Modeling Community Question-Answering Archives

Zainab Zolaktaf
Faculty of Computer Science
Dalhousie University
zolaktaf@cs.dal.ca

Fatemeh Riahi
Faculty of Computer Science
Dalhousie University
riahi@cs.dal.ca

Mahdi Shafiei
Faculty of Computer Science
Dalhousie University
shafiei@cs.dal.ca

Evangelos Milios
Faculty of Computer Science
Dalhousie University
eem@cs.dal.ca

Abstract

Community Question Answering (CQA) services contain large archives of previously asked questions and their answers. We present a statistical topic model for modeling Question-Answering archives. The model explicitly captures relationships between questions and their answers by modeling topical dependencies. We show that the model achieves improved performance in retrieving the correct answer for a query question compared to the LDA model. Our model can also be used for automatic tagging of questions and answers. This is useful for providing topical browsing capabilities for legacy Q&A archives.

1 Introduction

Community-based question answering (CQA) services [1] such as LinuxQuestions, Yahoo! Answers and Stackoverflow have recently become very popular. They enable members to ask questions and have them answered by the community. They provide an alternative to traditional web search, and allow users to directly acquire their information needs from other users. These services have the potential of rapidly creating large archives of questions and answers. A considerable portion of their archive can potentially be used as a valuable resource for the information needs of other people. However, one of the main drawbacks of existing CQA services is that the archive information is rarely exploited [2, 5, 6]. The high presence of redundant questions and answers is an indication. Moreover, many legacy question-answering archives lack semantic information necessary for browsing the archive.

The main problem with current search features arises from the characteristic of natural language in which semantically similar content can have different literal representations. Applying document representation techniques that rely on word occurrence will generate different representations for such content. Traditional lexical similarity measures are adequate if sufficient word overlap exists. However, questions and answers on CQAs are typically of a short length and have sparse representations often with little word overlap. The problem is further exacerbated considering the fact that different terminologies are used by users because their knowledge and expertise levels differ. Methods that bridge this vocabulary gap and enhance the representation of the documents by encoding information about their semantic structure are needed.

In this work, we propose a probabilistic topic model for the content of Question-Answering archives. We use the model for the task of Question Answering, in which existing question-answer pairs in the archive are automatically retrieved and ranked given a newly submitted question. In the following, we present a brief summary of our model and report some of our experiments and performance results.

2 Methodology: Question Answering Topic Model

To enhance the representation of questions and answers, and encode information about their semantics we propose a new topic model. Our model builds upon the common assumption in topic models [3] that a document is a mixture of topics, where each topic is defined to be a distribution over words. This assumption is appropriate for data from CQA services because questions are typically assigned multiple tags or topics. Furthermore, it is natural to expect that topics in the answers are influenced by topics in the question. However, subjects raised in answers are typically more technical and specific. This is because the knowledge and expertise of the answerers and askers differs; answerers, who can be regarded as experts on the subjects, are more likely to use terms appropriate for the particular realm of knowledge whereas the askers may use less technical terminology. Answers may also contain additional topics that are correlated to the topics in the question, topics that the asker was unaware of and are not explicitly contained in the question. For instance, given a question about string manipulation, the answer might contain topics such as regular expressions or pattern matching. Additional features relevant to text processing languages such as Python or Perl may also be introduced by the answerer. A simple topic model such as LDA [3] is incapable of modeling the dependencies between topics in the answers and topics in questions and may therefore prove to be ineffective for such a setting. The aforementioned aspects of topics in question and answers, emphasize the need for a model that distinguishes between topics in questions and answers and that can capture topic dependency and correlation across the whole corpus.

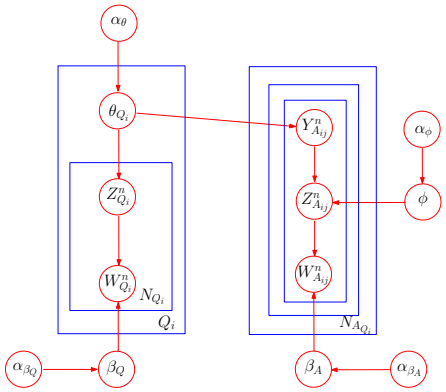


Figure 1: Question-Answering Topic Model (QATM). See text for details.

Using this intuition we introduce a model that incorporates two types of latent variables, question topic (Q-topics) and answer topic (A-topics). We refer to our model as Question-Answering Topic Model or QATM. The two types of topics allow us to model the differences in the vocabulary of questions and answers. They also allow the model to capture the correlation between topics.

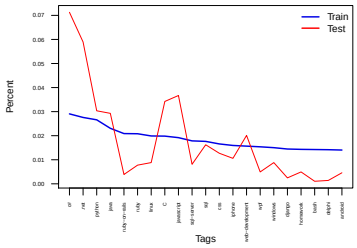
Q-topics (β_Q) and A-topics (β_A) are Multinomial distributions over distinct vocabularies for questions and asnwers respectively. We assume that there are K Q-topics and L A-topics. Each word ($W^n_{Q_i}$) in question Q_i is assigned to a Q-topic $Z^n_{Q_i}$ drawn from a Multinomial distribution θ_{Q_i} over Q-topics.

Each word ($W^n_{A_{i,j}}$) in answer j of question i is assigned to a A-topic ($Z^n_{A_{i,j}}$) that is conditioned on a Q-topic ($Y^n_{A_{i,j}}$). This Q-topic is drawn from the topic distribution of the corresponding question. By conditioning A-topics in an answer on Q-topics drawn from the topic distribution of the corresponding question, topics in answers are influenced by topics in the question and the model captures such a dependency. This is done through the latent variable ϕ, a $K \times L$ matrix. Each row k in ϕ defines mixture weights for A-topics corresponding to Q-topic k. This results in each Q-topic being associated with a distribution over A-topics. Dirichlet priors are defined over all θ_{Q_i} and rows in β_Q, β_A and ϕ with parameters α_θ, α_{β_Q}, α_{β_A} and α_ϕ respectively. We use the plate notation to show the QATM model in Figure 1.

3 Inference and Parameter Estimation

For doing inference, we need to compute the posterior probability of the latent variables Z_Q, Y_A, Z_A, θ_Q, ϕ, β_Q, and β_A given the input parameters $\alpha_\theta, \alpha_{B_Q}, \alpha_{B_A}, \alpha_\phi$ and observations W_Q and W_A. Exact inference is intractable for the model. We use collapsed Gibbs sampling [4] to sample the variables Z_Q, Y_A, and Z_A, integrating out θ_Q, ϕ, β_Q, and β_A.

For our model, we sample Y_A, and Z_A jointly and Z_Q separately. We need to compute two conditional distributions $\mathrm{P}(Z^n_{Q_i}|Z^{-n}_Q, Y_A, Z_A, W_Q, W_A)$ and $\mathrm{P}(Y^n_{A_{i,j}}, Z^n_{A_{i,j}}|Y^{-n}_{A_{i,j}}, Z^{-n}_{A_{i,j}}, Z_Q, W_Q, W_A)$ where $Z^n_{Q_i}$ represents Q-topic assignment for word n in question Q_i and Z^{-n}_Q denotes Q-topic as-



a) Tag Distribution in Training Set and Test Set

C#	.net	Sql
Sql-server	Java-script	css
Java	Ruby	Ruby-on-rails
Wpf	iphone	Web-development
Android	Windows	Delph
Django	Python	C
Bash	Linux	homework

b) Selected Tags

#Questions	1,188,585
#Answers	2,939,767
#Tags	27,659
#Users	440,672
Avg #answer per question	2.48
Avg #tags per question	2.93
Avg score of questions	1.50
Avg score of answers	1.85

c) Dataset Statistics

Figure 2: Dataset extracted from Stack Overflow Q&A website (details provided in the text).

signments for all other words except the current word $W^n_{Q_i}$. Moreover, $Y^n_{A_{i,j}}$ denotes the Q-topic assignment for word n in answer j of question i and $Z^n_{A_{i,j}}$ represents the A-topic assignment for the same word conditioned on $Y^n_{A_{i,j}}$. We have

$$\mathrm{P}(Z^n_{Q_i}{=}k|Z^{-n}_Q,Y_A,Z_A,W_Q,W_A)\propto\frac{\alpha_{B_Q}+C_Q^{kW^n_{Q_i}}}{\sum_{v=1}^{V}(\alpha_{B_Q}+C_Q^{kv})}\frac{\alpha_\theta+C^k_{Q_i}+C^k_{A_{Q_i}}}{\sum_{i'=1}^{k}(\alpha_\theta+C^{i'}_{Q_i}+C^{i'}_{A_{Q_i}})} \tag{1}$$

where C^{kv}_Q is the number of times word v is assigned to Q-topic k. Moreover, $C^k_{Q_i}$ is the number of times Q-topic k is assigned to words in question Q_i and $C^k_{A_{Q_i}}$ denotes the number of times A-topics for words in the set of answers for question Q_i are drawn conditioned on Q-topic k.

$$\mathrm{P}(Y^n_{A_{i,j}}{=}k,Z^n_{A_{i,j}}{=}l|Y^{-n}_{A_{i,j}},Z^{-n}_{A_{i,j}},Z_Q,W_Q,W_A)\propto\frac{\alpha_{B_A}+C_A^{lW^n_{A_{i,j}}}}{\sum_{v=1}^{V}(\alpha_{B_A}+C^{lv}_A)}\frac{\alpha_\theta+C^k_{Q_i}+C^k_{A_{Q_i}}}{\sum_{i'=1}^{k}(\alpha_\theta+C^{i'}_{Q_i}+C^{i'}_{A_{Q_i}})}\frac{C^l_k+\alpha_\phi}{\sum_{i=1}^{L}(\alpha_\phi+C^i_k)} \tag{2}$$

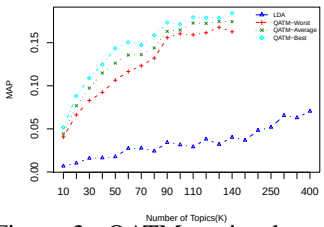
where C^{lv}_A is the number of times word v is assigned to A-topic l. Moreover, C^l_k is the number of times an A-topic l is drawn conditioned on a Q-topic k in the entire corpus.

4 Experiments

We evaluate our model on a real world dataset extracted from http://stackoverflow.com. Stackoverflow is a programming Q & A website, where developers can share technical information amongst themselves. To maintain an archive of high quality questions and answers, Stackoverflow employs popularity voting and allows users to vote upon and edit questions and answers. The users' contribution to the website is represented by reputation points and badges, based upon which they are granted more moderation capabilities and permissions. An archive of the content of this website is released every two months. For our experiments, we used the January 2011 data dump. Some statistics of this data are given in Figure 2.

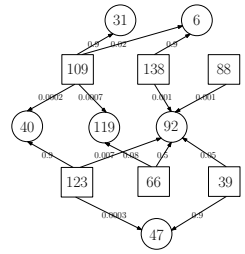
When a question is posted on Stackoverflow, it is tagged with labels or tags. To extract a representative subset of questions and answers from the large archive available on this website, we examined tag frequency and tag co-occurrence statistics, and manually selected a total of 21 tags. This subset was chosen such that a similar tag distribution as the the original data collection was maintained. The selected tags are shown in Figure 2.b. Subsequently, for each tag we randomly collected 200 questions (4200 questions in total). In addition, to allow the model to correctly learn the topic dependencies of a question and its answers, we extracted the 4 most relevant answers for each question using the scores given to answers by users based on their perceived relevance and correctness. At the end of this step we had extracted 15822 question-answer pairs for the train dataset.

To compare the answer retrieval performance of our model with the LDA model, we extracted a set of questions from Stackoverflow referred to as *duplicates*. These are questions that are similar to one or more existing questions in the archive but use different words and structure. Because they increase the archive's redundancy, duplicates are considered as a negative feature of CQA websites. Therefore, Stackoverflow users identify, vote upon and close such questions. We also tried to construct this "ground-truth" test dataset so that its tag distribution was similar to the tag distribution of our training dataset. This can be seen in Figure 2.a.



	Top1	Top2	Top3	Top4	Top5
LDA	0.023	0.026	0.029	0.03	0.032
QATM-Worst	0.108	0.127	0.138	0.144	0.148
QATM-Average	0.122	0.142	0.151	0.156	0.16
QATM-Best	0.131	0.152	0.161	0.164	0.168

Figure 3: QATM retrieval performance compared to the LDA model in terms of Mean Average Precision (a) and TopN (b) measures.



Topic Dependencies	Q-topic 39	Q-topic 66	Q-topic 88	Q-topic 109	Q-topic 123	Q-topic 138	A-topic 6	A-topic 31	A-topic 40	A-topic 47	A-topic 92	A-topic 119
	connect	end	java	email	stream	http	http	part	stream	connect	thi	answer
	server	begin	scala	piec	buffer	googl	www	email	buffer	socket	wai	question
	close	procedur	lang	send	mp	www	googl	send	data	port	ani	point
	port	assign	jvm	contact	audio	blog	org	home	length	network	onli	case
	open	declar	run	split	call	site	html	target	byte	send	doe	order
	client	val	util	mail	plai	org	blog	work	close	server	make	actual
	servic	direct	org	input	data	stackoverflow	articl	contact	mp	data	someth	follow
	remot	dir	class	final	download	en	en	messag	flush	client	check	give
	network	var	jdk	subject	decod	html	link	mail	reader	receiv	exampl	ye
	socket	proc	swing	sm	sound	sampl	aspx	sourc	upload	remot	doesn	expect
	resourc	const	jrubi	separ	util	wiki	wiki	sm	audio	summari	code	sinc
	notif	record	vm	pop	flush	place	post	relpath	output	proxi	befor	assum
	possibl	integ	librari	millisecond	upload	link	stackoverflow	split	memorystream	close	wa	consid
	transact	tmat	api	msxml	close	twitter	doc	address	payment	conn	note	happen
	local	vehicleid	apach	receiv	clap	msdn	archiv	smtp	dest	timeout	find	otherwis
	support	tform	bytecod	attach	audiomanag	home	msdn	gmail	bankhead	param	possibl	howev
	sqlconnect	fd	jython	regist	programmat	forum	librari	receiv	input	packet	abov	side
	asynchron	dynam	ee	programmat	listen	articl	github	subject	result	open	don	correct
	transactionscop	myobject	awt	current	video	aspx	found	spam	assert	implement	problem	inform
	applic	dcanva	sun	previousposit	inputstream	archiv	amp	bodi	filepath	host	put	sai

Figure 4: Topical dependencies captured by QATM with examples of Q-topics and A-topics represented by thier first 20 most probable words

We compare our model to the LDA model in terms of retrieving the right answer and report TopN (Mean Reciprocal Rank considered at a cutoff N) and Mean Average Precision (MAP) performance measures [7]. The latter emphasizes ranking relevant documents higher; this is important because each duplicate can have multiple correct answers (each query question can have several relevant answers). Results for MAP at various numbers of topics are plotted in Figure 3.a. Our model has two sets of topics, Q-topics and A-topics. Since the Q-topics in our model are similar to topics in the LDA model, when comparing to the LDA model, we report three MAP performance values for our model. Given the same number of topics for both models, we report an average, worst and best performance over a range of A-topic numbers for our model. These are denoted by QATM-Average, QATM-Worst and QATM-Best respectively. The results show that our model performs significantly better than LDA. Figure 3.b shows the TopN retrieval performance of the two models. Our model outperforms the LDA model. This indicates that our model can be used in combination with other information retrieval methods for improving results.

Our model is capable of capturing topical dependencies between questions and answers. Examples of topics from a model trained with 140 Q-topics and 120 A-topics are shown in Figure 4. Each topic is represented by its first 20 most probable words. In addition, the graph in Figure 4 shows the dependencies discovered between topics in questions and answers.

5 Conclusions and Future Work

We present a statistical topic model for question-answering archives. The model takes advantage of the assumption that topics in answers are dependent on topics discussed in questions. We apply the model to retrieve existing answers in the archive for new questions. Evaluating such a system is often challenging. We used information in Stackoverflow to extract a set of questions for which the right answers exist in the archive and are identified by users. This test subset makes quantitatively evaluating any answer retrieval model easier. Comparison of our model with the LDA model shows significant improvement in retrieval performance. Our model appears capable of capturing topic dependencies in questions and answers.

Our model can be used for automatic tagging of questions and answers on legacy Q&A websites lacking semantic information for browsing information. We are going to compare our tagging performance with the available tags on the Stack Overflow website for questions in our dataset.

References

[1] Eugene Agichtein, Carlos Castillo, Debora Donato, Aristides Gionis, and Gilad Mishne. Finding high-quality content in social media. In *WSDM '08: Proceedings of the international conference on Web search and web data mining*, pages 183–194, New York, NY, USA, 2008. ACM.

[2] Jiang Bian, Yandong Liu, Eugene Agichtein, and Hongyuan Zha. Finding the right facts in the crowd: factoid question answering over social media. In Jinpeng Huai, Robin Chen, Hsiao-Wuen Hon, Yunhao Liu, Wei-Ying Ma, Andrew Tomkins, and Xiaodong Zhang, editors, *WWW*, pages 467–476. ACM, 2008.

[3] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] G. Heinrich. Parameter estimation for text analysis. *Web: http://www. arbylon. net/publications/text-est.pdf*, 2005.

[5] Jiwoon Jeon, Bruce W. Croft, Joon H. Lee, and Soyeon Park. A framework to predict the quality of answers with non-textual features. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 228–235, New York, NY, USA, 2006. ACM Press.

[6] Jiwoon Jeon, W. Bruce Croft, and Joon Ho Lee. Finding similar questions in large question and answer archives. In Otthein Herzog, Hans-Jrg Schek, Norbert Fuhr, Abdur Chowdhury, and Wilfried Teiken, editors, *CIKM*, pages 84–90. ACM, 2005.

[7] Ellen M. Voorhees. The trec-8 question answering track report. In *In Proceedings of TREC-8*, pages 77–82, 1999.