Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 143 (2018) 333–340

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Advances in Computing and Communication (ICACC-2018)

Discovering Top Experts for Trending Domains on Stack Overflow

Polisetty Sumanth[a]*, Rajeshwari K[a]

[a] *Department of Information Science and Engineering, BMS College of Engineering, Bengaluru*

Abstract

Stack Overflow is an online Question Answer Community, generating huge user content for research analysis. Stack Overflow is used to find top experts on a specific domain. The domains of interest are java, javascript, C# and python. The study identifies topic specific expert as with the Stack Overflow statistics, where the latter provides top experts based on reputation score irrespective of the domain. The algorithms used are PageRank and HITS score. The Discounted Cumulative Gain is used to measure the correctness of the analysis.

Keywords: Stack Overflow; Social Network Analysis; SNAP; top experts; PageRank; HITS; DCG

1. Introduction

Networks are used as a tool for modelling complex systems in social networks. Social network is a social structure made up of actors (individual nodes) and ties (links between actors). Some of the social networks which can be analysed are Facebook, Google, Stack Overflow, and Gmail. The user interactions in these networks generate huge set of data which can be analysed to find the patterns, to do some useful predictions. There is creation of large amount

* Corresponding author. Tel.: 918892196239

E-mail address: sumanth472@gmail.com

1877-0509
10.1016/j.procs.2018.10.404

of user-generated content every day, due to growing internet accessibility to everyone and to help each other in many online communities. Jure Leskovec [1] addresses as how to collect massive data on social media, tracking and extracting interesting information, to find most emerging topics of discussions on social media. Stack Overflow is an online community among software developers, where some users query questions. And some users are providing solutions to the posts. Gang Wang, Konark Gill, Manish Mohanlal, Haitao Zheng and Ben Y Zhao in [2] prepare graphs for Quora, another QA online community. Individual graphs analysis is done, wherein a link exists between topics to users; between users; and between related questions. Data obtained from “Stack Overflow” can be represented in the form of graphs. The users are considered as nodes. The person who posts a question gets an answer from another user, and then an edge is created between them. In stack overflow, the most discussed topics are java, C#, python, Android, HTML, javascript, PHP, and jQuery. In this paper, the prediction of top experts of Stack Overflow in a specific domain is identified. The domains chosen are python, C#, javascript and java. The top experts' position is compared with the Stack Overflow statistics.

2. Related Work

There has been much work carried out on Stack Overflow analysis, various aspects of the Q&A community to provide useful results in improving the platform in user interests. Rabe Abdalkareem, Emad Shihab, and Juergen Rilling in [3] studied Stack Overflow community to determine the usage of Stack Overflow in regard of the delay of answering certain questions. They found from the analysis, that the community was mostly used for exchanging the development related problems or tasks. And comparatively found that programming related issues were answered quickly than framework related issues which took longer time to answer. Jure Leskovec, Jon Kleinberg, Christos Faloutsos in [4] discuss on how graphs evolve over time. They provide detailed analysis on the growth patterns in social, technological, information networks. It was found that graphs densify with more nodes over time with distance between those nodes reducing greatly. Jaewong Yang, Jure Leskovec in [5] provided a methodology to compare and evaluate structural definitions in large community networks. They discuss about four goodness metrics for scoring function, like separability, density, cohesiveness, clustering. The comparative study proved 30% improvement (efficiency, robustness, performance) over other similar methods used to evaluate the networks. Chunyang Chen, Zhenchang Xing in [6] analyzed the searches on Google and Stack Overflow community. They tried to find out correlations between the search keyword on Google, in fetching a Stack Overflow result. Tore Opsahl, Filip Agneessens, and John Skvoretz in [7] proposed few network measures used for weighted networks based on tie weights and number of ties for measuring node centrality (degree, closeness, betweenness). Bogdan Vasilescu, Andrea Capiluppi, and Alexander Serebrenik in [8] discussed the gender participation ratio in online Q&A communities and the initiative taken by different communities to encourage women participation in field of technology. Dana Movshovitz-Attias, Yair Movshovitz-Attias, Peter Steenkiste and Christos Faloutsos in [9] and Amiangshu Bosu, Christopher S. Corley, Dustin Heaton, Debarshi Chatterji, Jeffrey C. Carver, and Nicholas A. Kraft in [10] investigated the reputation of users in online community. They observed the participation patterns of the users with different reputation scores and found out that the higher the reputation score, the more questions are posted from them. They provided four aspects to understand the dynamics of reputation scores and various attributes involved in assigning the score to users. Luca Ponzanelli, Andrea Mocci, Alberto Bacchelli, Michele Lanza, David Fullerton in [11] proposed an improved method of analyzing and detecting low quality posts by the simple textual features and readability metrics to remove posts which are misclassified as low-quality posts. Eduardo Cunha Campos and Marcelo de Almeida Maia in [12] provided comparison between various classification algorithms to classify the questions on Stack Overflow community into one of the three categories like How-to-do-it, Need-to-know, Seeking-something. Galina E. Lezina, Artem M. Kuznetsov in [13] provided a classifier to predict whether a question can be closed and the reason for the same. They used support vector machine, random forests and vowpal wabbit algorithms by providing the features to the model to predict the closed questions. Clayton Stanley, Michael D. Byrne in [14] built a Bayesian probabilistic model to predict the tags used by owner of the posts on Stack Overflow. The tags which the model predicts is more similar for the users tag to get posted. Pooja Wadhwa and M.P.S Bhatia in [15] discussed the challenges and opportunities available in social networks and provided a structure to understand the domain and trending topics in social network analysis.

3. Methodology

3.1. Dataset

The dataset used in the project is Stack Overflow dataset containing posts of questions-answers related to various domains that were asked on the Stack Overflow community. The posts collected are from January 2008 to September 2017. The dataset is made available online by Stack Overflow on their official website in the xml format for usage. This dataset is used to analyze the network to predict top experts of specific domain and analyze various properties of the network. Varun Kumar, Niranjan Pedanekar in [16] provided observations on distribution of expertise on Stack Overflow in form of different levels (expert, beginner, and novice) to help recruiters identify specific users of expertise.

3.2. Parsing Relevant Information

- Parsing questions from stack-overflow posts using xml parser by extracting the questionID, questionOwnerID, acceptedAnswerID, and extracting the IDs of all posts related to that domain by parsing tag as input.
- Extracting all posts from the questions parsed earlier by matching questionIDs with the Ids of posts related to specific domain extracted earlier.
- Extracting the edge list to construct the network by matching the acceptedAnswerIDs of the questions related to specific domain and answerIDs of the answers parsed earlier.
- Edge list consists of the questionOwnerID and acceptedAnswerOwnerID
- Edge list consists of the questionOwnerID and acceptedAnswerOwnerID and is a directed edge from questionOwnerID to acceptedAnswerOwnerID.

3.3. Constructing the Network

Network is constructed by loading the edge list using SNAP python library and creating graph object. Jure Leskovec, Rok Sosic in [17] provided a tool called SNAP (Stanford Network Analysis Platform) for analyzing large social and information networks.

Table 1. Statistics of Python, C#, javascript, java Networks

Property	Python	C#	JavaScript	Java
Graph Type	Directed	Directed	Directed	Directed
Nodes	199156	222839	380844	303401
Edges	419712	599237	757489	624971
Zero Degree Nodes	0	0	0	0
Zero In-Degree Nodes	0	0	0	176246
Zero Out Degree Nodes	0	0	0	59405
Non-Zero In-out Degree Nodes	199156	222839	380844	67750
Unique directed Edges	813796	1155221	1463358	624971
Unique undirected edges	419712	599237	757489	624763
Self -Edges	25628	43253	51620	46006
Bi Directional Edges	813796	1155221	1463358	46422
Closed Triangles	160213	335499	214728	55689
Open Triangles	112141365	128030036	107957166	113644528
Fraction of closed triads	0.001427	0.002614	0.001985	0.000490
Connected component size	0.887902	0.895247	0.869829	0.868900
Strong connected component size	0.887902	0.895247	0.869829	0.021233
Approximate full diameter	12	12	16	13
90% effective diameter	5.64	5.45	5.93	5.77

Table 1 shows the various properties for the topic specific domains chosen. Using SNAP tool, the various built in algorithms were used to construct the network for huge data sets. The tool extracted the metadata of the graph too.

3.4. Predicting Top Experts

Jun Zhang, Mark S. Ackerman, Lada Adamic in [18] discussed ranking based algorithms used for social networks. The authors predict top experts for Java forum, using algorithms like similarity measure, Z score and PageRank and HITS. Top experts can be found out by the node centrality which indicates the most important nodes in the network. These nodes are considered as top experts. The top experts in specific domains like java, javascript, C# and python are predicted.

Page Rank Algorithm

Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd [19], PageRank algorithm is used to find out the most important nodes in the network constructed by distributing equal weights to all nodes and these weights are repeatedly distributed until convergence point is achieved (weights of each of the nodes stop changing after certain number of iterations). PageRank algorithm distributes the probability assigned in the range 0 to 1. The nodes in the network are assigned value of 1 and the value of a particular node is distributed among other nodes by equally dividing based on number of links that are directed to other nodes. The nodes which have more incoming edges towards it are the most important nodes. This algorithm helps in finding out top experts of specific domain in Stack Overflow.

Computation of Page Rank

$$PageRank\ (N) = 1 - DampingFactor + DampingFactor * \frac{PageRank(N_1)}{Count(N_1)} + \dots + \frac{PageRank(N_n)}{Count(N_n)} \tag{1}$$

- PageRank(N) – Page Rank of node N
- DampingFactor – The probability included to prevent the nodes with no outgoing links to take up the value of the nodes with outgoing links
- PageRank (N_n) - PageRank of node N_n which link to node N
- Count(N_n) – Number of outgoing links for node N_n

HITS Algorithm

HITS algorithm finds out most important and influential nodes in the network by using two scores, hubs and authorities score. Hub represents webpages that points to other webpages. Authority represents webpages that have many hubs pointed to it. HITS algorithm performs two operations iteratively until convergence is obtained.
Hub update: Kleinberg, Jon M. in [20] discusses about a node assignment. In iteration, a node is assigned a hub score which is equal to sum of authority scores of each node that points to hub node. Nodes are given highest hub score based on linking to nodes that are authorities.
Authority update: In iteration, a node is assigned authority score which is equal to sum of hub score of each node pointing to it. Nodes are given highest authority based on more number of hub nodes connected to it.

4. Results

This section describes the various results obtained after applying the PageRank and the HITS algorithm. The table 2 shows the results obtained for python. The results are tabulated for the existing Stack Overflow scores and the algorithm used. ID 100297 of Stack Overflow has a reputation score 591000 with Stack Overflow position at 10. As per the results, the ID 100297 is top expert for python. Difference between the scores is seen as Stack Overflow results are declared without considering a specific domain of the user and the reputation score is a generalized score assigned for the user's expertise. Top 10 experts determined by the algorithm are depicted in the table 2. The top experts predicted by Page Rank and HITS vary as some of the results predicted by Page Rank are missing in HITS prediction. The top 10 experts by PageRank and HITS algorithms are considered for comparison. NP (Not Present) is assigned for the missing results in the algorithm predictions in the tables. ID 22656 in present in table 3 and table 5, is identified to be expert in C# and Java domains.

Table 2. Results comparison of Page Rank and HITS of Python Network

Id	Page Rank Order	HITS Hub Order	HITS Authority Order	Stack Overflow Position	Reputation Score	Page Rank Score	HITS Hub Score	HITS Authority Score
100297	1	1	1	10	591000	0.009717	0.567832	0.713315
190597	2	2	2	29	454000	0.003104	0.099677	0.116712
771848	3	3	3	88	269000	0.003064	0.070402	0.087080
104349	4	7	6	42	378000	0.002760	0.054858	0.066083
29010002	5	10	NP	223	167000	0.002373	0.040047	NP
2225682	6	4	4	160	207000	0.002230	0.065016	0.078399
908494	7	6	5	183	188000	0.002150	0.057181	0.068617
2141635	8	NP	10	382	120000	0.002113	NP	0.046628
20862	9	9	9	21	505000	0.001723	0.042886	0.050038
7432	10	NP	NP	216	170000	0.001073	NP	NP
748858	NP	5	7	200	179000	NP	0.049025	0.058020
848692	NP	8	8	260	152000	NP	0.059467	0.065401

Table 3. Results comparison of Page Rank and HITS of C# Network

Id	Page Rank Order	HITS Hub Order	HITS Authority Order	Stack Overflow Position	Reputation Score	Page Rank Score	HITS Hub Score	HITS Authority Score
22656	1	1	1	1	1003378	0.005661	0.491652	0.493822
23354	2	2	2	6	713789	0.002949	0.2221112	0.222839
17034	3	3	3	5	735810	0.002612	0.166809	0.167278
29407	4	4	4	2	777525	0.001949	0.135675	0.136151
284240	5	7	7	58	327772	0.001740	0.074243	0.074407
34397	6	5	5	9	622520	0.001523	0.111772	0.112100
1197518	7	NP	NP	233	169010	0.001350	NP	NP
65358	8	6	6	38	439516	0.001308	0.093423	0.093720
1159478	9	10	10	234	168454	0.001239	0.061802	0.062004
470005	10	NP	NP	222	172725	0.001207	NP	NP
60761	NP	8	8	181	198383	NP	0.069089	0.069310
88656	NP	9	9	25	485411	NP	0.063738	0.063870

Table 4. Results comparison of Page Rank and HITS of JavaScript Network

ID	Page Rank Order	HITS Hub Order	HITS Authority Order	Stack Overflow Position	Reputation Score	Page Rank Score	HITS Hub Score	HITS Authority Score
157247	1	1	1	12	597109	0.002907	0.469888	0.490059

19068	2	4	4	13	577781	0.001854	0.161084	0.166907
816620	3	3	3	46	369173	0.001796	0.168041	0.172987
1048572	4	2	2	74	303170	0.001781	0.192908	0.197708
114251	5	6	6	73	303322	0.001716	0.123545	0.127746
182668	6	5	5	80	290760	0.001279	0.137493	0.140782
965051	7	8	8	121	244904	0.001277	0.096625	0.099242
218196	8	7	7	28	482405	0.001085	0.108510	0.111535
1491895	9	NP	NP	48	359136	0.001082	NP	NP
519413	10	NP	NP	162	213179	0.001048	NP	NP
263525	NP	9	9	116	249215	NP	0.079111	0.080762
14104	NP	10	10	303	139797	NP	0.056783	0.058558

Table 5. Results comparison of Page Rank and HITS of Java Network

ID	Page Rank Order	HITS Hub Order	HITS Authority Order	Stack Overflow Position	Reputation Score	Page Rank Score	HITS Hub Score	HITS Authority Score
22656	1	1	1	1	1003378	0.003030	0.358847	0.361963
992484	2	6	6	87	279106	0.002949	0.117175	0.118082
571407	3	2	2	27	483205	0.002293	0.218517	0.220037
139985	4	4	4	33	462510	0.002132	0.186818	0.188057
57695	5	3	3	40	406546	0.002129	0.209333	0.210604
522444	6	9	9	112	252709	0.002109	0.097593	0.098327
131872	7	10	10	105	257631	0.001979	0.097069	0.097759
157882	8	5	5	3	777442	0.001633	0.144975	0.145831
1221571	9	NP	NP	144	231087	0.001627	NP	NP
207421	10	NP	NP	133	239397	0.001604	NP	NP
438154	NP	7	7	194	187085	NP	0.108086	0.108475
203907	NP	8	8	37	445069	NP	0.104872	0.105373

Discounted Cumulative Gain (DCG)

Measuring quality of top experts predicted is an important property to determine the effectiveness of the results. Kalervo Jarvelin and Jaana Kekalainen in [21] discuss about DCG, relevance score assigned to top experts predicted to find the usefulness of the expert based on its order in the results. The gain is calculated based on summation of gains for each result predicted with gains discounted for each result which are present at lower position in the list of results. Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Wei Chen, Tie-Yan Liu in [22] found out by ordering the results ideally in decreasing order to reduce the value to the range 0 and 1. DCG is normalized to form Normalized Discounted Cumulative Gain (NDCG) to compare DCG with Ideal Discounted Cumulative Gain (IDCG) for ideal order (decreasing order) of top experts based on relevance score. Equations 2 and 3 are used for the calculations

DCG (Discounted Cumulative Gain) Computation:

$$DCG_k = \sum_{i=1}^{k} \frac{\text{relevance\_score}_\text{i}}{\log_2(\text{i}+1)} \tag{2}$$

k, top k experts predicted by PageRank and HITS

relevance_score $_i$, Relevance score of i^{th} expert

NDCG (Normalized Cumulative Gain) Computation:

$$NDCG = \frac{DCG_k}{IDCG_k} \tag{3}$$

DCG $_k$ - Discount Cumulative Gain for k experts

IDCG $_k$ – Ideal Discounted Cumulative Gain for k experts

Table 6. Assumption for relevance functions for nodes

	0- not relevant	1-lesser relevant	2-more relevant	3- highly relevant
relevance_func_1000	0 to 100	101 to 500	501 to 1000	>1001
relevance_func_5000	0 to 1000	1001 to 3000	3001 to 5000	>5001

Following are the assumptions carried out to find the measure of correctness. Relevance of each node within the range of 0-3 is considered as depicted in table 6. The results are tabulated in the table 7 and table 8, the values of the measure are close to 1. This infers that the measure of ranking is correct.

Table 7. DCG, NDCG, IDCG computation of top 50 values for various Networks of relevance_func_1000

Domain	DCG (PageRank)	DCG (HITS Hub)	DCG (HITS Authority)	IDCG	NDCG (PageRank)	NDCG (HITS Hub)	NDCG (HITS Authority)
Python	35.707	32.448	33.431	35.707	0.999	0.908	0.936
C#	36.258	34.166	34.168	36.362	0.997	0.939	0.939
JavaScript	37.175	33.165	33.386	37.196	0.999	0.891	0.897
Java	37.161	34.858	34.868	37.198	0.998	0.937	0.937

Table 8. DCG, NDCG, IDCG computation of top 50 values of various Networks of relevance_func_5000

Domain	DCG (PageRank)	DCG (HITS Hub)	DCG (HITS Authority)	IDCG	NDCG (PageRank)	NDCG (HITS Hub)	NDCG (HITS Authority)
Python	11.869	11.424	11.508	11.895	0.997	0.960	0.967
C#	12.946	12.754	12.754	13.05	0.992	0.977	0.977
JavaScript	14.294	12.918	12.928	14.317	0.998	0.902	0.903
Java	14.667	13.571	13.577	14.704	0.997	0.923	0.923

The top 10 experts for various Networks that NDCG values were 1 for all the four domains. The value implies that the results predicted are relevant. The computation for top 50 experts depicted in tables 7 and 8 were close to be perfect with NDCG value of above 0.90 which implies that the results predicted were closely relevant. The computation of DCG for results predicted by HITS are lesser compared to IDCG (Ideal Discounted Cumulative Gain), implies that one or more experts with higher relevance score is at lower position than its ideal position based on its Relevance Score. It was also found that there were many irrelevant users with relevance value of 0 (in-degree of less than 100 in case of relevance_func_1000 and in-degree of less than 1000 in case of relevance_func_5000).

5. Conclusion

Social Network Analysis on Stack Overflow is performed; networks are constructed for the chosen topics. Top experts of specific domain Java, JavaScript, C#, and python were found out by applying PageRank, HITS algorithms on the network constructed. The Top 10 experts found are compared with the stack overflow results. The results say that some of the users are experts in multiple domains. The order of the top 10 experts of four domains is analyzed using DCG algorithm to find out the correctness of the results. Employers can check for the individual domain score to recruit job seekers of a specific expertise. An analysis of finding the top experts for unpopular domains or non-trending domains could also be found out. Further Stack Overflow data can be used in analyzing Stack Overflow posts to predict popularity of the tags (domains) on the community over the years. When users query for a question in a specific domain, an optimized result can be provided by the expert's answers, thus reducing the search. Languages such as Perl, Delphi, Ruby, VBA, Objective-C, are found to be fastest-shrinking in Stack Overflow activity among the developer community. The top experts in unpopular domains can be easy to determine as the community is smaller compared to popular domains. The search results obtained are more relevant and close to the questions posted by the users. Social network analysis provides various patterns in the social network communities which can be used to analyze the user's involvement. Patterns can be found on how top experts react to a question raised by an amateur (lesser in-degree node). Similarly how an amateur reacts to an experts questionnaire. The top expert's correction behaviors for a generic post, their voting pattern to an answer can also be determined.

References

[1] Leskovec, Jure. "Social media analytics: tracking, modeling and predicting the flow of information through networks."Proceedings of the 20th international conference companion on World wide web. ACM, 2011.
[2] Wang, Gang, et al. "Wisdom in the social crowd: an analysis of quora." Proceedings of the 22nd international conference on World Wide Web. ACM, 2013.
[3] Abdalkareem, Rabe, Emad Shihab, and Juergen Rilling. "What do developers use the crowd for? a study using stack overflow." IEEE Software 34.2 (2017): 53-60.
[4] Leskovec, Jure, Jon Kleinberg, and Christos Faloutsos. "Graphs over time: densification laws, shrinking diameters and possible explanations." Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005.
[5] Yang, Jaewon, and Jure Leskovec. "Defining and evaluating network communities based on ground-truth." Knowledge and Information Systems 42.1 (2015): 181-213.
[6] Chen, Chunyang, and Zhenchang Xing. "Towards correlating search on google and asking on stack overflow." Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual. Vol. 1. IEEE, 2016.
[7] Opsahl, Tore, Filip Agneessens, and John Skvoretz. "Node centrality in weighted networks: Generalizing degree and shortest paths." Social networks 32.3 (2010): 245-251.
[8] Vasilescu, Bogdan, Andrea Capiluppi, and Alexander Serebrenik. "Gender, representation and online participation: A quantitative study of stackoverflow." Social Informatics (SocialInformatics), 2012 International Conference on. IEEE, 2012.
[9] Movshovitz-Attias, Dana, et al. "Analysis of the reputation system and user contributions on a question answering website: Stackoverflow." Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. ACM, 2013.
[10] Bosu, Amiangshu, et al. "Building reputation in stackoverflow: an empirical investigation." Proceedings of the 10th Working Conference on Mining Software Repositories. IEEE Press, 2013.
[11] Ponzanelli, Luca, et al. "Improving low quality stack overflow post detection." Software Maintenance and Evolution (ICSME), 2014 IEEE International Conference on. IEEE, 2014.
[12] Campos, Eduardo Cunha, and Marcelo de Almeida Maia. "Automatic categorization of questions from Q&A sites."Proceedings of the 29th Annual ACM Symposium on Applied Computing. ACM, 2014.
[13] Lezina, Galina E., and Artem M. Kuznetsov. "Predict closed questions on stackoverflow." (2013).
[14] Stanley, Clayton, and Michael D. Byrne. "Predicting tags for stackoverflow posts." Proceedings of ICCM. Vol. 2013. 2013.
[15] Wadhwa, Pooja, and M. P. S. Bhatia. "Social Networks analysis: Trends, techniques and future prospects." (2012): 1-6.
[16] Kumar, Varun, and Niranjan Pedanekar. "Mining Shapes of Expertise in Online Social Q&A Communities." Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion. ACM, 2016.
[17] Leskovec, Jure, and Rok Sosič. "Snap: A general-purpose network analysis and graph-mining library." ACM Transactions on Intelligent Systems and Technology (TIST) 8.1 (2016): 1.
[18] Zhang, Jun, Mark S. Ackerman, and Lada Adamic. "Expertise networks in online communities: structure and algorithms."Proceedings of the 16th international conference on World Wide Web. ACM, 2007.
[19] Page, Lawrence, et al. The PageRank citation ranking: Bringing order to the web. Stanford InfoLab, 1999.
[20] Kleinberg, Jon M. "Authoritative sources in a hyperlinked environment." Journal of the ACM (JACM) 46.5 (1999): 604-632.
[21] Kalervo Jarvelin, Jaana Kekalainen. "Cumulated gain-based evaluation of IR techniques." ACM Transactions on Information Systems, 2002.
[22] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Tie-Yan Liu, Wei Chen. "A Theoretical Analysis of NDCG Type Ranking Measures." Journal of Machine Learning Research, 2013