INCORPORATING AUTHOR'S ACTIVENESS IN ONLINE DISCUSSION IN THREAD RETRIEVAL MODEL

Zuriati Ismail[1, 2], Atefeh Heydari[1], Mohamadali Tavakoli[1] and Naomie Salim[1]
[1]Faculty of Computing, Universiti Teknologi Malaysia (UTM), Skudai, Johor, Malaysia
[2]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM), Malaysia
E-Mail: zuriati.ismail@gmail.com

ABSTRACT

Online forum is one of user-generated contents available on the Internet that provides platform for knowledge sharing. However, not all messages posted can be considered of high quality and as it increases in its availability, finding quality information becomes more important and challenging. Thread retrieval model is very important in helping users to find relevance information pertaining to their topic search. As quality of post messages depends upon the author, this study aims to look at how ranking threads based on author's activeness in a forum could improve thread retrieval task compared to non-quality based ranked list. Voting models were used to convert message level quality features into thread level features and learning to rank method to combine nine features of activeness dimension for thread scoring. Different combinations of nine features under the activeness dimension with different ranking strategies are studied and its performances also compared using normalized discounted cumulative gain (NDCG) as performance measure. 2555 models were generated and 23 models are identified as among the best model.

Keywords: thread retrieval, voting model, learning to rank.

INTRODUCTION

Online forum is one of user-generated content platforms available on the Internet that encourage knowledge sharing from many contributors (users). The applications or areas of interests vary widely and there are infinite numbers of online forums available online. Forums are online discussion sites where users hold conversations that started from a post that seeks an answer or information, solutions to real problem and even technical assistance depending on the virtual communities' interest. In comparison to some other platforms like blogs or wikis, most published information are predicted and shared publicly and tailored to a specific problem experienced by the author or known to the author. Forums on the other hand, provide platform that users can converse and discuss in detail of a particular problem through its conversational like structure. A user might post a question or a message looking or asking a particular solution or an opinion over a specific topic and he/she might get a response either in a form of another question for a more specific detail, or better yet a solution to the problem posted. Each response might be a response to the post or a response to a previous message. The post and every response it gets (either a direct reply or reply to another reply) is called a thread.

Even though it is interesting to note that there is a lot of valuable information or knowledge shared through an online forum (Seo *et al*., 2011) anyone who face with the same kind of situation or looking for an answer to the same question finds it hard to retrieve the information from a relevant thread. It is readily available but could not be retrieved conveniently. This problem contributed to the fact that the way the discussions are being displayed and stored is not like common web pages. Earlier posts will always be pushed backward and latest post will be displayed first. The larger the number of users registered to a particular forum and the longer the forum has been established, the more posts and messages are expected and the harder the process of retrieving threads because of information overload. It is also a fact that current search engines fail to consider the unique structure of online forum while forums internal search engines are not sophisticated enough to handle its complexity.

The fact that online forums maintained and handled differently by online communities also posed a problem in thread retrieval. In general, it is maintained and displayed either in a flat-view or threaded-view and some researches focuses on thread discovery structure before starting on thread retrieval (Seo *et al*., 2011; Seo *et al*., 2009). Threaded-view forums are more convenient to be viewed as compared to flat-view (Elsas and Carbonell, 2009). For the experiment we conducted, we use data from TripAdvisor-New York. The data have been preprocessed for thread discovery structure where all associated metadata (title, posts, user IDs etc.) have been identified (Bhatia and Mitra, 2010).

QUALITY AND THREAD RETRIEVAL

Content quality assessment

Agichtein *et al*. (2008) classified the quality of user generated content as either excellent, abusive or spam. The author also relates the increment of data available will make the task of identifying high quality data becomes more important but of course the task of retrieving it will be harder. Chai (2011) highlighted some reasons why quality of user generated contents differ; (1) because of the diversity of user background in terms of geographical location, beliefs, motivation and knowledge; (2) contents are mostly generated by users who have received little (if any) professional training in content creation and contents are mostly published without peer review (Weimer and Gurevych, 2007); and (3) in the consumption parts, contents are consumed by millions of users who have different motivations and requirements. Currently quality of user generated contents is being assessed manually through user ratings including online forums.

Dependent upon manual rating in assessing quality has to consider some problems such as ratings is voluntarily basis and not all users rate the contents. Furthermore we also could not assume honesty and biased free ratings and not all raters have sufficient knowledge in regards to the topics being assessed (Chai, 2011) and hence the need for content quality assessment to be done automatically.

As mentioned previously, retrieving threads is not an easy task because of information overload. Not all information in online forum can be considered as useful. There might be spam or information that is no longer relevance at the point of retrieval. This paper however is only focusing on thread retrieval and it is unique where models used for retrieval incorporating quality features under activeness quality dimension that studies the behaviour of each user's participation in a discussion. There are nine features used to assess the quality of the thread automatically by measuring author's activeness before the retrieval task. This study is focusing on whether by incorporating author's activeness can improve performance of retrieval model and at the same time looking at how different aggregating strategies improve model's performance. Analysis will also look at combinations of different features and strategies. For the purpose of this study, only four score based strategies (CombSum, CombMax, CombMed, CombMin) and one ranked based strategy (BordaFuse) were implemented.

Related works

Studies on thread retrieval has evolved from blog feed search (Albaham and Salim, 2013a; Elsas and Carbonell, 2009; Seo *et al*., 2011;2009) and expert finding (Albaham and Salim, 2012;2013b). The similarity of blogs and threads is blog has collection of blog posts and thread has collection of messages as document collection. It is also similar in expert finding where peoples (candidates) have collection of documents associated with them to be evaluated to estimate the expertise of the candidates. However, thread retrieval differs from both blog feed search and expert finding because of its conversational like structure. Macdonald and Ounis (2006) proposed an approach to rank experts with respects to users query by looking at ranking experts problem as a voting problem. Eleven data fusion techniques were adapted. The study that was conducted on an expert search task shows significant improvement of retrieval performance using the data fusion techniques. Albaham & Salim (2013b) use the voting models to convert message level quality features into thread score. The author incorporate amount of data quality dimension as message level quality features and found that voting model helps in scoring messages and then convert it into thread score.

Elsas and Carbonell (2009) divides the models in their research generally into two types (inclusive and selective). Two models were adapted from Elsas *et al*. (2008) namely large document and small document models. The large document model concatenate thread messages as a single document and similarity between document and query evaluated. The small document on the other hand treat each thread message as a single document and query relevance score calculated for each thread message. Threads are then scored by averaging messages relevance scores. These two models are the inclusive models. As the name implies, selective models only select few messages to score threads. The author applies three selective methods (1) threads scored by initial message relevance score; (2) thread scored by the maximum score of messages relevance score and (3) based on Pseudo Cluster Selection (PCS) method (Seo and Croft, 2008). The large document model was used as the baseline and it was found that selective models are statistically better than inclusive model. PCS also superior to all methods studied.

Bhatia and Mitra (2010); Seo *et al*. (2011;2009) studied on how thread structure beneficial in improving thread retrieval model. The authors found that by discovering the thread structure significantly improve thread retrieval over strong baseline. The former introduce a thread retrieval model based on inference network and utilises thread structure besides incorporating quality features in the proposed model.

Quality in thread retrieval model

Only limited studies have incorporated quality features in thread retrieval model (Albaham and Salim, 2013b; Bhatia and Mitra, 2010). The former incorporate amount of data quality dimension while the latter incorporate user's authority, length of thread and users reference link to a relevant thread in their study. Both studies found that by incorporating quality features does improve model's performance over baseline even though different methodology were proposed by these authors.

Zhang (2009) on the other hand propose an approach based on knowledge adoption model and genetic algorithm in incorporating quality features in the study. The study explored argument quality and source credibility based on member's social interaction in an online knowledge community and shows better performance.

These promising results prove that incorporation of quality features in thread retrieval is a plausible solution for thread retrieval.

Author's activeness dimension

Activeness measures how active a particular author in a forum. It is assume that the more active the author is, the more experience he/she is and therefore will increase the author's trustworthiness (Zhang, 2009). There are nine features under the activeness dimension. Each feature describes an author's history of participation in a particular forum. The description and measurement of each feature is written in Table-1. Features are measured during the indexing process and kept in a file to be used for message and thread scoring.

Table-1. List of features in Authors' activeness dimension.

	Activeness features	Description/measurement
1	AuthAge	Date of author's last post - date of his first post (Chai, 2011; Burel *et al*., 2012)
2	InitPost	Number of initiated author replies to his/her own initiated threads (Zhang, 2009)
3	RplyPost	Number of author's reply posts (Zhang, 2009)
4	TtlPost	Total number of posts created by the author (Chai, 2011)
5	RplyInit	RplyPost / InitPost
6	TtlPstAge	TtlPost / AuthAge (Zhang, 2009)
7	LstPost	Time of author's last post - the system base time (Zhang, 2009)
8	ThrPart	Number of threads the author has participated in (Zhang, 2009)
9	AvgTime	Average time between author's consecutive posts (Zhang, 2009)

EXPERIMENTS AND RESULTS

Experimental setup

In evaluating the proposed models, TripAdvisor-New York dataset is used (Bhatia and Mitra 2010). This forum is a travel site providing expert advice for anyone looking for information regarding travelling in New York. New York is one of the 42 countries where TripAdvisor operates. This forum enables forum members to ask and share their experiences, advices and opinions interactively through discussions among them.

The data had already been preprocessed with the following statistics (refer Table-2). Stemming was performed and stop words were removed using Porter's stemmer and Onix Test Retrieval Toolkit (Bhatia and Mitra, 2010).

Table-2. Statistics of TripAdvisor New York Forum.

No. of threads	83072
No. of users	39454
No. of messages	590021
No. of queries	25
No. of evaluated threads	4478

Twenty-five queries were generated by Bhatia and Mitra (2010) and sample of the queries are featured in Table-3.

Table-3. Queries samples.

Example of queries
how safe is New York
how much to tip people
New York to Niagara falls
Christmas day attractions

Retrieval models are developed by combining nine features of author's activeness (Table-1) with five different strategies (refer Table-4). All in all we have 2555 total combinations of thread retrieval models. The baseline is the retrieval model that rank thread without using any quality features.

In generating the rank list of messages, the following scoring function was used:

$$score\ (Q,M) = rel\ (Q,M) \times sigm(f) \qquad (1)$$

where rel(*Q*,*M*) is the query(*Q*)-message(*M*) relevance score estimated using Divergence from Randomness retrieval model (Amati and Van Rijsbergen, 2002) and sigm(*f*) is the sigmoid transformation (equation 2) of the quality feature *f* of the message *M* (Albaham and Salim 2013b).

$$sigm(f, k, w, a) = w\frac{f^{a}}{k_{a} + f^{a}} \qquad (2)$$

where *w* = 1.0, k = 1.0 and *a* = 0.6. Thread are then scored based on the aggregated ranked messages scores and ranks and then the threads are ranked in descending order (Albaham and Salim, 2012). The five aggregating strategies are listed in Table-4 (Macdonald and Ounis, 2006). Four of the ranking strategies (CombSum, CombMed, CombMin and CombMax) are score-based strategy that combine rankings using scores of the retrieved documents (in our case messages) and ranked-based strategy (Borda-Fuse) combines rankings based on ranks of the retrieved documents (in this case, the threads are ranked based on ranks of its messages).

Table-4. List of aggregation strategies used. D(C, Q) is a set of messages in a thread and ||. || is the size of the described set.

Name	Summary
CombSum	Sum of scores of messages in a thread
CombMed	Median of scores of messages in a thread
CombMin	Minimum of scores of messages in a thread
CombMax	Maximum of scores of messages in a thread
Borda-Fuse	Sum of (||R(Q)|| - rank of messages in D(C,Q))

Performance evaluation

In order to evaluate models' performance, Normalized Discounted Cumulative Gain (NDCG) is used as the performance measure. NDCG is a normalization of Discounted Cumulative Gain (DCG) that has two advantages compared to other performance measure. Firstly, NDCG allows degree of relevance between 0-1 while most performance measure only allow binary relevance where a document is only either relevant (1) or not relevant (0). Secondly, most performance measure weight all positions uniformly while the weight of NDCG is the decreasing function of the object's (document) rank (position) (Wang *et al*. 2013). DCG is calculated as the weighted sum of the degree of relevancy of the our ranked documents. NDCG measures the usefulness of *k* retrieved documents. Therefore the higher NDCG will means more similar documents to the query are retrieved for top-k ranked documents. This study we evaluate model's performance for NDCG@30 and NDCG@100.

Results and analysis

Combination of nine features and five strategies forms 2555 models. These models are then ranked in descending order based on the improvement over baseline for NDCG@30 and NDCG@100. Both ranks (NDCG@30 and NDCG@100) are then compared, and only models that improve at both NDCG@30 and NDCG@100 are left and the rest were removed. There are 42 models that improved at both NDCG@30 and NDCG@100. Each model's ranks are then summed up and the 42 models are then ranked in ascending order based on sum of ranks. Only sum of ranks of 100 or less are then selected and this gives us 27 models. Each model are then tested whether the improvement at both NDCGs was significant or not significant using paired sample T-test. Out of this 27, four models were removed since these models' improvement are not statistically significant for NDCG@30 and the 23 models listed in Table-5, are models with sum of ranks 100 or less and shows significant improvement at p-value <= 0.01 and <= 0.05 (results are bold) over baseline at both NDCG@30 and 100.

Table-5. List of 23 best models.

	Features	**Sum of improvement** (%)	**Strategy**	**NDCG@**	
	No quality feature			**100**	**0.332**
				30	**0.318**
1	TtlPost+ TTlPstAge	4.666	BF	100	0.352
				30	0.345
2	TtlPstAge+LstPost	4.4028	BF	100	0.350
				30	0.344
3	AuthAge+RplyPost+ TtlPostAge+InitPost	4.2108	BF	100	0.348
				30	0.345
4	AvgTime+ThrPart+ InitPost	4.28	BF	100	0.348
				30	0.345
5	TtlPost+AvgTime+ InitPost	4.0036	BF	100	0.348
				30	0.343
6	AuthAge+TtlPost+InitPost	3.9564	BF	100	0.348
				30	0.342
7	RplyPost+RplyInit+ ThrPart+InitPost	3.9272	BF	100	0.348
				30	0.342
8	TtlPstAge+LstPost+ AvgTime+InitPost	3.9264	BF	100	0.347
				30	0.342
9	AuthAge+TtlPost+RplyInit+Avg Time+ThrPart+InitPost	3.9188	BF	100	0.347
				30	0.342
10	AuthAge+RplyInit+ TtlPstAge+AvgTime+InitPost	3.8312	BF	100	0.348
				30	0.342
11	RplyPost+TtlPost+ RplyInit+LstPost	3.758	BF	100	0.347
				30	0.341
12	TtlPost+TtlPstAge+LstPost	3.7936	CS	100	0.347
				30	0.341
13	RplyPost+TtlPost+TtlPstAge+Ls tPost+ThrPart	3.8004	BF	100	0.347
				30	0.342
14	TtlPost+TtlPstAge+ ThrPart+InitPost	3.8068	CS	100	0.346
				30	0.342
15	AuthAge+LstPost+InitPost	3.7736	BF	100	0.346
				30	0.342
16	AuthAge+TtlPost+RplyInit+TtlP stAge+LstPost	3.9332	BF	100	0.346
				30	0.344
17	RplyPost+RplyInit+TtlPstAge+L stPost+InitPost	3.6564	CS	100	0.347
				30	0.340
18	RplyPost+TtlPost+RplyInit+LstP ost+AvgTime+ThrPart	3.7052	BF	100	0.346
				30	0.341
19	RplyPost+RplyInit+ LstPost+InitPost	3.6392	CS	100	0.347
				30	0.340
20	RplyPost+LstPost+ AvgTime+InitPost	3.864	BF	100	0.346
				30	0.343
21	AuthAge+RplyInit+ TtlPstAge+ThrPart	3.7588	CS	100	0.346
				30	0.342
22	RplyPost+TtlPost+RplyInit+Avg Time+ThrPart	3.6192	BF	100	0.347
				30	0.340
23	TtlPost+TtlPstAge+InitPost	3.6064	CS	100	0.350
				30	0.340

Based on the list of models in Table-5, almost 75% of the list are models with BordaFuse strategy and make up top eleven in the lists. It shows that BordaFuse is a competitive strategy for thread retrieval and Macdonald and Ounis (2006) has found that it is also perform well for expert search. Albaham and Salim (2012) found that BordaFuse shows comparable performance in thread retrieval when compared to virtual document model. The study also found that CombSum which favour threads with highly ranked messages shows comparable performance over virtual document even though the improvement was not statistically significant.

In terms of number of features, most of the models in the list are combinations of four features (eight out of 23 models are combinations of four features) but two of the best models are combinations of only two features. Maximum number of features in the list is a combination of six features. This shows that the greater the number of features in a combination does no good to the model and the best number of features in a combination is two while four combinations of features is the right combination as 35 percent of the best models combined four features in a model. One feature in a model is not advisable for thread retrieval. Therefore we would like to suggest quality features in a model should be between two to six features. However, we still need to test its consistency across datasets.

Quality features assessments

Table-6 shows interactions between features where frequency of feature A and feature B appear together in a model in the list of best 23 models are counted and the sum of frequency of each feature appear in the list of the best models also reported. Based on the Table, we can conclude that author’s age is the least likely feature to appear in the best 23 models. The feature is good to be combined with total number of post created by author, ratio of (total post and author’s age), number of author’s initial post and ratio of (reply post and number of author’s threads).

Number of initiated author replies in its own threads meanwhile is the feature that featured most in all the 23 models and shows that it is good to be combined with all of the other features except for number of threads author has participated in. This feature might be a good indicator of author’s experience as it could reflects the author’s involvement in a particular discussion Zhang (2009).

Beside the two features mentioned above, ratio of (total post and author’s age) can be considered as the best feature to be included in a model as it is in the top three of the best models. This feature measures author’s activeness in the duration of their existence. This shows that the value of the information shared by active authors is higher than the less active authors.

In general all features are useful to be used for scoring messages but it depends on the combination (interaction with other feature) that will determine whether a particular model will improve retrieval significantly or not.

Table-6. Frequency of interactions between features and frequency features featuring in the 23 models (Feature 1-9 are based on Table-1).

Feature	1	2	3	4	5	6	7	8	9
1	0	5	1	4	4	4	2	1	2
2		0	5	5	5	6	5	2	6
3			0	4	6	3	6	3	2
4				0	5	6	5	5	4
5					0	4	5	5	4
6						0	6	3	2
7							0	2	3
8								0	4
9									0
Freq. features	7	14	9	12	10	12	11	8	8

CONCLUSIONS

In this study, we are exploring the possibilities of combining different features with a number of aggregation strategies in thread retrieval and determine the best combination among all of the combinations. Out of total 2555 combinations, these 23 models are listed as among the best models but only after tested for high precision searches (NDCG@100 and NDCG@30). Based on results it shows that by combining features in a model will give better performance than models that only have single feature and when compared to baseline but however combinations of more than six features in a model do not improve model’s performance. BordaFuse and CombSum shows that these two strategies is good for high precision searches and therefore we need to study its performance in low precision searches in our future works beside looking closely at the performance of models across different datasets. In order to determine the best feature and model, we have to consider more performance measure in the future. These study list down all models that significantly perform better than the baseline (model that not incorporate quality) and this list shows that by incorporating author’s activeness in a model does improve models performance. Since this study only focusing on one quality dimension, we are also exploring other quality dimensions to be studied in the future to better improve our retrieval model.

ACKNOWLEDGEMENTS

The work is supported by the Ministry of Education (MOE) and Research Management Centre (RMC) of Universiti Teknologi Malaysia (UTM) under research university grant category (Vot:R.J130000.7828.4L630).

REFERENCES

Agichtein E. *et al*. 2008. Finding high-quality content in social media. In: Proceedings of the international conference on Web search and web data mining - WSDM '08. New York, New York, USA: ACM Press. p. 183. Available at: http://portal.acm.org/citation.cfm?doid=1341531.1341557.

Albaham A.T. and Salim N. 2012. Adapting Voting Techniques for Online Forum Thread Retrieval. In: A. E. Hassanien *et al*., eds. Advanced Machine Learning Technologies and Applications. Springer Berlin Heidelberg. pp. 439-448.

Albaham A.T. and Salim N. 2013a. Online Forum Thread Retrieval Using Pseudo Cluster Selection and Voting Techniques. In: R.-S. Chang, L. C. Jain and S.-L. Peng, eds. Advances in Intelligent Systems and Applications - Volume 1. Smart Innovation, Systems and Technologies. Berlin, Heidelberg: Springer Berlin Heidelberg. pp. 297-306. Available at: http://link.springer.com/10.1007/978-3-642-35452-6 [Accessed June 2, 2014].

Albaham A.T. and Salim N. 2013b. Quality biased thread retrieval using the voting model. In: Proceedings of the 18th Australasian Document Computing Symposium on - ADCS '13. New York, New York, USA: ACM Press, pp. 97–100. Available at: http://dl.acm.org/citation.cfm?doid=2537734.2537752.

Amati G. and Van Rijsbergen C.J. 2002. Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Transactions on Information Systems. 20(4): 357-389. Available at: http://portal.acm.org/citation.cfm?doid=582415.582416.

Bhatia S. and Mitra P. 2010. Adopting Inference Networks for Online Thread Retrieval. In: AAAI. pp. 1300-1305. Available at: http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/download/1886/2199 [Accessed August 11, 2014].

Burel G., He Y. and Alani H. 2012. Automatic Identification of Best Answers in Online Enquiry Communities. In: 9th Extended Semantic Web Forums. pp. 514-529.

Chai K.E.K. 2011. A machine learning-based approach for automated quality assessment of user generated content in web forums. Curtin University. Available at: http://espace.library.curtin.edu.au/cgi-bin/espace.pdf?file=/2011/11/24/file_1/169169 [Accessed August 11, 2014].

Elsas J.L. et al. 2008. Retrieval and feedback models for blog feed search. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '08. New York, USA: ACM Press, p. 347. Available at: http://portal.acm.org/citation.cfm?doid=1390334.1390394.

Elsas J.L. and Carbonell J.G. 2009. It Pays to be Picky : An Evaluation of Thread Retrieval in Online Forums. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. pp. 714–715. Available at: http://dl.acm.org/ft_gateway.cfm?id=1572092&ftid=652212&coll=DL&dl=GUIDE&CFID=326968707&CFTOKEN=83478497.

Macdonald C. and Ounis I. 2006. Voting for Candidates : Adapting Data Fusion Techniques for an Expert Search Task. In: CIKM '06 Proceedings of the 15th ACM international conference on Information and knowledge management. pp. 387-396.

Seo J., Bruce Croft W. and Smith D. a., 2011. Online community search using conversational structures. Information Retrieval. 14(6):547-571. Available at: http://link.springer.com/10.1007/s10791-011-9166-8 [Accessed June 15, 2014].

Seo J. and Croft W.B. 2008. Blog site search using resource selection. In: Proceeding of the 17th ACM conference on Information and knowledge mining - CIKM '08. New York, New York, USA: ACM Press. p. 1053. Available at: http://portal.acm.org/citation.cfm?doid=1458082.1458222.

Seo J., Croft W.B. and Smith D. a. 2009. Online community search using thread structure. In: Proceeding of the 18th ACM conference on Information and knowledge management - CIKM '09. New York, USA: ACM Press. p. 1907. Available at: http://portal.acm.org/citation.cfm?doid=1645953.1646262.

Wang Y. et al. 2013. A theoretical analysis of NDCG ranking measures. In: 26th Annual Conference. pp. 1-30. Available at: http://www.cis.pku.edu.cn/faculty/vision/wangliwei/pdf/NDCG.pdf [Accessed December 7, 2014].

Weimer M. and Gurevych I. 2007. Predicting the perceived quality of web forum posts. In: Proceedings of the conference …. Available at: http://www.tk.informatik.tu-darmstadt.de/fileadmin/user_upload/Group_UKP/publikationen/2007/ranlp2007.pdf [Accessed August 11, 2014].

Zhang X. 2009. Effective Search in Online Knowledge Communities : A Genetic Algorithm Approach. Virginia Polytechnic Institute and State University.