1998.

Hadi Amiri, Zheng-Jun Zha, and Tat-Seng Chua. A Pattern Matching Based Model for Implicit Opinion Question Identification. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*, pages 46–52. AAAI, 2013.

Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. Discovering Value from Community Activity on Focused Question Answering Sites: A Case Study of Stack Overflow. In *Proceedings of the 18th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 850–858. ACM, 2012.

Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. Steering User Behaviour with Badges. In *Proceedings of the 22nd International World Wide Web Conference*, pages 95–106. ACM, 2013.

J Anusha, V Smrithi Rekha, and P Bagavathi Sivakumar. A Machine Learning Approach to Cluster the Users of Stack Overflow Forum. In *Proceedings of the 2014 International Conference on Artificial Intelligence and Evolutionary Algorithms in Engineering Systems (ICAEES) (ICAEES)*, volume 2, pages 411–418. Springer, 2015.

Jaime Arguello and Kyle Shaffer. Predicting Speech Acts in MOOC Forum Posts. In *Proceedings of the 9th AAAI International Conference on Weblogs and Social Media (ICWSM)*, pages 2–11. AAAI, 2015.

Muhammad Asaduzzaman, Ahmed Shah Mashiyat, Chanchal K Roy, and Kevin A Schneider. Answering Questions about Unanswered Questions of Stack Overflow. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MRS)*, pages 97–100. IEEE, 2013.

JL Austin. *How to do Things with Words*. Oxford University Press, 1962.

Alberto Bacchelli. Mining Challenge 2013: Stack Overflow. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MRS)*, pages 53–56. IEEE, 2013.

Timothy Baldwin, David Martinez, and Richard B Penman. Automatic Thread Classification for Linux User Forum Information Access. In *Proceedings of the 12th Australasian Document Computing Symposium (ADCS)*, pages 72–79. ACM, 2007.

Krisztian Balog, Yi Fang, Maarten de Rijke, Pavel Serdyukov, and Luo Si. Expertise Retrieval. *FNTIR*, 6(2–3):127–256, 2012.

Antoaneta Baltadzhieva and Grzegorz Chrupała. Question Quality in Community Question Answering Forums: a Survey. *ACM SIGKDD Explorations Newsletter*, 17(1):8–13, 2015.

Xin-Qi Bao and Yun-Fang Wu. A Tensor Neural Network with Layerwise Pretraining: Towards Effective Answer Retrieval. *JCST*, 31(6):1151–1160, 2016.

Alberto Barrón-Cedeno, Simone Filice, Giovanni Da San Martino, Shafiq Joty, Lluís Màrquez, Preslav Nakov, and Alessandro Moschitti. Thread-Level Information for Comment Classification in Community Question Answering. In *Proceedings of the Joint 53rd Annual Meeting of the Association for Computational Linguistics (ACL) and the 7th International Joint Conference on Natural Language Processing*, volume Volume 2: Short Papers, pages 687–693. ACL, 2015.

Yonatan Belinkov, Mitra Mohtarami, Scott Cyphers, and James Glass. VectorSLU: A continuous word vector approach to answer selection in community question answering systems. In *Proceedings of the 9th Conference on Semantic Evaluation (SemEval)*, page 282. ACL, 2015.

Yoshua Bengio, Pascal Lamblin, Dan Popovici, Hugo Larochelle, et al. Greedy Layer-wise Training of Deep Networks. *NIPS*, 19:153, 2007.

Adam Berger and John Lafferty. Information Retrieval as Statistical Translation. In *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 222–229. ACM, 1999.

Adam Berger, Rich Caruana, David Cohn, Dayne Freitag, and Vibhu Mittal. Bridging the Lexical Chasm: Statistical Approaches to Answer-Finding. In *Proceedings of the 23rd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 192–199. ACM, 2000.

Delphine Bernhard and Iryna Gurevych. Answering Learners' Questions by Retrieving Question Paraphrases from Social Q&A Sites. In *Proceedings of the 3rd Workshop on Innovative Use of NLP for Building Educational Applications (BEA)*, pages 44–52. ACL, 2008.

Delphine Bernhard and Iryna Gurevych. Combining Lexical Semantic Resources with Question & Answer Archives for Translation-Based Answer Finding. In *Proceedings of the Joint 47th Annual Meeting of the Association for Computational Linguistics (ACL) and the 4th International Joint Conference on Natural Language Processing*, pages 728–736. ACL, 2009.

Abraham Bernstein and Esther Kaufmann. GINO - A Guided Input Natural Language Ontology Editor. In *Proceedings of the 5th International Semantic Web Conference (ISWC)*, pages 144–157. Springer, 2006.

Sumit Bhatia and Prasenjit Mitra. Adopting Inference Networks for Online Thread Retrieval. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, pages 1300–1305. AAAI, 2010.

Sumit Bhatia, Prakhar Biyani, and Prasenjit Mitra. Classifying User Messages for Managing Web Forum Data. In *Proceedings of the 15th International Workshop on the Web and Databases (WebDB)*, pages 13–18. ACM, 2012.

Sumit Bhatia, Prakhar Biyani, and Prasenjit Mitra. Summarizing Online Forum Discussions - Can Dialog Acts of Individual Messages Help? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2127–2131. ACL, 2014.

Sumit Bhatia, Prakhar Biyani, and Prasenjit Mitra. Identifying the Role of Individual User Messages in an Online Discussion and its Use in Thread Retrieval. *JASIST*, 67(2):276–288, 2016.

Jiang Bian, Yandong Liu, Eugene Agichtein, and Hongyuan Zha. Finding the Right Facts in the Crowd: Factoid Question Answering over Social Media. In *Proceedings of the 17th International World Wide Web Conference*, pages 467–476. ACM, 2008a.

Jiang Bian, Yandong Liu, Eugene Agichtein, and Hongyuan Zha. A Few Bad Votes Too Many?: Towards Robust Ranking in Social Media. In *Proceedings of the 4th International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, pages 53–60. ACM, 2008b.

Jiang Bian, Yandong Liu, Ding Zhou, Eugene Agichtein, and Hongyuan Zha. Learning to Recognize Reliable Users and Content in Social Media with Coupled Mutual Reinforcement. In *Proceedings of the 18th International World Wide Web Conference*, pages 51–60. ACM, 2009.

Prakhar Biyani. *Analyzing Subjectivity and Sentiment of Online Forums*. PhD thesis, The Pennsylvania State University, 2014.

Prakhar Biyani, Sumit Bhatia, Cornelia Caragea, and Prasenjit Mitra. Thread Specific Features are Helpful for Identifying Subjectivity Orientation of Online Forum Threads. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 295–310. ACL, 2012.

Prakhar Biyani, Sumit Bhatia, Cornelia Caragea, and Prasenjit Mitra. Using Non-lexical Features for Identifying Factual and Opinionative Threads in Online Forums. *KBS*, 69:170–178, 2014.

Prakhar Biyani, Sumit Bhatia, Cornelia Caragea, and Prasenjit Mitra. Using Subjectivity Analysis to Improve Thread Retrieval in Online Forums. In *Proceedings of the 37th Annual European Conference on Information Retrieval Research (ECIR): Advances in Information Retrieval*, pages 495–500. Springer, 2015.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *JourMLR*, 3:993–1022, 2003.

Mohan John Blooma, Alton Yeow-Kuan Chua, and Dion Hoe-Lian Goh. A Predictive Framework for Retrieving the Best Answer. In *Proceedings of the 23rd ACM Symposium on Applied Computing (SAC)*, pages 1107–1111. ACM, 2008.

Mohan John Blooma, Alton Yeow-Kuan Chua, and Dion Hoe-Lian Goh. What Makes a High-Quality User-Generated Answer? *IEEE Internet Computing*, 15(1):66–71, 2011.

Mohan John Blooma, Dion Hoe-Lian Goh, and Alton Yeow-Kuan Chua. Predictors of High-Quality Answers. *Online Information Review*, 36(3):383–400, 2012.

Avrim Blum and Tom Mitchell. Combining Labeled and Unlabeled Data with Co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT)*, pages 92–100. ACM, 1998.

Dasha Bogdanova and Jennifer Foster. This is how we do it: Answer Reranking for Open-Domain How Questions with Paragraph Vectors and Minimal Feature Engineering. In *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1290–1295. ACL, 2016.

Ingwer Borg and Patrick JF Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2005.

Mohamed Bouguessa and Lotfi Ben Romdhane. Identifying Authorities in Online Communities. *TIST*, 6(3):30, 2015.

Mohamed Bouguessa, Benoît Dumoulin, and Shengrui Wang. Identifying Authoritative Actors in Question-Answering Forums: the Case of Yahoo! Answers. In *Proceedings of the 14th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 866–874. ACM, 2008.

Thorsten Brants, Francine Chen, and Ayman Farahat. A System for New Event Detection. In *Proceedings of the 26th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 330–337. ACM, 2003.

Chris Brockett, William B Dolan, and Michael Gamon. Correcting ESL Errors Using Phrasal SMT Techniques. In *Proceedings of the 21st International Conference on Computational Linguistics (COLING) and the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 249–256. ACL, 2006.

Jane Bromley, James W Bentz, Léon Bottou, Isabelle Guyon, Yann LeCun, Cliff Moore, Eduard Säckinger, and Roopak Shah. Signature Verification Using a "Siamese" Time Delay Neural Network. *PRAI*, 7(04):669–688, 1993.

Peter F Brown, Stephen A Della-Pietra, Vincent J Della-Pietra, and Robert L Mercer. The Mathematics of Statistical Machine Translation. *Computational Linguistics*, 19(2):263–313, 1993.

Razvan Bunescu and Yunfeng Huang. Learning the Relative Usefulness of Questions in Community QA. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 97–107. ACL, 2010a.

Razvan Bunescu and Yunfeng Huang. A Utility-driven Approach to Question Ranking in Social QA. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, pages 125–133. ACL, 2010b.

Grégoire Burel. *Community and Thread Methods for Identifying Best Answers in Online Question Answering Communities*. PhD thesis, The Open University, 2016.

Grégoire Burel, Yulan He, and Harith Alani. Automatic Identification of Best Answers in Online Enquiry Communities. In *Proceedings of the Extended Semantic Web Conference (ESWC)*, pages 514–529. Springer, 2012.

Gregoire Burel, Paul Mulholland, and Harith Alani. Structural Normalisation Methods for Improving Best Answer Identification in Question Answering Communities. In *Proceedings of the 25th International World Wide Web Conference*, pages 673–678. ACM, 2016.

Moira Burke, Elisabeth Joyce, Tackjin Kim, Vivek Anand, and Robert Kraut. Introductions and Requests: Rhetorical Strategies that Elicit Response in Online Communities. In *Proceedings of the 3rd Communities and Technologies Conference*, pages 21–39. Springer, 2007.

Robin D Burke, Kristian J Hammond, Vladimir Kulyukin, Steven L Lytinen, Noriko Tomuro, and Scott Schoenberg. Question Answering from Frequently Asked Question Files: Experiences with the FAQ Finder System. *AI*, 18(2):57, 1997.

Li Cai, Guangyou Zhou, Kang Liu, and Jun Zhao. Learning the Latent Topics for Question Retrieval in Community QA. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 273–281. ACL, 2011.

Fabio Calefato, Filippo Lanubile, and Nicole Novielli. Moving to Stack Overflow: Best-Answer Prediction in Legacy Developer Forums. In *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, page 13. ACM, 2016.

Xin Cao, Gao Cong, Bin Cui, Christian SÃÿndergaard Jensen, and Ce Zhang. The Use of Categorization Information in Language Models for Question Retrieval. In *Proceedings of the 18th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 265–274. ACM, 2009.

Xin Cao, Gao Cong, Bin Cui, and Christian S Jensen. A Generalized Framework of Exploring Category Information for Question Retrieval in Community Question Answer Archives. In *Proceedings of the 19th International World Wide Web Conference*, pages 201–210. ACM, 2010.

Xin Cao, Gao Cong, Bin Cui, Christian S Jensen, and Quan Yuan. Approaches to Exploring Category Information for Question Retrieval in Community Question-Answer Archives. *TOIS*, 30(2):7, 2012.

Yunbo Cao, Wen-Yun Yang, Chin-Yew Lin, and Yong Yu. A Structural Support Vector Method for Extracting Contexts and Answers of Questions from Online Forums. *IPM*, 47(6):886–898, 2011.

Giuseppe Carenini, Raymond T Ng, and Xiaodong Zhou. Summarizing Email Conversations with Clue Words. In *Proceedings of the 16th International World Wide Web Conference*, pages 91–100. ACM, 2007.

David Carmel, Avihai Mejer, Yuval Pinter, and Idan Szpektor. Improving Term Weighting for Community Question Answering Search Using Syntactic Analysis. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 351–360. ACM, 2014.

Rose Catherine, Amit Singh, Rashmi Gangadharaiah, Dinesh Raghu, and Karthik Visweswariah. Does *Similarity* Matter? The Case of Answer Extraction from Technical Discussion Forums. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 175–184. ACL, 2012.

Rose Catherine, Rashmi Gangadharaiah, Karthik Visweswariah, and Dinesh Raghu. Semi-Supervised Answer Extraction from Discussion Forums. In *Proceedings of the 6th International Joint Conference on Natural Language Processing*, pages 1–9. ACL, 2013.

Huseyin Cavusoglu, Zhuolun Li, and Ke-Wei Huang. Can Gamification Motivate Voluntary Contributions?: The Case of StackOverflow Q&A Community. In *Proceedings of the 18th ACM Conference Companion on Computer Supported Cooperative Work & Social Computing (CSCW)*, pages 171–174. ACM, 2015.

Pedro Chahuara, Thomas Lampert, and Pierre Gancarski. Retrieving and Ranking Similar Questions from Question-Answer Archives Using Topic Modelling and Topic Distribution Regression. In *Proceedings of the 20th International Conference on Theory and Practice of Digital Libraries (TPDL): Research and Advanced Rechnology for Digital Libraries*, pages 41–53. Springer, 2016.

Kevin Chai, Pedram Hayati, Vidyasagar Potdar, Chen Wu, and Alex Talevski. Assessing Post Usage for Measuring the Quality of Forum Posts. In *Proceedings of the 4th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*, pages 233–238. IEEE, 2010.

Wen Chan, Xiangdong Zhou, Wei Wang, and Tat-Seng Chua. Community Answer Summarization for Multi-Sentence Question with Group L1 Regularization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, volume Volume 1: Long Papers, pages 582–591. ACL, 2012.

Wen Chan, Jintao Du, Weidong Yang, Jinhui Tang, and Xiangdong Zhou. Term Selection and Result Reranking for Question Retrieval by Exploiting Hierarchical Classification. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 141–150. ACM, 2014.

Guibin Chen, Chunyang Chen, Zhenchang Xing, and Bowen Xu. Learning a Dual-Language Vector Space for Domain-Specific Cross-Lingual Question Retrieval. In *Proceedings of the 31st IEEE/ACM International Conference On Automated Software Engineering (ASE)*, pages 744–755. ACM, 2016a.

Long Chen, Dell Zhang, and Levene Mark. Understanding User Intent in Community Question Answering. In *Proceedings of the 21st International World Wide Web Conference*, pages 823–828. ACM, 2012.

Long Chen, Joemon M Jose, Haitao Yu, and Fajie Yuan. A Hybrid Approach for Question Retrieval in Community Question Answerin. *The Computer Journal, Section C: Computational Intelligence, Machine Learning and Data Analytics*, pages 1–13, 2016b.

Zhi Chen, Li Zhang, and Weihua Wang. PostingRank: Bringing Order to Web Forum Postings. In *Proceedings of the Asia Information Retrieval Symposium (AIRS)*, pages 377–384. Springer, 2008.

Jason HD Cho, Parikshit Sondhi, Chengxiang Zhai, and Bruce R Schatz. Resolving Healthcare Forum Posts via Similar Thread Retrieval. In *Proceedings of the 5th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics (BCB)*, pages 33–42. ACM, 2014.

Erik Choi. Motivations and Expectations for Asking Questions within Online Q&A. *TCDL*, 9(2):29–35, 2013.

Erik Choi and Chirag Shah. Asking for More than an Answer: What do Askers Expect in Online Q&A Services? *JIS*, pages 1–12, 2016.

Erik Choi, Vanessa Kitzie, and Chirag Shah. Developing a Typology of Online Q&A Models and Recommending the Right Model for Each Question Type. *JASIST*, 49(1):1–4, 2012.

Erik Choi, Vanessa Kitzie, and Chirag Shah. Investigating Motivations and Expectations of Asking a Question in Social Q&A. *First Monday*, 19(3), 2014.

Alton YK Chua and Snehasish Banerjee. Measuring the Effectiveness of Answers in Yahoo! Answers. *Online Information Review*, 39(1):104–118, 2015a.

Alton YK Chua and Snehasish Banerjee. Answers or No Answers: Studying Question Answerability in Stack Overflow. *JIS*, pages 720–731, 2015b.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In *Proceedings of the 2014 Workshop on Deep Learning and Representation (held at NIPS 2014)*, pages 1–9. MIT Press, 2014.

Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. Novelty and Diversity in Information Retrieval Evaluation. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 659–666. ACM, 2008.

Charles LA Clarke, Nick Craswell, Ian Soboroff, and Azin Ashkan. A Comparative Analysis of Cascade Measures for Novelty and Diversity. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 75–84. ACM, 2011.

Derrick Coetzee, Armando Fox, Marti A. Hearst, and Björn Hartmann. Should your MOOC Forum use a Reputation System? In *Proceedings of the 17th ACM Conference Companion on Computer Supported Cooperative Work & Social Computing (CSCW)*, pages 1176–1187. ACM, 2014.

Gao Cong, Long Wang, Chin-Yew Lin, Young-In Song, and Yueheng Sun. Finding Question-Answer Pairs from Online Forums. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 467–474. ACM, 2008.

Gregorio Convertino, Massimo Zancanaro, Tiziano Piccardi, and Felipe Ortega. Toward a Mixed-Initiative QA system From Studying Predictors in Stack Exchange to Building a Mixed-Initiative Tool. *International Journal of Human-Computer Studies*, 99:1–20, 2017.

Denzil Correa and Ashish Sureka. Fit or Unfit: Analysis and Prediction of 'Closed Questions' on Stack Overflow. In *Proceedings of the 1st ACM Conference on Online Social Networks*, pages 201–212. ACM, 2013.

Denzil Correa and Ashish Sureka. Chaff from the Wheat: Characterization and Modeling of Deleted Questions on Stack Overflow. In *Proceedings of the 23rd International World Wide Web Conference*, pages 631–642. ACM, 2014.

Steve Cronen-Townsend, Yun Zhou, and W Bruce Croft. Predicting Query Performance. In *Proceedings of the 25th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 299–306. ACM, 2002.

Giovanni Da San Martino, Alberto Barrón Cedeño, Salvatore Romeo, Antonio Uva, and Alessandro Moschitti. Learning to Re-Rank Questions in Community Question Answering Using Advanced Features. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1997–2000. ACM, 2016.

Daniel Hasan Dalip, Marcos André Gonçalves, Marco Cristo, and Pavel Calado. Exploiting User Feedback to Learn to Rank Answers in Q&A Forums: A Case Study with Stack Overflow. In *Proceedings of the 36th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 543–552. ACM, 2013.

Arpita Das, Manish Shrivastava, and Manoj Chinnakotla. Mirror on the Wall: Finding Similar Questions with Deep Structured Topic Modeling. In *Proceedings of the 2016 Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pages 454–465. Springer, 2016a.

Arpita Das, Harish Yenala, Manoj Chinnakotla, and Manish Shrivastava. Together We Stand: Siamese Networks for Similar Question Retrieval. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 378–387. ACL, 2016b.

John Davies, York Sure, Holger Lausen, Ying Ding, Michael Stollberg, Dieter Fensel, Rubén Lara Hernández, and Sung-Kook Han. Semantic Web Portals: State-of-the-Art Survey. *Journal of Knowledge Management*, 9(5): 40–49, 2005.

David Dearman and Khai N Truong. Why Users of Yahoo! Answers do not Answer Questions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 329–332. ACM, 2010.

P Deepak. MixKMeans: Clustering Question-Answer Archives. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1576–1585. ACL, 2016.

Shilin Ding, Gao Cong, Chin-Yew Lin, and Xiaoyan Zhu. Using Conditional Random Fields to Extract Contexts and Answers of Questions from Online Forums. *ACL*, 8:710–718, 2008.

Byron Dom and Deepa Paranjpe. A Bayesian Technique for Estimating the Credibility of Question Answerers. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, pages 399–409. SIAM, 2008.

Hualei Dong, Jian Wang, Hongfei Lin, Bo Xu, and Zhihao Yang. Predicting Best Answerers for New Questions: An Approach Leveraging Distributed Representations of Words in Community Question Answering. In *Proceedings of the 9th International Conference on Frontier of Computer Science and Technology*, pages 13–18. IEEE, 2015.

Cıcero dos Santos, Luciano Barbosa, Dasha Bogdanova, and Bianca Zadrozny. Learning Hybrid Representations to Retrieve Semantically Equivalent Questions. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL) and the 7th International Joint Conference on Natural Language Processing*, volume 2, pages 694–699. ACL, 2015.

Gideon Dror, Yehuda Koren, Yoelle Maarek, and Idan Szpektor. I want to Answer; who has a Question?: Yahoo! Answers Recommender System. In *Proceedings of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 1109–1117. ACM, 2011.

Gideon Dror, Yoelle Maarek, and Idan Szpektor. Will my Question be Answered? Predicting “Question Answerability" in Community Question-Answering Sites. In *Proceedings of the 2013 Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, pages 499–514. Springer, 2013.

Lan Du, Wray Buntine, and Huidong Jin. A Segmented Topic Model Based on the Two-Parameter Poisson-Dirichlet Process. *Machine Learning*, 81(1): 5–19, 2010.

Huizhong Duan and Chengxiang Zhai. Exploiting Thread Structures to Improve Smoothing of Language Models for Forum Post Retrieval. In *Proceedings of the 33rd Annual European Conference on Information Retrieval Research (ECIR): Advances in Information Retrieval*, pages 350–361. Springer, 2011.

Huizhong Duan, Yunbo Cao, Chin-Yew Lin, and Yong Yu. Searching Questions by Identifying Question Topic and Question Focus. In *Proceedings of the 2008 Annual Meeting of the Association for Computational Linguistics (ACL)-hlt*, pages 156–164. ACL, 2008.

Pablo Ariel Duboue. Extractive Email Thread Summarization: Can we do Better than He Said She Said? In *Proceedings of the 7th International Conference on Natural Language Generation (INLG)*, pages 85–89. ACL, 2012.

Jonathan L Elsas and Jaime G Carbonell. It Pays to be Picky: an Evaluation of Thread Retrieval in Online Forums. In *Proceedings of the 32nd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 714–715. ACM, 2009.

Micha Elsner and Eugene Charniak. You Talking to Me? A Corpus and Algorithm for Conversation Disentanglement. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL)-hlt*, pages 834–842. ACL, 2008.

Anthony Fader, Luke S Zettlemoyer, and Oren Etzioni. Paraphrase-Driven Learning for Open Question Answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1608–1618. ACL, 2013.

Ch Muhammad Shahzad Faisal, Ali Daud, Faisal Imran, and Seungmin Rho. A Novel Framework for Social Web Forums' Thread Ranking Based on Semantics and Post Quality Features. *The Journal of Supercomputing*, pages 1–20, 2016.

Weiguo Fan. *Effective Search in Online Knowledge Communities: A Genetic Algorithm Approach.* PhD thesis, Virginia Polytechnic Institute and State University, 2009.

Robert Farrell, Peter G Fairweather, and Kathleen Snyder. Summarization of Discussion Groups. In *Proceedings of the 10th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 532–534. ACM, 2001.

Donghui Feng, Jihie Kim, Erin Shaw, and Eduard Hovy. Towards Modeling Threaded Discussions Using Induced Ontology Knowledge. In *Proceedings of the 21st AAAI Conference on Artificial Intelligence*, pages 1289–1294. AAAI, 2006a.

Donghui Feng, Erin Shaw, Jihie Kim, and Eduard Hovy. Learning to Detect Conversation Focus of Threaded Discussions. In *Proceedings of the 2006 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 208–215. ACL, 2006b.

Minwei Feng, Bing Xiang, Michael R Glass, Lidan Wang, and Bowen Zhou. Applying Deep Learning to Answer Selection: A Study and an Open Task. In *Proceedings of the 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 813–820. IEEE, 2015.

Alejandro Figueroa and Günter Neumann. Context-Aware Semantic Classification of Search Queries for Browsing Community Question-Answering Archives. *KBS*, 96:1–13, 2016.

Simone Filice, Danilo Croce, Alessandro Moschitti, and Roberto Basili. KeLP at SemEval-2016 Task 3: Learning Semantic Relations between Questions and Answers. *Proceedings of the 10th Conference on Semantic Evaluation (SemEval)*, pages 1116–1123, 2016.

Therese Firmin and Michael J Chrzanowski. An Evaluation of Automatic Text Summarization Systems. *AATS*, 325:336, 1999.

Blaz Fortuna, Eduarda Mendes Rodrigues, and Natasa Milic-Frayling. Improving the Classification of Newsgroup Messages Through Social Network Analysis. In *Proceedings of the 16th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 877–880. ACM, 2007.

Daniel Fried, Peter Jansen, Gustave Hahn-Powell, Mihai Surdeanu, and Peter Clark. Higher-order Lexical Semantic Models for Non-Factoid Answer Reranking. *TACL*, 3:197–210, 2015.

Hongping Fu, Zhendong Niu, Chunxia Zhang, Hanchao Yu, Jing Ma, Jie Chen, Yiqiang Chen, and Junfa Liu. ASELM: Adaptive Semi-Supervised ELM with Application in Question Subjectivity Identification. *Neurocomputing*, 207:599–609, 2016a.

Min Fu, Min Zhu, Yabo Su, Qiuhui Zhu, and Mingzhao Li. Modeling Temporal Behavior to Identify Potential Experts in Question Answering Communities. In *Proceedings of the 2016 International Conference on Cooperative Design, Visualization and Engineering (CDVE)*, pages 51–58. Springer, 2016b.

Bojan Furlan, Bosko Nikolic, and Veljko Milutinovic. A Survey of Intelligent Question Routing Systems. In *Proceedings of the 6th IEEE International Conference Intelligent Systems (IS)*, pages 014–020. IEEE, 2012.

Adabriand Furtado, Nazareno Andrade, Nigini Oliveira, and Francisco Brasileiro. Contributor Profiles, their Dynamics, and their Importance in Five Q&A Sites. In *Proceedings of the 2013 ACM Conference Companion on Computer Supported Cooperative Work & Social Computing (CSCW)*, pages 1237–1252. ACM, 2013.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. Posterior Regularization for Structured Latent Variable Models. *JourMLR*, 11 (Jul):2001–2049, 2010.

Li Gao, Yao Lu, Qin Zhang, Hong Yang, and Yue Hu. Query Expansion for Exploratory Search with Subtopic Discovery in Community Question Answering. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pages 4715–4720. IEEE, 2016.

Nikesh Garera, Chris Callison-Burch, and David Yarowsky. Improving Translation Lexicon Induction from Monolingual Corpora via Dependency Contexts and Part-of-Speech Equivalences. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL)*, pages 129–137. ACL, 2009.

Rich Gazan. Social Q&A. *JASIST*, 62(12):2301–2312, 2011.

S Geerthik, S Venkatraman, and Rajiv Gandhi. AnswerRank: Identifying Right Answers in QA system. *IJECE*, 6(4):1889, 2016.

T Georgiou, M Karvounis, and Y Ioannidis. Extracting Topics of Debate Between Users on Web Discussion Boards. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data.* ACM, 2010.

Andreas Girgensohn and Alison Lee. Making Web Sites Be Places for Social Interaction. In *Proceedings of the 2002 ACM Conference Companion on Computer Supported Cooperative Work & Social Computing (CSCW)*, pages 136–145. ACM, 2002.

George Gkotsis, Karen Stepanyan, Carlos Pedrinaci, John Domingue, and Maria Liakata. It's All in the Content: State of the Art Best Answer Prediction Based on Discretisation of Shallow Linguistic Features. In *Proceedings of the 2014 Web Science Conference (WebSci)*, pages 202–210. ACM, 2014.

Swapna Gottipati, David Lo, and Jing Jiang. Finding Relevant Answers in Software Forums. In *Proceedings of the 26th IEEE/ACM International Conference On Automated Software Engineering (ASE)*, pages 323–332. IEEE, 2011.

Barbara J Grosz and Candace L Sidner. Attention, Intention and the Structure of Discourse. *Computational Linguistics*, 12(3):175–204, 1986.

Vladislav A Grozin, Natalia F Gusarova, and Natalia V Dobrenko. Feature Selection for Language Independent Text Forum Summarization. In *Proceedings of the 6th International Conference on Knowledge Engineering and the Semantic Web (KESW)*, pages 63–71. Springer, 2015.

Toni Gruetze, Ralf Krestel, and Felix Naumann. Topic Shifts in StackOverflow: Ask it Like Socrates. In *Natural Language Processing and Information Systems: Proceedings of the 21st International Conference on Applications of Natural Language to Information Systems, NLDB 2016*, pages 213–221. Springer, 2016.

Jeanette K Gundel and Thorstein Fretheim. Topic and Focus. *The Handbook of Pragmatics*, 175:196, 2004.

Jinwen Guo, Shengliang Xu, Shenghua Bao, and Yong Yu. Tapping on the Potential of Q&A Community by Recommending Answer Providers. In *Proceedings of the 17th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 921–930. ACM, 2008.

Lifan Guo and Xiaohua Hu. Identifying Authoritative and Reliable Contents in Community Question Answering with Domain Knowledge. In *Proceedings of the 2013 Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pages 133–142. Springer, 2013.

Iryna Gurevych, Delphine Bernhard, Kateryna Ignatova, and Cigdem Toprak. Educational Question Answering Based on Social Media Content. In *Proceedings of the International Conference on Artificial Intelligence in Education (IJAIED)*, pages 133–140. Springer, 2009.

Zoltan Gyongyi, Georgia Koutrika, Jan Pedersen, and Hector Garcia-Molina. Questioning Yahoo! Answers. Technical Report Technical Report, Stanford Infolab, 2007.

Xiaohui Han, Jun Ma, Yun Wu, and Chaoran Cui. A Novel Machine Learning Approach to Rank Web Forum Posts. *Soft Computing*, 18(5):941–959, 2014.

Benjamin V Hanrahan, Gregorio Convertino, and Les Nelson. Modeling Problem Difficulty and Expertise in StackOverflow. In *Proceedings of the 2012 ACM Conference Companion on Computer Supported Cooperative Work & Social Computing (CSCW)*, pages 91–94. ACM, 2012.

Tianyong Hao and Eugene Agichtein. Finding Similar Questions in Collaborative Question Answering Archives: Toward Bootstrapping-based Equivalent Pattern Learning. *Information Retrieval*, 15(3):332–353, 2012a.

Tianyong Hao and Eugene Agichtein. Bootstrap-based Equivalent Pattern Learning for Collaborative Question Answering. In *Proceedings of the 13th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 318–329. Springer, 2012b.

Sanda M Harabagiu, Dan I Moldovan, Marius Paşca, Rada Mihalcea, Mihai Surdeanu, Răzvan Bunescu, Corina R Gîrju, Vasile Rus, and Paul Morărescu. Falcon: Boosting Knowledge for Answer Engines. In *Proceedings of the 9th Text REtrieval Conference (TREC)*, pages 479–488. NIST, 2000.

F Maxwell Harper, Daphne Raban, Sheizaf Rafaeli, and Joseph A Konstan. Predictors of Answer Quality in Online Q&A Sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 865–874. ACM, 2008.

F Maxwell Harper, Daniel Moy, and Joseph A Konstan. Facts or Friends?: Distinguishing Informational and Conversational Questions in Social Q&A Sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 759–768. ACM, 2009.

Mithak I Hashem. Improvement of Email Summarization Using Statistical Based Method. *International Journal of Computer Science and Mobile Computing (IJCSMC)*, 3(2):382–388, 2014.

Ahmed Hassan, Vahed Qazvinian, and Dragomir Radev. What's with the Attitude?: Identifying Sentences with Attitude in Online Discussions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1245–1255. ACL, 2010.

Taher H Haveliwala. Topic-Sensitive PageRank. In *Proceedings of the 11th International World Wide Web Conference*, pages 517–526. ACM, 2002.

Jing He and Decheng Dai. Summarization of Yes/No Questions Using a Feature Function Model. In *Proceedings of the 3rd Asian Conference on Machine Learning (ACML)*, pages 351–366. Springer, 2011.

Ulf Hermjakob. Parsing and Question Classification for Question Answering. In *Proceedings of the ACL Workshop on Open-Domain Question Answering (QA)*, volume 12, pages 1–6. ACL, 2001.

Atefeh Heydari, Mohammadali Tavakoli, Zuriati Ismail, and Naomie Salim. Leveraging Quality Metrics in Voting Model Based Thread Retrieval. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 10(1):117–123, 2016.

Geoffrey E Hinton. Training Products of Experts by Minimizing Contrastive Divergence. *Neural Computation*, 14(8):1771–1800, 2002.

Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 18(7):1527–1554, 2006.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

Thomas Hofmann. Probabilistic Latent Semantic Indexing. In *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 50–57. ACM, 1999.

Liangjie Hong and Brian D Davison. A Classification-Based Approach to Question Answering in Discussion Boards. In *Proceedings of the 32nd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 171–178. ACM, 2009.

Doris Hoogeveen, Karin M Verspoor, and Timothy Baldwin. CQADupStack: A Benchmark Data Set for Community Question-Answering Research. In *Proceedings of the 20th Australasian Document Computing Symposium (ADCS)*, pages 3–9. ACM, 2015.

Eduard Hovy and Chin-Yew Lin. Automated Text Summarization and the SUMMARIST System. In *Proceedings of the workshop on TIPSTER held at Baltimore, Maryland: October 13-15, 1998*, pages 197–214. ACL, 1998.

Eduard Hovy, Laurie Gerber, Ulf Hermjakob, Chin-Yew Lin, and Deepak Ravichandran. Toward Semantics-based Answer Pinpointing. In *Proceedings of the 2001 Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1–7. ACL, 2001.

Wei-Ning Hsu, Yu Zhang, and James Glass. Recurrent Neural Network Encoder with Attention for Community Question Answering. *CoRR*, arXiv preprint arXiv:1603.07044, 2016.

Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme Learning Machine: Theory and Applications. *Neurocomputing*, 70(1):489–501, 2006.

Jizhou Huang, Ming Zhou, and Dan Yang. Extracting Chatbot Knowledge from Online Discussion Forums. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 423–428. Morgan Kaufmann Publishers, 2007.

Zhiheng Huang, Marcus Thint, and Zengchang Qin. Question Classification Using Head Words and Their Hypernyms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 927–936. ACL, 2008.

Rodney Huddleston. *English Grammar: An Outline.* Cambridge University Press, 1988.

Daisuke Ishikawa, Tetsuya Sakai, and Noriko Kando. Overview of the NTCIR-8 Community QA Pilot Task (Part I): The Test Collection and the Task. In *Proceedings of the 8th NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization*, pages 421–432. ACM, 2010.

Abraham Ittycheriah, Martin Franz, Wei-Jing Zhu, Adwait Ratnaparkhi, and Richard J Mammone. Question Answering Using Maximum Entropy Components. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1–7. ACL, 2001.

Peter Jansen, Mihai Surdeanu, and Peter Clark. Discourse Complements Lexical Semantics for Non-Factoid Answer Reranking. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 977–986. ACL, 2014.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated Gain-based Evaluation of IR Techniques. *TOIS*, 20(4):422–446, 2002.

Jiwoon Jeon, W Bruce Croft, and Joon Ho Lee. Finding Semantically Similar Questions Based on Their Answers. In *Proceedings of the 28th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 617–618. ACM, 2005a.

Jiwoon Jeon, W Bruce Croft, and Joon Ho Lee. Finding Similar Questions in Large Question and Answer Archives. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 84–90. ACM, 2005b.

Jiwoon Jeon, W Bruce Croft, Joon Ho Lee, and Soyeon Park. A Framework to Predict the Quality of Answers with Non-textual Features. In *Proceedings of the 29th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 228–235. ACM, 2006.

Minwoo Jeong, Chin-Yew Lin, and Gary Geunbae Lee. Semi-Supervised Speech Act Recognition in Emails and Forums. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1250–1259. ACL, 2009.

Zongcheng Ji, Fei Xu, Bin Wang, and Ben He. Question-Answer Topic Model for Question Retrieval in Community Question Answering. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2471–2474. ACM, 2012.

Jian Jiao. *A Framework for Finding and Summarizing Product Defects, and Ranking Helpful Threads from Online Customer Forums Through Machine Learning*. PhD thesis, Virginia Polytechnic Institute and State University, 2013.

Blooma Mohan John, Dion Hoe Lian Goh, Alton Yeow Kuan Chua, and Nilmini Wickramasinghe. Graph-based Cluster Analysis to Identify Similar Questions: A Design Science Approach. *JAIS*, 17(9):590, 2016.

Shafiq Joty, Alberto Barrón-Cedeno, Giovanni Da San Martino, Simone Filice, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. Global Thread-Level Inference for Comment Classification in Community Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 573–578. ACL, 2015.

Pawel Jurczyk and Eugene Agichtein. Discovering Authorities in Question Answer Communities by Using Link Analysis. In *Proceedings of the 16th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 919–922. ACM, 2007a.

Pawel Jurczyk and Eugene Agichtein. Hits on Question Answer Portals: Exploration of Link Analysis for Author Ranking. In *Proceedings of the 30th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 845–846. ACM, 2007b.

Wei-Chen Kao, Duen-Ren Liu, and Shiu-Wen Wang. Expert Finding in Question-Answering Websites: A Novel Hybrid Approach. In *Proceedings of the 25th ACM Symposium on Applied Computing (SAC)*, pages 867–871. ACM, 2010.

Pairin Katerattanakul and Keng Siau. Measuring Information Quality of Web Sites: Development of an Instrument. In *Proceedings of the 20th International Conference on Information Systems*, pages 279–285. AIS, 1999.

Boris Katz, Sue Felshin, Deniz Yuret, Ali Ibrahim, Jimmy Lin, Gregory Marton, Alton Jerome McFarland, and Baris Temelkuran. Omnibase: Uniform Access to Heterogeneous Data for Question Answering. In *Proceedings of the 2002 International Conference on Application of Natural Language to Information Systems*, pages 230–234. Springer, 2002.

Jihie Kim, Grace Chern, Donghui Feng, Erin Shaw, and Eduard Hovy. Mining and Assessing Discussions on the Web Through Speech Act Analysis. In *Proceedings of the 5th International Semantic Web Conference (ISWC), Workshop on Web Content Mining with Human Language Technologies*. Springer, 2006.

Jong Wook Kim, K Selçuk Candan, and Mehmet E Dönderler. Topic Segmentation of Message Hierarchies for Indexing and Navigation Support. In *Proceedings of the 14th International World Wide Web Conference*, pages 322–331. ACM, 2005.

Soojung Kim and Sanghee Oh. Users' Relevance Criteria for Evaluating Answers in a Social Q&A Site. *JASIST*, 60(4):716–727, 2009.

Soojung Kim, Jung Sun Oh, and Sanghee Oh. Best-Answer Selection Criteria in a Social Q&A Site from the User-oriented Relevance Perspective. *JASIST*, 44(1):1–15, 2007.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. Classifying Dialogue Acts in One-on-one Live Chats. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 862–871. ACL, 2010a.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. Classifying Dialogue Acts in One-on-one Live Chats. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 862–871. ACL, 2010b.

Su Nam Kim, Li Wang, and Timothy Baldwin. Tagging and Linking Web Forum Posts. In *Proceedings of the 14th Conference on Computational Natural Language Learning (CoNLL)*, pages 192–202. ACL, 2010c.

Vanessa Kitzie, Erik Choi, and Chirag Shah. Analyzing Question Quality Through Intersubjectivity: World Views and Objective Assessments of Questions on Social Question-Answering. *JASIST*, 50(1):1–10, 2013.

Mike Klaas. Toward Indicative Discussion Fora Summarization. Technical Report TR-2005-04, UBC CS, 2005.

Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632, 1999.

Philipp Koehn and Kevin Knight. Learning a Translation Lexicon from Monolingual Corpora. In *Proceedings of the ACL-02 Workshop on Unsupervised Lexical Acquisition*, pages 9–16. ACL, 2002.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical Phrase-based Translation. In *Proceedings of the 2003 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 48–54. ACL, 2003.

Giridhar Kumaran and James Allan. Text Classification and Named Entities for New Event Detection. In *Proceedings of the 27th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 297–304. ACM, 2004.

John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the 18th International Conference on Machine Learning (ICML)*, pages 282–289. JMLR, 2001.

Derek Lam. *Exploiting E-mail Structure to Improve Summarization.* PhD thesis, MIT, 2002.

Andrew Lampert, Robert Dale, and Cécile Paris. The Nature of Requests and Commitments in Email Messages. In *Proceedings of the AAAI Workshop on Enhanced Messaging (WS-08-04)*, pages 42–47. AAAI, 2008.

Man Lan, Guoshun Wu, Chunyun Xiao, Yuanbin Wu, and Ju Wu. Building Mutually Beneficial Relationships between Question Retrieval and Answer Ranking to Improve Performance of Community Question Answering. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pages 832–839. IEEE, 2016.

Mirella Lapata. Automatic Evaluation of Information Ordering: Kendall's Tau. *Computational Linguistics*, 32(4):471–484, 2006.

Thomas D LaToza and Brad A Myers. Hard-to-Answer Questions about Code. In *Evaluation and Usability of Programming Languages and Tools*, page 8. ACM, 2010.

Jey Han Lau and Timothy Baldwin. An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation. In *Proceedings of the 1st Workshop on Representation Learning for NLP (RepL4NLP)*, pages 78–86. ACL, 2016.

Long T Le, Chirag Shah, and Erik Choi. Evaluating the Quality of Educational Answers in Community Question-Answering. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries (JCDL)*, pages 129–138. ACM, 2016.

Quoc V Le and Tomas Mikolov. Distributed Representations of Sentences and Documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pages 1188–1196. JMLR, 2014.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

JunChoi Lee and Yu-N Cheah. Semantic Relatedness Measure for Identifying Relevant Answers in Online Community Question Answering Services. In *Proceedings of the 9th International Conference on IT in Asia (CITA)*. IEEE, 2015.

Jung-Tae Lee, Sang-Bum Kim, Young-In Song, and Hae-Chang Rim. Bridging Lexical Gaps Between Queries and Questions on Large Online Q&A Collections with Compact Translation Models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 410–418. ACL, 2008.

Tao Lei, Hrishikesh Joshi, Regina Barzilay, Tommi Jaakkola, Katerina Tymoshenko, Alessandro Moschitti, and Lluis Marquez. Denoising Bodies to Titles: Retrieving Similar Questions with Recurrent Convolutional Models. *CoRR*, abs/1512.05726, 2015.

Tao Lei, Hrishikesh Joshi, Regina Barzilay, Tommi S Jaakkola, Kateryna Tymoshenko, Alessandro Moschitti, and Lluıs Màrquez. Semi-Supervised Question Retrieval with Gated Convolutions. In *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1279–1289. ACL, 2016.

Oliver Lemon, Alex Gruenstein, and Stanley Peters. Collaborative Activities and Multi-Tasking in Dialogue Systems. *Traitement Automatique des Langues (TAL), Special Issue on Dialogue*, 43(2):131–154, 2002.

Baichuan Li. *A Computational Framework for Question Processing in Community Question Answering Services.* PhD thesis, Chinese University of Hong Kong, 2014.

Baichuan Li and Irwin King. Routing Questions to Appropriate Answerers in Community Question Answering Services. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1585–1588. ACM, 2010.

Baichuan Li, Irwin King, and Michael R Lyu. Question Routing in Community Question Answering: Putting Category in its Place. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2041–2044. ACM, 2011.

Baichuan Li, Tan Jin, Michael R Lyu, Irwin King, and Barley Mak. Analyzing and Predicting Question Quality in Community Question Answering Services. In *Proceedings of the 21st International World Wide Web Conference*, pages 775–782. ACM, 2012.

Baoli Li, Yandong Liu, and Eugene Agichtein. CoCQA: Co-training over Questions and Answers with an Application to Predicting Question Subjectivity Orientation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 937–946. ACL, 2008a.

Baoli Li, Yandong Liu, Ashwin Ram, Ernest V Garcia, and Eugene Agichtein. Exploring Question Subjectivity Prediction in Community QA. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 735–736. ACM, 2008b.

Shuguang Li and Suresh Manandhar. Improving Question Recommendation by Exploiting Information Need. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)-hlt*, volume 1, pages 1425–1434. ACL, 2011.

Xin Li and Dan Roth. Learning Question Classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics*, volume 1, pages 1–7. ACL, 2002.

Yiyang Li, Lei Su, Jun Chen, and Liwei Yuan. Semi-Supervised Learning for Question Classification in CQA. *Natural Computing*, pages 1–11, 2016.

Chen Lin, Jiang-Ming Yang, Rui Cai, Xin-Jing Wang, Wei Wang, and Lei Zhang. Modeling Semantics and Structure of Discussion Threads. In *Proceedings of the 18th International World Wide Web Conference*, pages 1103–1104. ACM, 2009.

Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Proceedings of the Text Summarization Branches Out Workshop*, pages 74–81. ACL, 2004.

Chin-Yew Lin and Eduard Hovy. From Single to Multi-Document Summarization: A Prototype System and its Evaluation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 457–464. ACL, 2002.

Fei Liu, Alistair Moffat, Timothy Baldwin, and Xiuzhen Zhang. Quit While Ahead: Evaluating Truncated Rankings. In *Proceedings of the 39th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 953–956. ACM, 2016.

Qiaoling Liu and Eugene Agichtein. Modeling Answerer Behavior in Collaborative Question Answering Systems. In *Proceedings of the 33rd Annual European Conference on Information Retrieval Research (ECIR): Advances in Information Retrieval*, pages 67–79. Springer, 2011.

Qiaoling Liu, Eugene Agichtein, Gideon Dror, Evgeniy Gabrilovich, Yoelle Maarek, Dan Pelleg, and Idan Szpektor. Predicting Web Searcher Satisfaction with Existing Community-based Answers. In *Proceedings of the 34th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 415–424. ACM, 2011.

Qiaoling Liu, Eugene Agichtein, Gideon Dror, Yoelle Maarek, and Idan Szpektor. When Web Search Fails, Searchers Become Askers: Understanding the Transition. In *Proceedings of the 35th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 801–810. ACM, 2012.

Xiaoyong Liu, W Bruce Croft, and Matthew Koll. Finding Experts in Community-Based Question-Answering Services. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 315–316. ACM, 2005.

Yandong Liu and Eugene Agichtein. You've Got Answers: Towards Personalized Models for Predicting Success in Community Question Answering. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL): Human Language Technologies*, pages 97–100. ACL, 2008a.

Yandong Liu and Eugene Agichtein. On the Evolution of the Yahoo! Answers QA Community. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 737–738. ACM, 2008b.

Yandong Liu, Jiang Bian, and Eugene Agichtein. Predicting Information Seeker Satisfaction in Community Question Answering. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 483–490. ACM, 2008a.

Yuanjie Liu, Shasha Li, Yunbo Cao, Chin-Yew Lin, Dingyi Han, and Yong Yu. Understanding and Summarizing Answers in Community-based Question Answering Services. In *Proceedings of the 22nd International Conference on Computational Linguistics (COLING)*, pages 497–504. ACL, 2008b.

Zhe Liu and Bernard J. Jansen. Identifying and Predicting the Desire to Help in Social Question and Answering. *Information Processing & Management*, 53(2):490–504, 2016.

Ziming Liu and Xiaobin Huang. Evaluating the Credibility of Scholarly Information on the Web: A Cross Cultural Study. *The International Information & Library Review*, 37(2):99–106, 2005.

Byron Long and Ronald Baecker. A Taxonomy of Internet Communication Tools. In *Proceedings of the 1997 World Conference on the WWW, Internet & Intranet (WebNet)*, pages 1–15. Association for the Advancement of Computing in Education (AACE), 1997.

Vanessa Lopez, Michele Pasin, and Enrico Motta. Aqualog: An Ontology-Portable Question Answering System for the Semantic Web. In *Proceedings of the European Semantic Web Conference (ESWC)*, pages 546–562. Springer, 2005.

Jie Lou, Kai Hin Lim, Yulin Fang, and Jerry Zeyu Peng. Drivers Of Knowledge Contribution Quality And Quantity In Online Question And Answering Communities. In *Proceedings of the Pacific Asia Conference on Information Systems (PACIS)*, page 121. AIS, 2011.

Jie Lou, Yulin Fang, Kai H Lim, and Jerry Zeyu Peng. Contributing High Quantity and Quality Knowledge to Online Q&A Communities. *JASIST*, 64(2):356–371, 2013.

Pamela J Ludford, Dan Cosley, Dan Frankowski, and Loren Terveen. Think Different: Increasing Online Community Participation Using Uniqueness and Group Dissimilarity. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 631–638. ACM, 2004.

Marco Lui and Timothy Baldwin. You Are What You Post: User-level Features in Threaded Discourse. In *Proceedings of the 14th Australasian Document Computing Symposium (ADCS)*, pages 98–105. ACM, 2009.

Marco Lui and Timothy Baldwin. Classifying User Forum Participants: Separating the Gurus from the Hacks, and Other Tales of the Internet. In *Proceedings of the 2010 Australasian Language Technology Association Workshop (ALTA)*, pages 49–57. ACL, 2010.

Steven Lytinen and Noriko Tomuro. The Use of Question Types to Match Questions in FAQFinder. In *Proceedings of the AAAI Spring Symposium on Mining Answers from Texts and Knowledge Bases (SS-02-06)*, pages 46–53. AAAI, 2002.

Craig Macdonald and Iadh Ounis. Voting Techniques for Expert Search. *Knowledge and information systems*, 16(3):259–280, 2008a.

Craig Macdonald and Iadh Ounis. Key Blog Distillation: Ranking Aggregates. In *Proceedings of the 17th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1043–1052. ACM, 2008b.

Craig Macdonald and Iadh Ounis. Learning Models for Ranking Aggregates. In *Proceedings of the 33rd Annual European Conference on Information Retrieval Research (ECIR): Advances in Information Retrieval*, pages 517–529. Springer, 2011.

Preetham Madeti. Using Apache Spark's MLlib to Predict Closed Questions on Stack Overflow. Master's thesis, Youngstown State University, 2016.

Juha Makkonen, Helena Ahonen-Myka, and Marko Salmenkivi. Simple Semantics in Topic Detection and Tracking. *Information retrieval*, 7(3-4): 347–368, 2004.

Krissada Maleewong. Predicting Quality-Assured Consensual Answers in Community-Based Question Answering Systems. In *Recent Advances in Information and Communication Technology 2016: Proceedings of the 12th International Conference on Computing and Information Technology (IC2IT)*, pages 117–127. Springer, 2016.

Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann. Design Lessons from the Fastest Q&A Site in the West. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2857–2866. ACM, 2011.

Daniel Marcu and William Wong. A Phrase-based, Joint Probability Model for Statistical Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 133–139. ACL, 2002.

Walaa Medhat, Ahmed Hassan, and Hoda Korashy. Sentiment Analysis Algorithms and Applications: A Survey. *Ain Shams Engineering Journal*, 5 (4):1093–1113, 2014.

Donald Metzler and W Bruce Croft. Analysis of Statistical Question Classification for Fact-based Questions. *Information Retrieval*, 8(3):481–504, 2005.

Rada Mihalcea and Paul Tarau. TextRank: Bringing Order into Texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 404–411. ACL, 2004.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *CoRR*, abs/1301.3781, 2013.

Zhao-Yan Ming, Tat-Seng Chua, and Gao Cong. Exploring Domain-specific Term Weight in Archived Question Search. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1605–1608. ACM, 2010.

Alistair Moffat and Justin Zobel. Rank-biased Precision for Measurement of Retrieval Effectiveness. *TOIS*, 27(1):2, 2008.

Piero Molino, Luca Maria Aiello, and Pasquale Lops. Social Question Answering: Textual, User, and Network Features for Best Answer Prediction. *TOIS*, 35(1):4, 2016.

Dana Movshovitz-Attias, Yair Movshovitz-Attias, Peter Steenkiste, and Christos Faloutsos. Analysis of the Reputation System and User Contributions on a Question Answering Website: StackOverflow. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 886–893. IEEE, 2013.

Preslav Nakov, Lluís Màrquez, Walid Magdy, Alessandro Moschitti, Jim Glass, and Bilal Randeree. SemEval-2015 Task 3: Answer Selection in Community Question Answering. In *Proceedings of the 9th Conference on Semantic Evaluation (SemEval)*, pages 269–281. ACL, 2015.

Preslav Nakov, Lluís Màrquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, abed Alhakim Freihat, Jim Glass, and Bilal Randeree. SemEval-2016 Task 3: Community Question Answering. In *Proceedings of the 10th Conference on Semantic Evaluation (SemEval)*, pages 525–545. ACL, 2016.

Preslav Nakov, Doris Hoogeveen, Lluís Màrquez, Alessandro Moschitti, Hamdy Mubarak, Timothy Baldwin, and Karin M Verspoor. SemEval-2017 Task 3: Community Question Answering. In *Proceedings of the 11th Conference on Semantic Evaluation (SemEval)*. ACL, 2017.

Henry Nassif, Mitra Mohtarami, and James Glass. Learning Semantic Relatedness in Community Question Answering Using Neural Models. In *Proceedings of the 1st Workshop on Representation Learning for NLP (RepL4NLP)*, pages 137–147. ACL, 2016.

Ani Nenkova and Amit Bagga. Facilitating Email Thread Access by Extractive Summary Generation. In *Proceedings of the 2003 International Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 287–296. ACL, 2003.

Ani Nenkova, Rebecca Passonneau, and Kathleen McKeown. The Pyramid Method: Incorporating Human Content Selection Variation in Summarization Evaluation. *TSLP*, 4(2):1–13, 2007.

Paula S Newman. Exploring Discussion Lists: Steps and Directions. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 126–134. ACM, 2002.

Paula S. Newman and John C. Blitzer. Summarizing Archived Discussions: A Beginning. In *Proceedings of the 8th International Conference on Intelligent User Interfaces (IUI '03)*, pages 273–276. ACM, 2003.

Lan Nie, Brian D Davison, and Xiaoguang Qi. Topical Link Analysis for Web Search. In *Proceedings of the 29th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 91–98. ACM, 2006.

Yuanping Nie, Jiuming Huang, Zongsheng Xie, Hai Li, Pengfei Zhang, and Yan Jia. NudtMDP at TREC 2015 LiveQA Track. In *Proceedings of the 24th Text REtrieval Conference (TREC) (LiveQA Track)*. NIST, 2015.

Michael Niemann. *The Duality of Expertise: Identifying Expertise Claims and Community Opinions within Online Forum Dialogue*. PhD thesis, Monash University, 2015.

Blair Nonnecke and Jenny Preece. Lurker Demographics: Counting the Silent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 73–80. ACM, 2000.

Adekunle Isiaka Obasa, Naomie Salim, and Atif Khan. Hybridization of Bag-of-Words and Forum Metadata for Web Forum Question Post Detection. *Indian Journal of Science and Technology*, 8(32):1–12, 2016.

Franz Josef Och, Christoph Tillmann, Hermann Ney, et al. Improved Alignment Models for Statistical Machine Translation. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28. ACL, 1999.

Adi Omari, David Carmel, Oleg Rokhlenko, and Idan Szpektor. Novelty Based Ranking of Human Answers for Community Questions. In *Proceedings of the 39th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 215–224. ACM, 2016.

Daniel F.O. Onah, Jane E. Sinclair, and Russell Boyatt. Exploring the Use of MOOC Discussion Forums. In *Proceedings of the London International Conference on Education*, pages 1–4. Infonomics Society, 2014.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical Report 1999-66, Stanford InfoLab, 1999.

Aditya Pal and Joseph A Konstan. Expert Identification in Community Question Answering: Exploring Question Selection Bias. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1505–1508. ACM, 2010.

Aditya Pal, Shuo Chang, and Joseph A Konstan. Evolution of Experts in Question Answering Communities. In *Proceedings of the 6e AAAI International Conference on Weblogs and Social Media (ICWSM)*, pages 274–281. AAAI, 2012a.

Aditya Pal, F Maxwell Harper, and Joseph A Konstan. Exploring Question Selection Bias to Identify Experts and Potential Experts in Community Question Answering. *TOIS*, 30(2):10, 2012b.

C Pechsiri and R Piriyakul. Developing a WhyâĂŞHow Question Answering System on Community Web Boards with a Causality Graph Including Procedural Knowledge. *Information Processing in Agriculture*, 3(1):36–53, 2016.

Dan Pelleg and Andrew W Moore. X-means: Extending K-means with Efficient Estimation of the Number of Clusters. In *Proceedings of the 17th International Conference on Machine Learning (ICML)*, pages 727–734. JMLR, 2000.

Anselmo Peñas and Alvaro Rodrigo. A Simple Measure to Assess Non-Response. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL): Human Language Technologies*, pages 1415–1424. ACL, 2011.

Florent Perronnin and Christopher Dance. Fisher Kernels on Visual Vocabularies for Image Categorization. In *Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

Boaz Petersil, Avihai Mejer, Idan Szpektor, and Koby Crammer. That's not my Question: Learning to Weight Unmatched Terms in CQA Vertical Search. In *Proceedings of the 39th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 225–234. ACM, 2016.

Yuval Pinter, Roi Reichart, and Idan Szpektor. Syntactic Parsing of Web Queries with Question Intent. *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 670–680, 2016.

Jay M Ponte. *A Language Modeling Approach to Information Retrieval.* PhD thesis, University of Massachusetts Amherst, 1998.

Jay M Ponte and W Bruce Croft. A Language Modeling Approach to Information Retrieval. In *Proceedings of the 21st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 275–281. ACM, 1998.

Luca Ponzanelli, Andrea Mocci, Alberto Bacchelli, and Michele Lanza. Understanding and Classifying the Quality of Technical Forum Questions. In *Proceedings of the 14th International Conference on Quality Software*, pages 343–352. IEEE, 2014.

Ana-Maria Popescu, Oren Etzioni, and Henry Kautz. Towards a Theory of Natural Language Interfaces to Databases. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, pages 149–157. ACM, 2003.

Jagat Sastry Pudipeddi, Leman Akoglu, and Hanghang Tong. User Churn in Focused Question Answering Sites: Characterizations and Prediction. In *Proceedings of the 23rd International World Wide Web Conference*, pages 469–474. ACM, 2014.

Minghui Qiu and Jing Jiang. A Latent Variable Model for Viewpoint Discovery from Threaded Forum Posts. In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1031–1040. ACL, 2013.

Xipeng Qiu and Xuanjing Huang. Convolutional Neural Tensor Network Architecture for Community-based Question Answering. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1305–1311. AAAI, 2015.

Bo Qu, Gao Cong, Cuiping Li, Aixin Sun, and Hong Chen. An Evaluation of Classification Models for Question Topic Categorization. *JASIST*, 63(5): 889–903, 2012.

Mingcheng Qu, Guang Qiu, Xiaofei He, Cheng Zhang, Hao Wu, Jiajun Bu, and Chun Chen. Probabilistic Question Recommendation for Question Answering Communities. In *Proceedings of the 18th International World Wide Web Conference*, pages 1229–1230. ACM, 2009.

Daphne Ruth Raban. The Incentive Structure in an Online Information Market. *JASIST*, 59(14):2284–2295, 2008.

Daphne Ruth Raban. Self-Presentation and the Value of Information in Q&A Websites. *JASIST*, 60(12):2465–2473, 2009.

Davood Rafiei, Krishna Bharat, and Anand Shukla. Diversifying Web Search Results. In *Proceedings of the 19th International World Wide Web Conference*, pages 781–790. ACM, 2010.

Preethi Raghavan, Rose Catherine, Shajith Ikbal, Nanda Kambhatla, and Debapriyo Majumdar. Extracting Problem and Resolution Information from Online Discussion Forums. In *Proceedings of the 16th International Conference on Management of Data (COMAD)*, pages 77–88. Computer Society of India, 2010.

Owen Rambow, Lokesh Shrestha, John Chen, and Chirsty Lauridsen. Summarizing Email Threads. In *Proceedings of the 2004 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 105–108. ACL, 2004.

Marc'Aurelio Ranzato, Christopher Poultney, Sumit Chopra, and Yann LeCun. Efficient Learning of Sparse Representations with an Energy-based Model. *NIPS*, pages 1137–1144, 2007.

Zhaochun Ren, Hongya Song, Piji Li, Shangsong Liang, Jun Ma, and Maarten de Rijke. Using Sparse Coding for Answer Summarization in Non-Factoid Community Question-Answering. In *Proceedings of the 2016 SIGIR WebQA Workshop*. ACM, 2016.

Fatemeh Riahi, Zainab Zolaktaf, Mahdi Shafiei, and Evangelos Milios. Finding Expert Users in Community Question Answering. In *Proceedings of the 21st International World Wide Web Conference*, pages 791–798. ACM, 2012.

Soo Young Rieh. Judgment of Information Quality and Cognitive Authority in the Web. *JASIST*, 53(2):145–161, 2002.

Soo Young Rieh and Nicholas J Belkin. Understanding Judgment of Information Quality and Cognitive Authority in the WWW. In *Proceedings of the 61st Annual Meeting of the American Society for Information Science*, volume 35, pages 279–289. ASIS&T, 1998.

R. Rienks. *Meetings in Smart Environments: Implications of Progressing Technology*. PhD thesis, University of Twente, 2007.

Stephen Robertson, Hugo Zaragoza, and Michael Taylor. Simple BM25 Extension to Multiple Weighted Fields. In *Proceedings of the 13th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 42–49. ACM, 2004.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference (TREC)*, pages 109–126. NIST, 1994.

Anthony C. Robinson. Exploring Class Discussions from a Massive Open Online Course (MOOC) on Cartography. *Modern Trends in Cartography*, pages 173–182, 2015.

Carolyn P. Rosé, Barbara S. Di Eugenio, Lori Levin, and Carol Van Ess-Dykema. Discourse Processing of Dialogues with Multiple Threads. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 31–38. ACL, 1995.

Daniel E Rose and Danny Levinson. Understanding User Goals in Web Search. In *Proceedings of the 13th International World Wide Web Conference*, pages 13–19. ACM, 2004.

Lorenzo A Rossi and Omprakash Gnawali. Language Independent Analysis and Classification of Discussion Threads in Coursera MOOC Forums. In *Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IRI)*, pages 654–661. IEEE, 2014.

Ripon K Saha, Avigit K Saha, and Dewayne E Perry. Toward Understanding the Causes of Unanswered Questions in Software Information Sites: A Case Study of Stack Overflow. In *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*, pages 663–666. ACM, 2013.

Tirath Prasad Sahu, Naresh Nagwani, and Shrish Verma. Multivariate Beta Mixture Model for Automatic Identification of Topical Authoritative Users in Community Question Answering Sites. *IEEE Access*, 4:5343–5355, 2016a.

Tirath Prasad Sahu, Naresh Kumar Nagwani, and Shrish Verma. TagLDA based User Persona Model to Identify Topical Experts for Newly Posted Questions in Community Question Answering Sites. *International Journal of Applied Engineering Research*, 11(10):7072–7078, 2016b.

Tirath Prasad Sahu, Naresh Kumar Nagwani, and Shrish Verma. Topical Authoritative Answerer Identification on Q&A Posts using Supervised Learning in CQA Sites. In *Proceedings of the 9th Annual ACM India Conference*, pages 129–132. ACM, 2016c.

Tetsuya Sakai, Daisuke Ishikawa, Noriko Kando, Yohei Seki, Kazuko Kuriyama, and Chin-Yew Lin. Using Graded-Relevance Metrics for Evaluating Community QA Answer Selection. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 187–196. ACM, 2011.

Daniel Schall and Florian Skopik. An Analysis of the Structure and Dynamics of Large-scale Q/A Communities. In *Proceedings of the East European Conference on Advances in Databases and Information Systems*, pages 285–301. Springer, 2011.

Kim Schouten and Flavius Frasincar. Finding Implicit Features in Consumer Reviews for Sentiment Analysis. In *Proceedings of the 2014 International Conference on Web Engineering*, pages 130–144. Springer, 2014.

Anne Schuth, Maarten Marx, and Maarten de Rijke. Extracting the Discussion Structure in Comments on News-Articles. In *Proceedings of the 9th Annual ACM International Workshop on Web Information and Data Management*, pages 97–104. ACM, 2007.

John R. Searle. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, 1969.

Jangwon Seo and W Bruce Croft. Blog Site Search Using Resource Selection. In *Proceedings of the 17th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1053–1062. ACM, 2008.

Jangwon Seo, W. Bruce Croft, and David A. Smith. Online Community Search Using Thread Structure. In *Proceedings of the 18th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1907–1910. ACM, 2009.

Jangwon Seo, W Bruce Croft, and David A Smith. Online Community Search Using Conversational Structures. *Information Retrieval*, 14(6):547–571, 2011.

Aliaksei Severyn and Alessandro Moschitti. Learning to Rank Short Text Pairs with Convolutional Deep Neural Networks. In *Proceedings of the 38th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 373–382. ACM, 2015.

Pnina Shachaf. Answer Reliability on Q&A Sites. In *Proceedings of the Americas Conference on Information Systems (AMCIS)*, page 376. AIS, 2010.

Pnina Shachaf. A Comparative Assessment of Answer Quality on Four Question Answering Sites. *JIS*, 37(5):476–486, 2011.

Giovanni Da Shafiq Joty, Lluís Màrquez, and Preslav Nakov. Joint Learning with Global Inference for Comment Classification in Community Question Answering. In *Proceedings of the 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 703–713. ACL, 2016.

Chirag Shah. Building a Parsimonious Model for Identifying Best Answers Using Interaction History in Community Q&A. *JASIST*, 52(1):1–10, 2015.

Chirag Shah and Vanessa Kitzie. Social Q&A and Virtual Reference âĂŤ Comparing Apples and Oranges with the Help of Experts and Users. *JASIST*, 63(10):2020–2036, 2012.

Chirag Shah and Jefferey Pomerantz. Evaluating and Predicting Answer Quality in Community QA. In *Proceedings of the 33rd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 411–418. ACM, 2010.

Chirag Shah, Sanghee Oh, and Jung Sun Oh. Research Agenda for Social Q&A. *Library & Information Science Research*, 31(4):205–209, 2009.

Chirag Shah, Marie L Radford, Lynn Silipigni Connaway, Erik Choi, and Vanessa Kitzie. “How much change do you get from $40” - Analyzing and Addressing Failed Questions on Social Q&A. *JASIST*, 49(1):1–10, 2012.

Chirag Shah, Vanessa Kitzie, and Erik Choi. Modalities, Motivations, and Materials – Investigating Traditional and Social Online Q&A Services. *JIS*, pages 1–19, 2014.

Rebecca Sharp, Peter Jansen, Mihai Surdeanu, and Peter Clark. Spinning Straw into Gold: Using Free Text to Train Monolingual Alignment Models for Non-Factoid Question Answering. In *Proceedings of the 2015 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 231–237. ACL, 2015.

Libin Shen and Aravind K Joshi. Ranking and Reranking with Perceptron. *Machine Learning*, 60(1-3):73–96, 2005.

Yikang Shen, Wenge Rong, Nan Jiang, Baolin Peng, Jie Tang, and Zhang Xiong. Word Embedding based Correlation Model for Question/Answer Matching. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 3511–3517. AAAI, 2015a.

Yikang Shen, Wenge Rong, Zhiwei Sun, Yuanxin Ouyang, and Zhang Xiong. Question/answer matching for cqa system via combining lexical and sequential information. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, pages 275–281. AAAI, 2015b.

Lokesh Shrestha and Kathleen McKeown. Detection of Question-Answer Pairs in Email Conversations. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 889–895. ACL, 2004.

Anna Shtok, Gideon Dror, Yoelle Maarek, and Idan Szpektor. Learning from the Past: Answering New Questions with Past Answers. In *Proceedings of the 21st International World Wide Web Conference*, pages 759–768. ACM, 2012.

Joao Silva, Luísa Coheur, Ana Cristina Mendes, and Andreas Wichert. From Symbolic to Sub-Symbolic Information in Question Classification. *Artificial Intelligence Review*, 35(2):137–154, 2011.

Amit Singh. Entity Based Q&A Retrieval. In *Proceedings of the Joint Meeting of the Conference on Empirical Methods in Natural Language Processing (EMNLP) and the Conference on Computational Natural Language Learning (CoNLL)*, pages 1266–1277. ACL, 2012.

Amit Singh, Dinesh Raghu, et al. Retrieving Similar Discussion Forum Threads: A Structure Based Approach. In *Proceedings of the 35th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 135–144. ACM, 2012.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning with Neural Tensor Networks for Knowledge Base Completion. *NIPS*, 26:926–934, 2013.

Parikshit Sondhi and ChengXiang Zhai. Mining Semi-Structured Online Knowledge Bases to Answer Natural Language Questions on Community QA Websites. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 341–350. ACM, 2014.

Young-In Song, Chin-Yew Lin, Yunbo Cao, and Hae-Chang Rim. Question Utility: A Novel Static Ranking of Question Search. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pages 1231–1236. AAAI, 2008.

Cleyton Souza, Franck Aragão, José Remígio, Evandro Costa, and Joseana Fechine. Using CQA History to Improve Q&A Experience. In *Proceedings of the 2016 International Conference on Computational Science and Its Applications*, pages 570–580. Springer, 2016.

Ivan Srba. Promoting Sustainability and Transferability of Community Question Answering. *Information Sciences and Technologies Bulletin of the ACM Slovakia*, pages 1–7, 2011.

Ivan Srba and Maria Bielikova. A Comprehensive Survey and Classification of Approaches for Community Question Answering. *TWEB*, 10(3):18, 2016.

Nicola Stokes and Joe Carthy. Combining Semantic and Syntactic Document Classifiers to Improve First Story Detection. In *Proceedings of the 24th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 424–425. ACM, 2001.

Diane M Strong, Yang W Lee, and Richard Y Wang. Data Quality in Context. *Communications of the ACM*, 40(5):103–110, 1997.

Qi Su, Dmitry Pavlov, Jyh-Herng Chow, and Wendell C Baker. Internet-Scale Collection of Human-Reviewed Data. In *Proceedings of the 16th International World Wide Web Conference*, pages 231–240. ACM, 2007.

Sai Praneeth Suggu, Kushwanth N Goutham, Manoj K Chinnakotla, and Manish Shrivastava. Deep Feature Fusion Network for Answer Quality Prediction in Community Question Answering. In *Proceedings of the Neu-IR 2016 SIGIR Workshop on Neural Information Retrieval.* arXiv, 2016.

Ke Sun, Yunbo Cao, Xinying Song, Young-In Song, Xiaolong Wang, and Chin-Yew Lin. Learning to Recommend Questions Based on User Ratings. In *Proceedings of the 18th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 751–758. ACM, 2009.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. Learning to Rank Answers on Large Online QA Collections. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics (ACL): Human Language Technologies*, pages 719–727. ACL, 2008.

Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. Learning to Rank Answers to Non-Factoid Questions from Web Collections. *Computational Linguistics*, 37(2):351–383, 2011.

Maggy Anastasia Suryanto, Ee Peng Lim, Aixin Sun, and Roger HL Chiang. Quality-Aware Collaborative Question Answering: Methods and Evaluation. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining (WSDM)*, pages 142–151. ACM, 2009.

Jun Suzuki, Hirotoshi Taira, Yutaka Sasaki, and Eisaku Maeda. Question Classification using HDAG Kernel. In *Proceedings of the ACL 2003 workshop on Multilingual summarization and question answering*, volume 12, pages 61–68. ACL, 2003.

Saori Suzuki, Shin'ichi Nakayama, and Hideo Joho. Formulating Effective Questions for Community-based Question Answering. In *Proceedings of the 34th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 1261–1262. ACM, 2011.

Ming Tan, Bing Xiang, and Bowen Zhou. LSTM-based Deep Learning Models for Non-Factoid Answer Selection. In *Proceedings of the 2016 International Conference on Learning Representations (ICLR) Workshop Track*. arXiv, 2016.

Jaime Teevan, Susan T Dumais, and Daniel J Liebling. To Personalize or not to Personalize: Modeling Queries with Variation in User Intent. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 163–170. ACM, 2008.

Qiongjie Tian and Baoxin Li. Weakly Hierarchical Lasso based Learning to Rank in Best Answer Prediction. In *Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 307–314. IEEE, 2016.

Qiongjie Tian, Peng Zhang, and Baoxin Li. Towards Predicting the Best Answers in Community-based Question-Answering Services. In *Proceedings of the 7th AAAI International Conference on Weblogs and Social Media (ICWSM)*, pages 725–728. AAAI, 2013a.

Yuan Tian, Pavneet Singh Kochhar, Ee-Peng Lim, Feida Zhu, and David Lo. Predicting Best Answerers for New Questions: An Approach Leveraging Topic Modeling and Collaborative Voting. In *Proceedings of the 5th International Conference on Social Informatics (SocInfo), International Workshops*, pages 55–68. Springer, 2013b.

Almer S Tigelaar, Rieks op den Akker, and Djoerd Hiemstra. Automatic Summarisation of Discussion Fora. *Natural Language Engineering*, 16(02): 161–192, 2010.

Mattia Tomasoni. Metadata-aware Measures for Answer Summarization in Community Question Answering. Master's thesis, University of Uppsala, Sweden, 2003.

Mattia Tomasoni and Minlie Huang. Metadata-aware Measures for Answer Summarization in Community Question Answering. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 760–769. ACL, 2010.

Noriko Tomuro. Question Terminology and Representation for Question Type Classification. In *Proceedings of the Second International Workshop on Computational Terminology (COMPUTERM 2002)*, pages 1–7. ACL, 2002.

Noriko Tomuro and Steven L Lytinen. Selecting Features for Paraphrasing Question Sentences. In *Proceedings of the Workshop on Automatic Paraphrasing at Natural Language Processing Pacific Rim Symposium (NLPRS)*, pages 55–62. National Electronics and Computer Technology Center (NECTC), 2001.

Quan Hung Tran, Vu Tran, Tu Vu, Minh Nguyen, and Son Bao Pham. JAIST: Combining Multiple Features for Answer Selection in Community Question Answering. In *Proceedings of the 9th Conference on Semantic Evaluation (SemEval)*, pages 215–219. ACL, 2015.

Christoph Treude, Ohad Barzilay, and Margaret-Anne Storey. How do Programmers Ask and Answer Questions on the Web? (Nier Track). In *Proceedings of the 33rd International Conference on Software Engineering (ICSE)*, pages 804–807. IEEE, 2011.

Xudong Tu, Xin-Jing Wang, Dan Feng, and Lei Zhang. Ranking Community Answers via Analogical Reasoning. In *Proceedings of the 18th International World Wide Web Conference*, pages 1227–1228. ACM, 2009.

Kateryna Tymoshenko, Daniele Bonadiman, and Alessandro Moschitti. Learning to Rank Non-Factoid Answers: Comment Selection in Web Forums. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2049–2052. ACM, 2016.

Jan Ulrich. *Supervised Machine Learning for Email Thread Summarization*. PhD thesis, University of British Columbia, 2008.

David Vallet and Pablo Castells. Personalized Diversification of Search Results. In *Proceedings of the 35th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 841–850. ACM, 2012.

Jelica Vasiljevic, Tom Lampert, and Milos Ivanovic. The Application of the Topic Modeling to Question Answer Retrieval. In *Proceedings of the 6th International Conference of Information Society and Technology (ICIST)*, volume 1, pages 241–246. ICIST, 2016.

Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and Composing Robust Features with Denoising Autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1096–1103. ACM, 2008.

Ellen M Voorhees et al. The TREC-8 Question Answering Track Report. In *Proceedings of the 8th Text REtrieval Conference (TREC)*, pages 77–82. NIST, 1999.

Stephen Wan and Kathy McKeown. Generating Overview Summaries of Ongoing Email Thread Discussions. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING)*, pages 549–555. ACL, 2004.

Nayer Wanas, Motaz El-Saban, Heba Ashour, and Waleed Ammar. Automatic Scoring of Online Discussion Posts. In *Proceedings of the 2nd ACM Workshop on Information Credibility on the Web (WICOW'08)*, pages 19–26. ACM, 2008.

Baoxun Wang, Bingquan Liu, Chengjie Sun, Xiaolong Wang, and Bo Li. Adaptive Maximum Marginal Relevance Based Multi-Email Summarization. In *Proceedings of the 2009 International Conference on Artificial Intelligence and Computational Intelligence*, pages 417–424. Springer, 2009a.

Baoxun Wang, Bingquan Liu, Chengjie Sun, Xiaolong Wang, and Lin Sun. Extracting Chinese Question-Answer Pairs from Online Forums. In *Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics (SMC'09)*, pages 1159–1164. IEEE, 2009b.

Baoxun Wang, Xiaolong Wang, Chengjie Sun, Bingquan Liu, and Lin Sun. Modeling Semantic Relevance for Question-Answer Pairs in Web Social Communities. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1230–1238. ACL, 2010a.

Di Wang and Eric Nyberg. A Long Short-term Memory Model for Answer Sentence Selection in Question Answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, pages 707–712. ACL, 2015a.

Di Wang and Eric Nyberg. CMU OAQA at TREC 2015 LiveQA: Discovering the Right Answer with Clues. In *Proceedings of the 24th Text REtrieval Conference (TREC) (LiveQA Track)*, pages 1–6. NIST, 2015b.

G Alan Wang, Jian Jiao, and Weiguo Fan. Searching for Authoritative Documents in Knowledge-base Communities. *Proceedings of the 2009 International Conference on Information Systems (ICIS)*, page 109, 2009c.

G Alan Wang, Jian Jiao, Alan S Abrahams, Weiguo Fan, and Zhongju Zhang. ExpertRank: A Topic-aware Expert Finding Algorithm for Online Knowledge Communities. *Decision Support Systems*, 54(3):1442–1451, 2013a.

Hongning Wang, Chi Wang, ChengXiang Zhai, and Jiawei Han. Learning Online Discussion Structures by Conditional Random Fields. In *Proceedings of the 34th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 435–444. ACM, 2011a.

Jian Wang, Jiqing Sun, Hongfei Lin, Hualei Dong, and Shaowu Zhang. Predicting Best Answerers for New Questions: An Approach Leveraging Convolution Neural Networks in Community Question Answering. In *Proceedings of the 2016 Chinese National Conference on Social Media Processing*, pages 29–41. Springer, 2016.

Kai Wang, Zhaoyan Ming, and Tat-Seng Chua. A Syntactic Tree Matching Approach to Finding Similar Questions in Community-based QA Services. In *Proceedings of the 32nd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 187–194. ACM, 2009d.

Kai Wang, Zhao-Yan Ming, Xia Hu, and Tat-Seng Chua. Segmentation of Multi-Sentence Questions: Towards Effective Question Retrieval in cQA Services. In *Proceedings of the 33rd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 387–394. ACM, 2010b.

Li Wang, Su Nam Kim, and Timothy Baldwin. Thread-level Analysis over Technical User Forum Data. In *Proceedings of the 2010 Australasian Language Technology Association Workshop (ALTA)*, pages 27–31. ACL, 2010c.

Li Wang, Su Nam Kim, and Timothy Baldwin. The Utility of Discourse Structure in Identifying Resolved Threads in Technical User Forums. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 2739–2756. ACL, 2012.

Li Wang, Su Nam Kim, and Timothy Baldwin. The Utility of Discourse Structure in Forum Thread Retrieval. In *Proceedings of the 9th Asian Information Retrieval Societies Conference (AIRS 2013)*, pages 284–295. Springer, 2013b.

Richard Y Wang and Diane M Strong. Beyond Accuracy: What Data Quality Means to Data Consumers. *Journal of management information systems*, 12(4):5–33, 1996.

Wei Wang, Baichuan Li, and Irwin King. Improving Question Retrieval in Community Question Answering with Label Ranking. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pages 349–356. IEEE, 2011b.

Xin-Jing Wang, Xudong Tu, Dan Feng, and Lei Zhang. Ranking Community Answers by Modeling Question-Answer Relationships via Analogical Reasoning. In *Proceedings of the 32nd International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 179–186. ACM, 2009e.

Yi-Chia Wang, Mahesh Joshi, and Carolyn P Rosé. A Feature Based Approach to Leveraging Context for Classifying Newsgroup Style Discussion Segments. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL) Companion Volume Proceedings of the Demo and Poster Sessions*, pages 73–76. ACL, 2007.

Yida Wang, Jiang-Ming Yang, Wei Lai, Rui Cai, Lei Zhang, and Wei-Ying Ma. Exploring Traversal Strategy for Web Forum Crawling. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 459–466. ACM, 2008.

Yu Wang and Eugene Agichtein. Query Ambiguity Revisited: Clickthrough Measures for Distinguishing Informational and Ambiguous Queries. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 361–364. ACL, 2010.

Zhe Wang and Pengyi Zhang. Examining User Roles in Social Q&A: the Case of Health Topics in Zhihu.com. In *Proceedings of the 2016 Annual Meeting of the Association of Information Science and Technology (ASIS&T)*, pages 1–6. Wiley, 2016.

Wei Wei, ZhaoYan Ming, Liqiang Nie, Guohui Li, Jianjun Li, Feida Zhu, Tianfeng Shang, and Changyin Luo. Exploring Heterogeneous Features for Query-focused Summarization of Categorized Community Answers. *Information Sciences*, 330:403–423, 2016.

Markus Weimer and Iryna Gurevych. Predicting the Perceived Quality of Web Forum Posts. In *Proceedings of the 2007 Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 643–648. ACL, 2007.

Markus Weimer, Iryna Gurevych, and Max Mühlhäuser. Automatically Assessing the Post Quality in Online Discussions on Software. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL): Interactive Poster and Demonstration Sessions*, pages 125–128. ACL, 2007.

Howard T Welser, Eric Gleave, Danyel Fisher, and Marc Smith. Visualizing the Signatures of Social Roles in Online Discussion Groups. *Journal of Social Structure (JoSS)*, 8(2):1–32, 2007.

Miaomiao Wen, Diyi Yang, and Carolyn Rosé. Sentiment Analysis in MOOC Discussion Forums: What does it tell us? In *Proceedings of the 7th International Conference on Educational Data Mining (EDM)*, pages 130–137. International Educational Data Mining Society, 2014.

Florian Wolf and Edward Gibson. Representing Discourse Coherence: A Corpus-based Study. *Computational Linguistics*, 31(2):249–287, 2005.

Jian-Syuan Wong, Bart Pursel, Anna Divinsky, and Bernard J Jansen. An Analysis of MOOC Discussion Forum Interactions from the Most Active Users. In *Proceedings of the 2015 International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction (SBP-BRiMS)*, pages 452–457. Springer, 2015.

Guoshun Wu and Man Lan. Leverage Web-based Answer Retrieval and Hierarchical Answer Selection to Improve the Performance of Live Question Answering. In *Proceedings of the 24th Text REtrieval Conference (LiveQA Track)*. NIST, 2015.

Hu Wu, Yongji Wang, and Xiang Cheng. Incremental Probabilistic Latent Semantic Analysis for Automatic Question Recommendation. In *Proceedings of the 2008 ACM Conference on Recommender Systems*, pages 99–106. ACM, 2008.

Wensi Xi, Jesper Lind, and Eric Brill. Learning Effective Ranking Functions for Newsgroup Search. In *Proceedings of the 27th sigir*, pages 394–401. ACM, 2004.

Yang Xianfeng and Liu Pengfei. Question Recommendation and Answer Extraction in Question Answering Community. *International Journal of Database Theory and Application (IJDTA)*, 9(1):35–44, 2016.

Siqi Xiang, Wenge Rong, Yikang Shen, Yuanxin Ouyang, and Zhang Xiong. Multidimensional Scaling Based Knowledge Provision for New Questions in Community Question Answering Systems. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pages 115–122. IEEE, 2016.

Sihong Xie, Qingbo Hu, Weixiang Shao, Jingyuan Zhang, Jing Gao, Wei Fan, and Philip S Yu. Effective Crowd Expertise Modeling via Cross Domain Sparsity and Uncertainty Reduction. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, pages 648–656. SIAM, 2016.

Congfu Xu, Xin Wang, and Yunhui Guo. Collaborative Expert Recommendation for Community-Based Question Answering. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 378–393. Springer, 2016.

Fei Xu, Zongcheng Ji, and Bin Wang. Dual Role Model for Question Recommendation in Community Question Answering. In *Proceedings of the 35th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 771–780. ACM, 2012.

Gu Xu and Wei-Ying Ma. Building Implicit Links from Content for Forum Search. In *Proceedings of the 29th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 300–307. ACM, 2006.

Xiaobing Xue, Jiwoon Jeon, and W Bruce Croft. Retrieval Models for Question and Answer Archives. In *Proceedings of the 31st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 475–482. ACM, 2008.

Baoguo Yang and Suresh Manandhar. Exploring User Expertise and Descriptive Ability in Community Question Answering. In *Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 320–327. IEEE, 2014.

Diyi Yang, Mario Piergallini, Iris Howley, and Carolyn Rose. Forum Thread Recommendation for Massive Open Online Courses. In *Proceedings of the 7th International Conference on Educational Data Mining (EDM)*, pages 257–260. International Educational Data Mining Society, 2014a.

Jiang-Ming Yang, Rui Cai, Yida Wang, Jun Zhu, Lei Zhang, and Wei-Ying Ma. Incorporating Site-Level Knowledge to Extract Structured Data from Web Forums. In *Proceedings of the 18th International World Wide Web Conference*, pages 181–190. ACM, 2009a.

Jie Yang, Ke Tao, Alessandro Bozzon, and Geert-Jan Houben. Sparrows and Owls: Characterisation of Expert Behaviour in StackOverflow. In *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization*, pages 266–277. Springer, 2014b.

Lichun Yang, Shenghua Bao, Qingliang Lin, Xian Wu, Dingyi Han, Zhong Su, and Yong Yu. Analyzing and Predicting Not-Answered Questions in Community-based Question Answering Services. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, pages 1273–1278. AAAI, 2011.

Liu Yang, Minghui Qiu, Swapna Gottipati, Feida Zhu, Jing Jiang, Huiping Sun, and Zhong Chen. CQARank: Jointly Model Topics and Expertise in Community Question Answering. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 99–108. ACM, 2013.

Wen-Yun Yang, Yunbo Cao, and Chin-Yew Lin. A Structural Support Vector Method for Extracting Contexts and Answers of Questions from Online Forums. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 514–523. ACL, 2009b.

Yiming Yang, Tom Pierce, and Jaime Carbonell. A Study of Retrospective and On-line Event Detection. In *Proceedings of the 21st International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 28–36. ACM, 1998.

Yuan Yao, Hanghang Tong, Tao Xie, Leman Akoglu, Feng Xu, and Jian Lu. Want a Good Answer? Ask a Good Question First! *CoRR*, arXiv preprint arXiv:1311.6876, 2013.

Yuan Yao, Hanghang Tong, Tao Xie, Leman Akoglu, Feng Xu, and Jian Lu. Detecting High-Quality Posts in Community Question Answering Sites. *Information Sciences*, 302:70–82, 2015.

David M Zajic, Bonnie J Dorr, and Jimmy Lin. Single-Document and Multi-Document Summarization Techniques for Email Threads Using Sentence Compression. *Information Processing & Management*, 44(4):1600–1610, 2008.

Zhongwu Zhai, Bing Liu, Lei Zhang, Hua Xu, and Peifa Jia. Identifying Evaluative Sentences in Online Discussions. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, pages 933–938. AAAI, 2011.

Dell Zhang and Wee Sun Lee. Question Classification Using Support Vector Machines. In *Proceedings of the 26th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 26–32. ACM, 2003.

Jingyuan Zhang, Xiangnan Kong, Roger Jie Luo, Yi Chang, and Philip S Yu. NCR: A Scalable Network-based Approach to Co-ranking in Question-and-Answer Sites. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 709–718. ACM, 2014a.

Jun Zhang, Mark S Ackerman, and Lada Adamic. Expertise Networks in Online Communities: Structure and Algorithms. In *Proceedings of the 16th International World Wide Web Conference*, pages 221–230. ACM, 2007a.

Kai Zhang, Wei Wu, Haocheng Wu, Zhoujun Li, and Ming Zhou. Question Retrieval with High Quality Answers in Community Question Answering. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 371–380. ACM, 2014b.

Kai Zhang, Wei Wu, Fang Wang, Ming Zhou, and Zhoujun Li. Learning Distributed Representations of Data in Community Question Answering for Question Retrieval. In *Proceedings of the 9th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 533–542. ACM, 2016.

Kuo Zhang, Juan Zi, and Li Gang Wu. New Event Detection Based on Indexing-Tree and Named Entity. In *Proceedings of the 30th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 215–222. ACM, 2007b.

Weinan Zhang, Zhaoyan Ming, Yu Zhang, Liqiang Nie, Ting Liu, and Tat-Seng Chua. The Use of Dependency Relation Graph to Enhance the Term Weighting in Question Retrieval. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 3105–3120. ACL, 2012.

Shiqi Zhao, Haifeng Wang, Chao Li, Ting Liu, and Yi Guan. Automatically Generating Questions from Queries for Community-based Question Answering. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 929–937. ACL, 2011.

Zhou Zhao, Lijun Zhang, Xiaofei He, and Wilfred Ng. Expert Finding for Question Answering via Graph Regularized Matrix Completion. *TKDE*, 27(4):993–1004, 2015.

Guangyou Zhou, Li Cai, Jun Zhao, and Kang Liu. Phrase-Based Translation Model for Question Retrieval in Community Question Answer Archives. In *Proceedings of the 2011 Annual Meeting of the Association for Computational Linguistics (ACL): Human Language Technologies*, pages 653–662. ACL, 2011a.

Guangyou Zhou, Siwei Lai, Kang Liu, and Jun Zhao. Topic-Sensitive Probabilistic Model for Expert Finding in Question Answer Communities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1662–1666. ACM, 2012a.

Guangyou Zhou, Kang Liu, and Jun Zhao. Exploiting Bilingual Translation for Question Retrieval in Community-Based Question Answering. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, pages 3153–3170. ACL, 2012b.

Guangyou Zhou, Kang Liu, and Jun Zhao. Topical Authority Identification in Community Question Answering. In *Proceedings of the Chinese Conference on Pattern Recognition*, pages 622–629. Springer, 2012c.

Guangyou Zhou, Yubo Chen, Daojian Zeng, and Jun Zhao. Towards Faster and Better Retrieval Models for Question Search. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2139–2148. ACM, 2013a.

Guangyou Zhou, Fang Liu, Yang Liu, Shizhu He, and Jun Zhao. Statistical Machine Translation Improves Question Retrieval in Community Question Answering via Matrix Factorization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 852–861. ACL, 2013b.

Guangyou Zhou, Yang Liu, Fang Liu, Daojian Zeng, and Jun Zhao. Improving Question Retrieval in Community Question Answering Using World Knowledge. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2239–2245. AAAI, 2013c.

Guangyou Zhou, Yubo Chen, Daojian Zeng, and Jun Zhao. Group Non-negative Matrix Factorization with Natural Categories for Question Retrieval in Community Question Answer Archives. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, pages 89–98. ACL, 2014.

Guangyou Zhou, Tingting He, Jun Zhao, and Po Hu. Learning Continuous Word Embedding with Metadata for Question Retrieval in Community Question Answering. In *Proceedings of the() Joint 53rd Annual Meeting of the Association for Computational Linguistics (ACL) and the 7th International Joint Conference on Natural Language Processing*, volume Volume 1: Long Papers, pages 250–259. ACL, 2015.

Guangyou Zhou, Zhiwen Xie, Tingting He, Jun Zhao, and X Hu. Learning the Multilingual Translation Representations for Question Retrieval in Community Question Answering via Non-negative Matrix Factorization. *TASLP*, 24(7):1305–1314, 2016a.

Guangyou Zhou, Yin Zhou, Tingting He, and Wensheng Wu. Learning Semantic Representation with Neural Networks for Community Question Answering Retrieval. *KBS*, 93:75–83, 2016b.

Liang Zhou and Eduard Hovy. Digesting Virtual "Geek" Culture: The Summarization of Technical Internet Relay Chats. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 298–305. ACL, 2005.

Liang Zhou and Eduard Hovy. On the Summarization of Dynamically Introduced Information: Online Discussions and Blogs. In *Proceedings of the AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs (SS-06-03)*, pages 237–242. AAAI, 2006.

Shu Zhou and Simon Fong. Exploring the Feature Selection-Based Data Analytics Solutions for Text Mining Online Communities by Investigating the Influential Factors: A Case Study of Programming CQA in Stack Overflow. *Big Data Applications and Use Cases*, pages 49–93, 2016.

Tom Chao Zhou, Chin-Yew Lin, Irwin King, Michael R Lyu, Young-In Song, and Yunbo Cao. Learning to Suggest Questions in Online Forums. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, pages 1298–1303. AAAI, 2011b.

Tom Chao Zhou, Xiance Si, Edward Y Chang, Irwin King, and Michael R Lyu. A Data-Driven Approach to Question Subjectivity Identification in Community Question Answering. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 164–170. AAAI, 2012d.

Yanhong Zhou, Gao Cong, Bin Cui, Christian S Jensen, and Junjie Yao. Routing Questions to the Right Users in Online Communities. In *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE)*, pages 700–711. IEEE, 2009.

Yun Zhou and W Bruce Croft. Query Performance Prediction in Web Search Environments. In *Proceedings of the 30th International Conference on Research and Development in Information Retrieval (SIGIR)*, pages 543–550. ACM, 2007.

Zhi-Min Zhou, Man Lan, Zheng-Yu Niu, and Yue Lu. Exploiting User Profile Information for Answer Ranking in CQA. In *Proceedings of the 21st International World Wide Web Conference*, pages 767–774. ACM, 2012e.

Hengshu Zhu, Huanhuan Cao, Hui Xiong, Enhong Chen, and Jilei Tian. Towards Expert Finding by Leveraging Relevant Categories in Authority Ranking. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2221–2224. ACM, 2011.

Mingliang Zhu, Weiming Hu, and Ou Wu. Topic Detection and Tracking for Threaded Discussion Communities. In *Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01*, pages 77–83. IEEE, 2008.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. *A Multi-Dimensional Model for Assessing the Quality of Answers in Social Q&A Sites.* PhD thesis, Technische Universität Darmstadt, 2009.

Zainab Zolaktaf, Fatemeh Riahi, Mahdi Shafiei, and Evangelos Milios. Modeling Community Question-Answering Archives. In *Proceedings of the 2nd Workshop on Computational Social Science and the Wisdom of Crowds (held at NIPS 2011)*, pages 1–5. MIT Press, 2011.