Email Social Network Extraction and Search

Michal Laclavík, Štefan Dlugolinský, Marcel Kvassay, Ladislav Hluchý
Institute of Informatics, Slovak Academy of Sciences
michal.laclavik@savba.sk

Abstract

The article discusses email search prototype, which exploits social networks hidden in email archives and spread of activation algorithm. Prototype offers new way of searching email archives as knowledge repository. The prototype was partially evaluated on Enron email corpus.

1. Introduction

Email communication analysis allows the extraction of social networks with links to people, organizations, locations, topics or time. Social Networks included in email archives are becoming increasingly valuable assets in organizations, enterprises and communities, though to date they have been little explored.

Social networks in the area of email communication have been studied to some extent. Communication on the Apache Web Server mailing lists and its relation to CVS activity was studied in [1]. Extracting social networks and contact information from emails and the Web and combining this information is discussed in [2]. Another research effort [3] exploits social networks to identify relations, and tests the proposed approaches on the Enron corpus. We are using a similar approach to that of IBM Galaxy [4] in the Nepomuk[1] project, where the concept of multidimensional social network was introduced. Similar spread of activation inference was also tested on Wikipedia [7].

In the article we present a new approach for email search exploiting information extraction, graphs or networks and spread of activation. We build on our previous work [5] [6], where we have presented an approach for the extraction, spreading activation and we have also evaluated the extraction and inference algorithm with satisfactory results. In this paper we discuss Email Social Network Search prototype, which uses the same algorithm as described in [5] and [6] but user interface for controlling the search was built and we also performed scalability evaluation on a part of the Enron email corpus[2].

2. Extraction of Social Network Graph

In order to provide the social network graph hidden in the email communication, the important task is to identify objects and object properties in the email and thus to formalize the message content and context. For object identification we use the information extraction (IE) techniques [6] based on regular expression patterns and gazetteers[3] as can be seen in Figure 1.

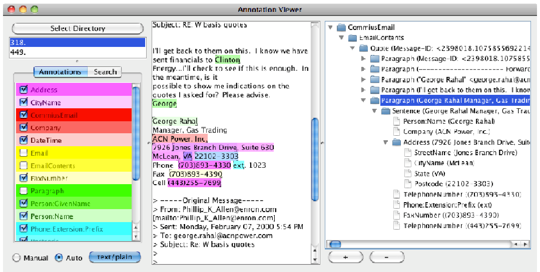

Figure 1. User interface of Ontea [6] information extraction tool with highlighted extracted objects (middle) and tree structure (right) which is used to built social network graph (Figure 2).

Patterns and gazetteers extract key-value pairs (object type – object value) from email textual content as seen in middle of Figure 1. If there is textual data presented in binary form (e.g. PDF attachment) it is, if possible, converted to text before the information extraction begins. Extracted key-value pairs are then used to build the tree (see Figure 1 on right side) and the graph of social network as can be seen in Figure 2.

[1] http://nepomuk.semanticdesktop.org/

[2] http://www.cs.cmu.edu/~enron/

[3] Gazetteer is simply the list of keywords (e.g. list of given names) representing an object type, which are matched against the text of emails.

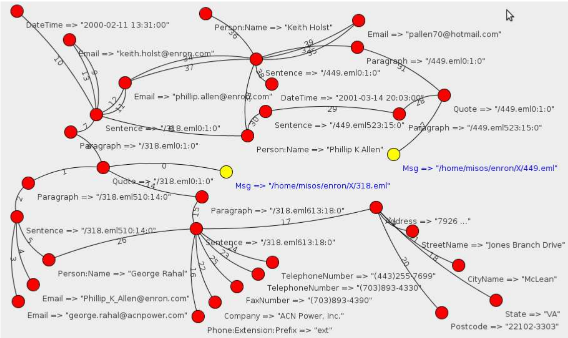

Figure 2. Social Network Graph built from 2 Enron emails. Please note that email or person name connects these 2 emails. Graph also contains sentence and paragraph nodes.

3. Email Social Network Search Prototype

Email Social Network prototype exploits email analysis and extraction in the form of key-value pairs [6] (Figure 1, middle), semantic trees (Figure 1, right side) as well as their interconnection into a graph of social network [5] (Figure 2). It also exploits the spreading activation inference algorithm discussed in [5].

The idea is that we can infer relations of activated object using spread of activation in social network graph. For example we can discover related objects for person as seen on Figure 3.

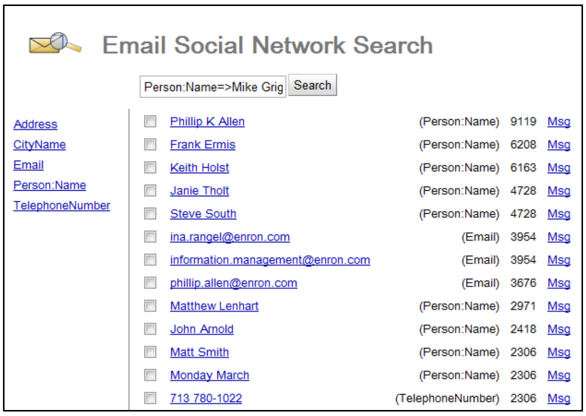

Figure 3. Email Social Network Search user interface. Relevant objects of various types returned for person.

The Email Social Network Search Module enables user to search real world objects mentioned in email or email archive, and their relations. Exploiting the prototype, user can discover personal email addresses, telephone numbers, company names, organizational names and so on. When user searches for the object and accesses it, all the objects related to the searched object (i.e. person names, phone numbers, organizations, addresses) are shown as results and the user can navigate deeper into the object graph by clicking on any object. For instance, user can select a person's name to get the company where this person works, then select the company to get its phone number or address, and so on.

Figure 3 and 4 shows the user interface of the Email Social Network Search prototype. Its input is a key-value pair representing the business object (in this case, person) extracted from the email. When pressing the search button, the graph node in search box is activated and spreading activation algorithm is executed on the social network graph. The algorithm returns all the relevant objects (key-value pairs) for the searched object (Figure 3). For example, it returns organization names, email addresses, postal addresses, telephone numbers or person names. Figure 4 shows the restriction of the search to return only the objects of a certain type (type is defined by the key of the key-value pair). In this case we wanted to get the phone number of the person, which was quite low in the main (mixed-type) list, so the search was further restricted to return only the relevant phone numbers for this person by clicking on the type TelephoneNumber in the left panel of GUI. In addition we can click on "Msg" link next to return object in order to validate search result. For example the prototype can return telephone number of other person (possibly somehow related) if search person does not have any phone in the email archive. Requested object is then highlighted in returned message. Returned message is also inferred using spread of activation, because archive can contains hundreds of messages with requested object, so we try to infer most relevant message.

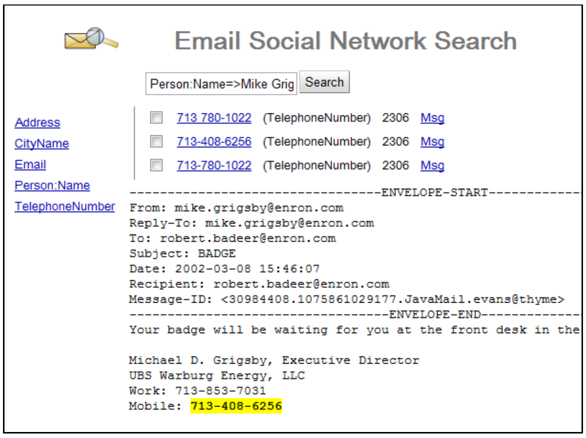

Figure 4. Email Social Network Search user interface. Same results as in Figure 3 restricted for phone numbers only. Phone numbers of desired person are displayed. Inferred objects (in this case phone numbers) are highlighted in concrete email message by clicking on "Msg" link.

The search algorithm can also be improved by allowing the users to delete the wrongly extracted objects or to connect the various aliases of the same object (e.g. the same company or personal names spelled differently or written differently) as seen in Figure 5. Such user feedback enables the search to learn and return better results in future. For example, if we merge 3 selected person name aliases seen in Figure 5, there will be better results (e.g. phone number, address or organization) returned for any of the aliases.

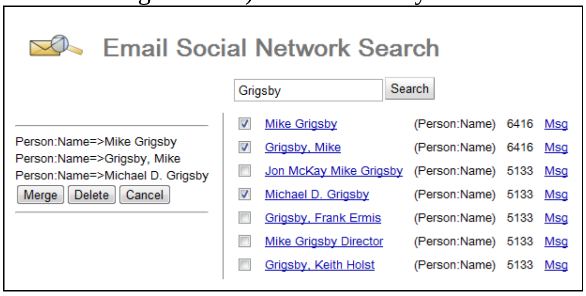

Figure 5. Prototype GUI with displayed results of full-text search. In addition, objects are selected for possible merge or delete. When these objects are merged, search returns better results for any of the object aliases.

In the future, we also plan to extend the prototype to let users annotate their own key-value pairs in the emails. (using the Email Social Network Search GUI). In this way they can later easily search for the additional valuable information hidden in the emails, such as passwords, links or documents. In addition annotated values will be search also in other emails where they also appear. An interesting feature would be to extend the search to include the attachments. We were not able to do it on Enron corpus since it does not include attachments, but our prototype can now identify objects in them.

The prototype and video explaining search features is available at http://ikt.ui.sav.sk/esns/ and it works on several Enron mailboxes.

4. Evaluation

In [5] we have evaluated success (precision and recall) of the information extraction and the spreading activation algorithms. The prototype and algorithm was further fine-tuned with focus on higher precision of results. In this section we discuss mainly scalability of the algorithm.

Our hypothesis was that the performance (search time) should be stable even with large graphs, because we always activate only a small portion of the graph. This was found to be valid only to some extent. One problem is that the created social network graph has properties of small world networks. For example, in a similar work performed on Wikipedia graph [7], spreading activation could be performed only in 2 iterations because otherwise it would visit too many nodes. In [5] we have used 30 iterations, but on large graphs the impact on performance was too high. Now we set up number of iterations to 4 experimentally. It seems to have no or minimal impact on the relevance of the returned results. The second problem is the implementation of our algorithm, which even with 4 iterations seems to visit too many nodes without firing the value and it visits the same nodes several times.

Table 1. Email Social Network Search performance evaluation on 5 datasets.

Number of Mailboxes	1	5	7	10	15
Number of Emails	3 033	9 939	20 521	36 532	50 845
Number of Verticles	41812	159 776	369 932	608 146	835 025
Number of Edges	98566	380 254	971 929	1 796 403	2 514 031
Processing time (ms)	81 672	430 025	1 199 463	1 948 847	2 680 171
Processing time (minutes)	1	7	20	32	45
One Email processing time	27	43	58	53	53
Person:Name=>Mike Grigsby					
Search Response Time	**144**	**446**	**758**	**1 396**	**1 696**
Results	344	463	494	781	761
Fired	6 363	20 732	19 045	23 466	23 839
Visited	112 280	281 060	476 324	939 642	1 174 400
Visited Unique	18 382	53 772	82 219	145 192	178 829
Search Slowed down x Times	**1**	**3,1**	**5,3**	**9,7**	**11,8**
Fired x Times	**1**	**3,3**	**3,0**	**3,7**	**3,7**
Number of messages x Times	1	3,3	6,8	12,0	16,8
Number of verticles x Times	1	3,8	8,8	14,5	20,0
Number of edges x Times	1	3,9	9,9	18,2	25,5
TelephoneNumber=>713 780-1022					
Search Response Time	**5**	**8**	**8**	**12**	**13**
Results	4	4	4	4	4
Fired	116	150	157	181	183
Visited	6 318	8 776	9 550	13 424	14 710
Visited Unique	698	954	1 059	1 424	1 513
Search Slowed down x Times	**1**	**1,5**	**1,6**	**2,3**	**2,5**
Fired x Times	**1**	**1,3**	**1,4**	**1,6**	**1,6**
Number of messages x Times	1	3,3	6,8	12,0	16,8
Number of verticles x Times	1	3,8	8,8	14,5	20,0
Number of edges x Times	1	3,9	9,9	18,2	25,5
Address=>6201 Meadow Lake, Houston, TX 77057					
Search Response Time	**7**	**14**	**28**	**40**	**59**
Results	23	38	71	91	170
Fired	236	515	701	896	1 546
Visited	8 134	15 571	32 336	40 563	58 571
Visited Unique	1 097	1 952	6 526	8 029	11 295
Search Slowed down x Times	**1**	**2,1**	**4,3**	**6,0**	**8,9**
Fired x Times	**1**	**2,2**	**3,0**	**3,8**	**6,6**
Number of messages x Times	1	3,3	6,8	12,0	16,8
Number of verticles x Times	1	3,8	8,8	14,5	20,0
Number of edges x Times	1	3,9	9,9	18,2	25,5
Email=>ina.rangel@enron.com					
Search Response Time	**106**	**552**	**1 162**	**2 156**	**3 017**
Results	732	1 764	2 668	2 809	2 952
Fired	5 165	16 062	17 629	19 716	20 997
Visited	91 199	369 584	865 300	1 694 065	2 326 867
Visited Unique	13 355	54 987	81 757	134 876	168 955
Search Slowed down x Times	**1**	**5,2**	**11,0**	**20,3**	**28,5**
Fired x Times	**1**	**3,1**	**3,4**	**3,8**	**4,1**
Number of messages x Times	1	3,3	6,8	12,0	16,8
Number of verticles x Times	1	3,8	8,8	14,5	20,0
Number of edges x Times	1	3,9	9,9	18,2	25,5

As you can see in Table 1, we have tested search response time on 5 datasets from Enron emails. Datasets contained from 3,000 up to 50,000 emails. The biggest dataset contained of 800 thousands nodes and 2.5 million of edges.

The graph was loaded in memory. We have used Jung[4] library for graph representation. We plan to test email search also with SGDB which will allows us to work with whole Email corpus or larger email archives and also test it with different version of spreading activation algorithm implementation. We would like to also exploit new ways for large graph processing such as Pregel [9], if some open source implementation will become available.

We have performed the same searches on all 5 datasets for 4 different objects: person, telephone number, address and email address as seen in Table 1. The response time was computed as the average from 3 searches. We expected search time will not increase much with number of nodes in graph, but this was valid partially only for phone number and address, which nodes were deeper in graph with less connecting edges. Nodes such as emails or person names appear in emails more often and thus activation with 4 iteration activated (and especially visited) too many nodes. We believe that the problem is also implementation of our algorithm. You can notice that fired and unique visited nodes count does not increase so much with increase of nodes and edges, but processed nodes increase quite heavily. We believe this can be overcome in better implementation and thus significantly decrease the performance (search time). In addition, We probably need to change number of iteration of spread of activation depending on topology of graph near activated nodes, e.g. to allow less iterations with nodes were too many edges are present.

5. Conclusion

In this paper we discussed new approach for email search exploiting Email Social Networks based on spread of activation and information extraction. We have tested and partially validated prototype on Enron email corpus. We believe that the developed search interface, which allows user interaction with social network graph, offers innovative ideas for searching email archives as knowledge repository. The search response time increases with number of emails and nodes in the graph in some cases. We have tested it with 5 datasets from 3,000 upto 50,000 emails, where on some objects (person, email) response time was slow and not so satisfactory, but on other object types such as phone or address the response time kept very good. We have suggested possible improvements which will be part of our future work. We believe we can improve the response time through fine-tuning the spread of activation algorithm, which should not visit the same nodes several times without firing the value. The prototype and video is available at the web for possible user testing and feedback.

[4] http://jung.sourceforge.net/

Acknowledgments

This work is supported by the projects ITMS: 26240220029 and VEGA 2/0184/10.

10. References

[1] Bird, C., Gourley, A., Devanbu, P., Gertz, M., Swaminathan, A., “Mining Email Social Networks”, In: *MSR ’06: Proceedings of the 2006 Workshop on Mining Software Repositories.* ACM, New York (2006) 137–143.

[2] Culotta, A., Bekkerman, R., McCallum, A.: “Extracting Social Networks and Contact Information from Email and the Web”. In: *CEAS’04*. http://www.ceas.cc/papers-2004/176.pdf

[3] Diehl, C. P., Namata, G., Getoor, L., “Relationship Identification for Social Network Discovery” In: *The AAAI 2008 Workshop on Enhanced Messaging, AAAI Conference On Artificial Intelligence*, pp 546-552 (2008)

[4] Judge, J., Sogrin, M., Troussov, A.: “Galaxy: IBM Ontological Network Miner” In: *Proceedings of the 1st Conference on Social Semantic Web*, Volume P-113 of Lecture Notes in In-formatics (LNI) series (ISSN 16175468, ISBN 9783-88579207-9). (2007)

[5] Laclavík M., Kvassay M., Dlugolinský Š., Hluchý L.: Use of Email Social Networks for Enterprise Benefit; In: International Workshop on Computational Social Networks (IWCSN 2010), IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2010, pp 67-70, DOI 10.1109/WI-IAT.2010.126

[6] Laclavík M., Dlugolinský Š., Šeleng M., Kvassay M., Gatial E., Balogh Z., Hluchý L.: Email Analysis and Information Extraction for Enterprise Benefit; In Computing and Informatics, 2011, vol. 30, no. 1, p. 57-87. ISSN 1335-9150

[7] Ciglan M., Nørvåg K.: WikiPop - Personalized Event Detection System Based on Wikipedia Page View Statistics (demo paper), Proceedings of CIKM'2010, Toronto, Canada, October 2010.

[8] Ciglan M., Nørvåg K.: SGDB - Simple graph database optimized for activation spreading computation, Proceedings of GDM'2010 (in conjunction with DASFAA'2010)

[9] Malewicz G. et al.; Pregel: a system for large-scale graph processing, SIGMOD '10 Proceedings of the 2010 international conference on Management of data, 2010, DOI: 10.1145/1807167.1807184