A New Architecture for Email Knowledge Extraction

Majdi Beseiso[1], Abdul Rahim Ahmad[2], Roslan Ismail[2]

[1]Yanbu University College, Yanbu, Saudi Arabia
majdibsaiso@yahoo.com
[2]Universiti Tenaga Nasional (UNITEN), Selangor, Malaysia
{abdrahim, roslan}@uniten.edu.my

ABSTRACT

The Semantic Web was designed to represent the enormous data that is existing on the World Wide Web in a machine readable format. The research shows the long period of time that was spent on the Emails for communication and information exchange. Adding the semantics to the existing Email systems could not only provide for the valuable usage of time and resources, but also refreshes the meaning of Email communication. The presented research work examines the ontology extraction process from the Email systems adopting scalable pattern rules that is based on the extracted techniques. The proposed architecture is designed to handle the unstructured Emails and the ontologies that are extracted from the Email which is divided into four main components as follows: the Ontology Learning Component, the Management Component, the Semantic Email Component and the Client Side Plugin.

KEYWORDS

Ontology Learning, Ontology Extraction, Semantic Web, Semantic Email

1. INTRODUCTION AND RESEARCH CHALLENGES

The Email is one of the essential feature of the internet, that is being utilized by a great number of users. Because more of the business and personal activities rely on Email services, users are faced with too much information [1]. The manual categorization of the Emails decide how to act upon the Emails' time consuming. Users usually get distracted from the current tasks [2]. With the development of Email services and clients, there are improvements of handling large Email archives and offering searching and tagging functionalities. Some Email services could also integrate with other systems, such as Microsoft Sharepoint1, to organize collaborative tasks. However, these functionalities are still not mature in that most of the time, tedious manual work is still inevitable especially when the Email archive is large and not well organized.

The Semantic Web [3] was designed to represent enormous data existing on the World Wide Web in a machine readable format using technologies such as Resource Description Framework (RDF) [4], Web Ontology Language (OWL) [5] and SPARQL [6]. Recent research about ontology learning [7], which is a subtask of the information extraction (IE) using the semantic Web technologies has improved ontology extraction from various resources such as XML, database, spreadsheet, etc. Ontology-based IE [8, 9] which is based on Natural Language Processing (NLP) and machine learning could help us in extracting knowledge from plain text or even from Web pages. We believe that ontology learning and IE together could help to solve the Email overload and task collaborative problems for users by providing automatic semantic annotations for

[1] http://sharepoint.microsoft.com/en-us/pages/default.aspx

DOI : 10.5121/ijwest.2012.3301

Emails. When the Emails are semantically annotated, agents in Email clients, as well as end users, could process more effective search and reasoning over the knowledge extracted from the Emails. Different kinds of knowledge could be extracted from the Emails such as, scheduling a meeting, technical request, task collaboration, legal documents, etc [10]. The main challenge in this work is that the content of the Email may involve different domains. There are sophisticated methods to extract ontology from a specific domain, for instance biomedical ontologies [11]. Some are using more generic tools such as WordNet [12] for IE. However, the knowledge in Emails falls into the middle of the "specific domain" and the "totally generic". We can expect some dozens of categories for the purposes of Emails depending on the users' activity. So, the specific domain ontology will not be sufficient to cover all the aspects, while the generic ontologies on a very high abstract level cannot extract enough information. In this paper, we need to strike a balance between the specific domain and the generic one.

The organization of this paper goes as follows: firstly, reviewing the state of art of ontology learning and the semantic Emails (Section 2). Then, will propose the basic theory regarding the Email ontology learning (Section 3). Then, the architecture will be presented (Section 4). (Section 5 & 6) give the evaluation and the conclusion.

2. LITERATURE REVIEW

Ontology is the conceptualization of the specifications [13], in which a specific group or people share mutual understanding in the domain of interest. Ontology plays the central role in the ontology learning process and the semantic annotations for Emails.

The notion of the Semantic Email is referred to as “an Email message consisting of a structured query coupled with a corresponding explanatory text” [14]. The semantic Email can be seen as an approach to manifest the information contained in the Email systems by adding the semantic annotations. Ontology has been widely used for Email classification [15,16] and spam filtering [17]. Some researches bring Speech Act theory [18] into the Email so that the Emails could be categorized according to the senders tension. W. Cohen et al. have developed an ontology of Email Acts [19] to capture and coordinating joint activities. Another direction of the semantic Email is to integrate it with task management system. U. Riss et al. [20] implement a MS Outlook plugin which allows users to start task management activities directly from the Emails. The plugin combines the semantic Web technologies, Email and task management applications together. Another application of Speech Act Theory is the sMail (Semantic Mail) Conceptual Framework [21]. The sMail summarizes Email content into a number of pre-defined sMail Speech Act Models, including Speech Act Model, Speech Act Process Model and Speech Act Process Flow Model. With the help of these models, the lifecycle of workflows can be detected and processed through the sending and receiving the Emails.

Semanta[22] has been integrated with the Social Semantic Desktop (SSD) [23]. The Social Semantic Desktop is different from the Semantic Web in that, the Social Semantic Desktop tries to apply the semantic Web technologies to the personal com-puters. NEPOMUK (Networked Environment for Personal Ontology-based Manage-ment of Unified Knowledge) is the metadata library for SSD. It aims to make possible Web style associations for the resources in PCs, so that these resources could inter-connect with each other even across PCs and other users. NEPOMUK Message Ontology (NMO) extends the NEPOMUK Information Element Framework into the domain of messages, such as Emails and instant messages. The purpose of NMO is to model the structure of messages via ontology, so that the messages could be semantically linked to other domain ontologies.

3. THE THEORY

The basic idea in this research, is to represent the ontologies obtained from different domains of the Emails that are using rule based approach in terms of concepts and the relations in OWL. This section will introduce the basic theory of the multiple domain ontologies design and the pattern rules usage for ontology extraction.

3.1. Ontology Design for Multiple Domains

Some concepts must be included in the proposed Email ontology. At first, the ontology should describe the structure of the Email such as the sender, the receivers of the Emails, the subject and the content body. The next step, the ontology must include the domain specific information so that the useful information could be extracted from the Email.

In this research, we will extend NEPOMUK Message Ontology (namespace as "nmo") with domain specific ontologies in ontology learning. Twelve categories (domains) for Emails are already obtained, refer to [10]. These are: Schedule and Meetings, Technical Requests and Support, Collaboration, Business Letters and Documents, News, Press Release, Legal Documents, Announcements, Discussion and Comments, Invoices and Bills, Attachments and SPAM.

We suppose $\mathbb{D} = \{\mathbb{D}_1, \mathbb{D}_2, \mathbb{D}_3, \dots, \mathbb{D}_n\}$ as the full set of domains (or categories) that an Email can belong to. The general Email ontology contains the main Email objects like the sender, receiver, date, etc… for each Email, the instance of general Email ontology will be defined according to the NEPOMUK in addition to the specific domain ontology which is based on the Email category. Figure 1 presents the ontologies for the Email.

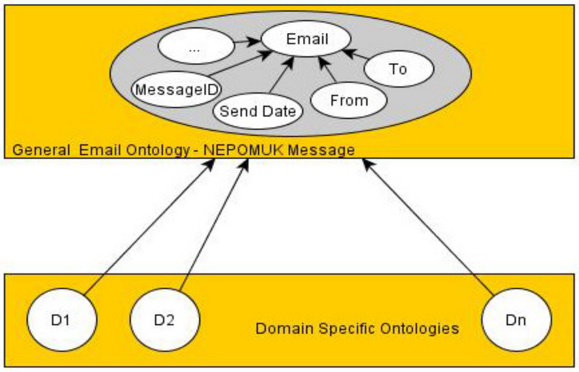
...
Email
To
MessageID
Send Date
From
General Email Ontology - NEPOMUK Message
D1
D2
Domain Specific Ontologies
Dn

Figure 1: The Email Ontology

Figure 2 illustrates the domain ontology for Email categorization. Email Categorization ontology (namespace as "ec") defines the different domains which the Emails belong to. The MeetingEmail, BusinessDocEmail, etc are the subclasses of nmo:Email and they do not disjoint, because an Email may belong to different categories. We can create new ontologies or reuse existing ontologies for each domain to model the concepts in this domain. We need to define the relationships between the Email categories and the domain-specific ontologies.

Figure 2 also gives one example for the "Meeting" domain. We reuse the existing Linked Open Descriptions of Events [24] (LODE) ontology to model the meeting events. "ec:MeetingEmail" is connected to "lode:Event" through "ec:hasEvent". The domain ontology could be generated by ontology learning models that use different means such as ontologies written by experts, reusing existing ontologies, automatic ontology extraction or hybrid methods.

The purpose of this design on the theoretical level is to separate the ontologies according to the domains. This separation is because many ontology learning and IE methods are domain specific [25, 26].

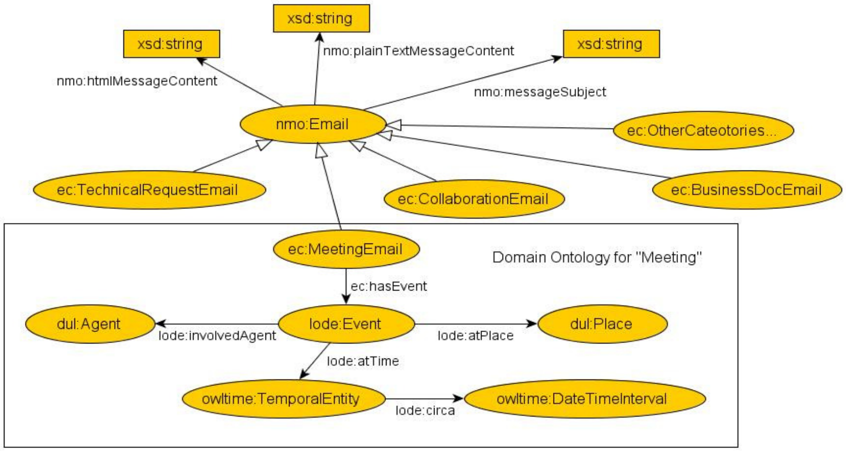

Figure 2: Example of the ontology for email categorization

3.2. Rules for Ontology Extraction

The use of rule based approach is adopted for ontology extraction. The rules are considered to be scaled for efficient information extraction. To understand the ontology extraction process using pattern rules, let us consider a content of the Email body $\mathcal{E}_{\mathrm{body}}$ represented as

$$\mathcal{E}_{\mathrm{body}} - \{\mathcal{X}_1, \mathcal{X}_2, \mathcal{X}_3, \ldots \mathcal{X}_n\}$$

Where $\mathcal{X}$ represents the strings, the delimiters, the punctuations appearing in the Email body. To extract the information, we need to extract the strings from $\mathcal{E}_{\mathrm{body}}$. A space, a new line character, tabs, and other text formatting characters are filtered to extract strings.

Now to define the rule set as

1. Let $\mathcal{X}$ be a set of character strings.
(Example: June, Hall, Room, Hotel ,Tommorow.. etc)

2. Let $\mathcal{T}$ represent a set of finite Tags
(Example :<Date:>,<Location>,<Participants>,<Agenda>)

3. The rule set of $\mathcal{X}$ is represented as
$$\mathcal{XS} = \{ \mathcal{XS}_1 \mathcal{XS}_2, \ldots\ldots\ldots\ldots\ldots\ldots\ldots \mathcal{XS}_n \}$$
The terms of $\mathcal{XS}$ may be considered as an extension set used to identify the various sections of Email content. The elements of the set $\mathcal{XS}_i$ should be the elements that are in $\mathcal{XS}_i$ but not in other sets $\mathcal{XS}_j$ for $i \neq j$.

4. Let $\mathcal{F}$ be a set of rules represented as $\mathcal{P}_i \rightarrow \mathcal{A}_i$ where $\mathcal{P}_i \in \mathcal{I}$ and each $\mathcal{A}_i$ is a string of rule such that $\mathcal{A}_i \in \mathcal{XS}$ or $\mathcal{A}_i - \mathcal{Q}_{i1} \wedge \mathcal{Q}_{i2} \wedge \mathcal{Q}_{i3} \ldots\ldots\mathcal{Q}_{in}$ where $n \geq 1$ and each $\mathcal{Q}_{ij}$ is a set of character strings $\mathcal{X}$ or a tag $\mathcal{I}$. i.e. either $\mathcal{Q}_{ij} \in \mathcal{I}$ or $\mathcal{Q}_{ij} \in \mathcal{X}$.

Up to $n-1$ rules in the rule set $\mathcal{Q}_{ij}$may be enclosed in brackets as $[\mathcal{Q}_{ij}]$ to show that they are optional in the ontology extraction process. Also, there exists at least one rule such that $\mathcal{P}_i \rightarrow \mathcal{XS}_i$ for each rule set $\mathcal{XS}_i$ and that every tag in $\mathcal{I}$ appears as the head of at least one rule. Therein, we require a strict partial order defined over $\mathcal{I}$ represented by $\subset$ such that $\mathcal{P}_i \subset \mathcal{P}_j$ if only $\mathcal{P}_j \rightarrow \mathcal{A}_j$ is a rule of $\mathcal{F}$ and either $\mathcal{A}_j$ contains $\mathcal{Q}_j$ or $\mathcal{P}_j$ contains a tag $\mathcal{P}_k$ such that $\mathcal{P}_i \subset \mathcal{P}_k$.

4. SYSTEM ARCHITECTURE

The purpose of the design is to provide a feasible architecture to resolve the challenges mentioned in Section 1 and to implement the theories of the semantic Email ontology learning based on the use of rules defined in Section 3. The architecture of the semantic Email system proposed in this paper, incorporates the rules for effective ontology extraction and integrates the ontology learning models with MS Outlook. The proposed architecture is represented in Figure3.

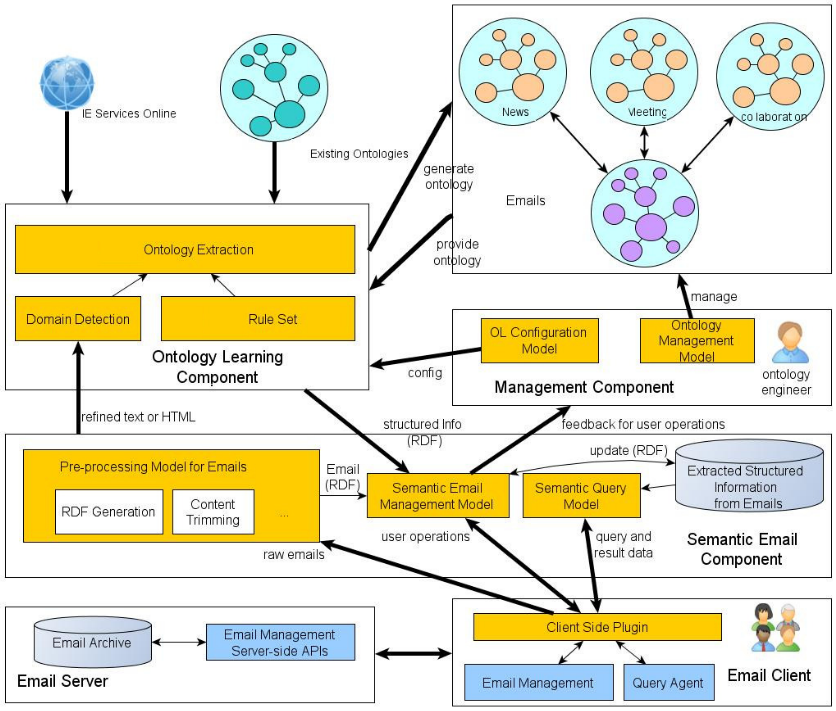

Figure 3: System Architecture

There are mainly four components in the architecture: these components are as follows: the Client Side Plugin, the Semantic Email Component, the Ontology Learning Component and the Management Component. We build an extra layer on top of the Email Server and the Client, so that the original server and the client can operate as they are. The rest of this section will explain the models within each component and the interaction among the components in general.

4.1. The Ontology Learning and the Management Component

The Ontology Learning Component is the key component for adding the semantics to the Emails. It consists of several rules in each certain library, where a set of rules will be used for a specific ontology learning process according to the domain. The Domain Detection Model will extract the domain information from the Emails to enable the system decide which set of rules should be used to extract the information.

The input of the Ontology Learning Component includes the preprocessed text from the Emails, the existing ontologies, the IE services online, the configuration about which the rules should be chosen. The extraction ontologies are originally generated from the ontology learning models. One output of the Ontology Learning Component is the structured information that is extracted from the Emails. The structured information will be saved as a RDF store for updating and querying. The Ontology Learning Component also generates or updates the extraction ontology according to the rules.

The Management Component, mainly manages the ontology learning models and the extraction ontologies. As stated in [27] and [7], during the ontology generation process, we need to maintain the life cycle of the extraction ontology. The maintenance may involve some human interactions, especially in the validation and evolution process. In the traditional ontology learning system, an ontology engineer needs to correct the wrong concepts and relationships. When new resources and user feed-back about the IE are available, the ontology engineer may change the extraction ontology. Accordingly, we developed the Ontology Management Model to carry out the maintenance of the extraction ontologies both manually and automatically.

4.2. Some Vocabularies for Extraction Ontology

Each of the twelve different domains that where mentioned before, has its own ontology. We will give several examples for reusing the existing ontologies to model concepts in these domains. Figure 4 demonstrates the use of LODE vocabulary for the "Schedule and Meetings" domain. For the "News" domain, there are many existing ontologies. The most widely used one is the rNews 1.0 [2] that was developed by the International Press Telecommunications Council (IPTC). NewsItem is the central concept in the rNews ontology and it is related to the body of news content, the date and place of publishing, the multimedia objects and the user comments, etc. So, we can relate "ec:NewsEmail" class to "NewsItem" through the use of "ec:hasNewsitem" property. For the "Discussion and Comments" domain, we can use RDF Review Vocabulary[3]. "Review" is the central class in Review ontology. The comments and feedbacks can be extracted from the body of the Email, and the reviewer can be the sender of the Emails. The review ontology also models the comment on a review so that the comments can form a discussion or conversation. The "Collaborative" and the "Technical support and request" domains could be modeled by sMail Ontology[4]. People, files, tasks, projects, which are involved in the communication of Emails, will be a part of a workflow in the ontology. The most important thing is that, this ontology can be integrated with the calendar and the task management in MS outlook.

4.3. Semantic Email Component and Client Side Plugin

To integrate the existing Email applications with the ontology learning models, we might introduce the Semantic Email Component and the Client Side Plugin to the system. The Client Side Plugin converts the user operations on the client side into the operations for the RDF store

[2] http://dev.iptc.org/rNews

[3] http://vocab.org/review/terms.html

[4] http://ontologies.smile.deri.ie/smail.rdf

and the ontology learning models. The Semantic Email Management Model also converts the extracted structured information into some operations for the Email client.

For example, if a user tags an Email as "meeting", the plugin will ask the Semantic Email Management Model to update the triples in RDF store and to give feedback to the Management Component, passing information to use the corresponding algorithm to extract the meeting information from this Email. Then the Semantic Email Management Model will ask the Client Side Plugin to send this Email through the ontology learning process and save the structured information into the RDF store. Then, and if necessary, the plugin also puts this meeting into the calendar according to the time given in the structured information. On the server side, the original functions of tagging will continue to be carried out. So, the client side plugin will not affect the original functions of the Email applications. Similarly, if a new Email is available in the inbox, the client side plugin will automatically send it through the ontology learning models and see which domain the Email belongs to. Then the Semantic Email Management Model will save the triples in the RDF store and ask the client side plugin to put the Email in an appropriate folder or to automatically annotate the Email with the domain information.

The Pre-processing Model for the Emails is to process the raw Email for the ontol-ogy learning models. This model will trim the redundant information in the Email for ontology learning models. Usually, the algorithms in ontology learning models only deal with the subject and the body of the Emails. So, some information in a standard Email such as, the sender, receivers, messageid, etc., could be ignored. However, this information will be necessary later to track the Email for updating and query. So, the Email RDFGeneration will convert the whole Email to RDF using NMO and NEPOMUK Contact Ontology (NCO) ontology, and will save it together with the extracted information in the RDF store as a big RDF graph for this Email. The RDF statements generated by the RDFGeneration only contain the basic information that could be obtained from the structure of an Email, such as the sender, the receiver, the date, etc. After then, the ontology learning models will extract information from the body and the subject of the Email. The Semantic Query Model will convert the user queries from Client Side Plugin into SPARQL queries. The resulting data will be sent back to the Client Side Plugin and the original Emails will be retrieved via the Email server..

5. EVALUATION

There are four domains considered for evaluation purposes. These domain are: "Schedules and Meetings", "News", "Review" and the "Technical support and request". The results generated by the ontology learning engine are evaluated by some domain experts. Only correctness of the extracted concepts and its relations with other concepts is approved by all domain experts. The result is marked as correct. We used recall and precision measures [28], to evaluate the performance of our method and the quality of the learned ontologies. 1200 emails were selected from the below mentioned three Email dataset corpuses for evaluation:

1. Enron Email Corpus with Categories [5]: obtained from the UC Berkeley Enron Email Analysis Project [29], mainly focusing on Business Communications eliminating personal messages.
2. British Columbia Conversation Corpus (BC3 Corpus[6]): The BC3 Corpus [30]. The BC3 corpus is a part of the W3C corpus.
3. Custom E-Mail Corpus: The data set corpus consists of Business communication Emails. This data set was collected by the researchers for this purpose.

[5] http://bailando.sims.berkeley.edu/enron/enron_with_categories.tar.gz

[6] http://www.cs.ubc.ca/nest/lci/bc3.html

The ontology engine which is used for evaluation is interfaced by the Outlook Mail Client. The Emails clustering has been based on the concepts that are extracted from the four domains. The ontology visualization demonstrates the relations between the concepts extracted as in figure 4.

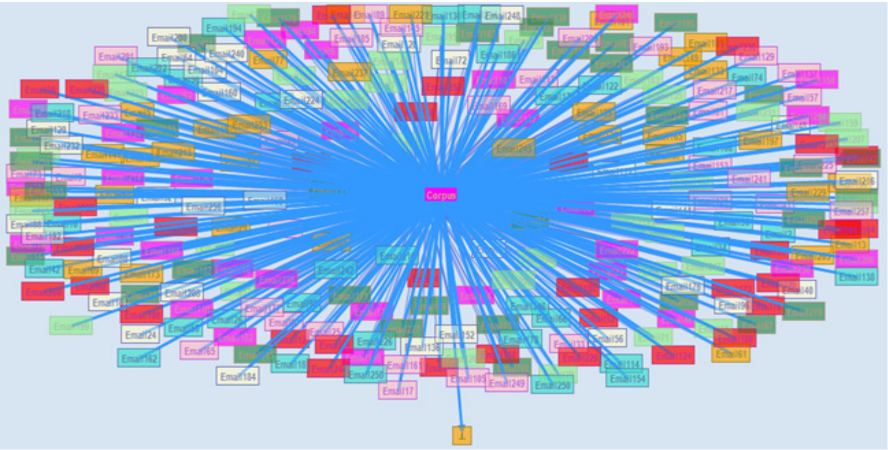

Figure 4: Visualization of email ontology

The obtained results and the extracted concepts in terms of the numbers and the relations established amongst the concepts are shown in table-1, throughout the four domain tests, precision ranged from 0.68 to 0.89 and recall from 0.61 to 0.83, as Table 2 shows.

Table 1: Summary of Results

Domain	**Schedule & Meetings**	**News**	**Review**	**Technical support**	**Overall**
No. of emails	400	300	200	300	1200
No. of extracted concepts	7687	4654	2654	5955	20950
No. of extracted relations`	5756	3022	1549	2446	12773
Recall	0.83	0.69	0.61	0.66	0.71
Precision	0.89	0.78	0.68	0.71	0.78

The obtained results, showed that the proposed architecture which is based on the rules for ontology extraction provide the accepted results with the considerable preci-sion of the ontology extraction which is coupled with the clustering. It is also observed that efficiency depends on the rule sets that define for each domain.

6. CONCLUSION

Nowadays, the Emails which are used for personal or business proposals are be-coming more and more difficult to handle. The abundance in Emails and the lack of assistance from the Email clients will certainly lead to losing tracks of information contained in the Emails. In this paper, we have explained the necessity to utilizing ontology learning processes for solving this problem. The research on the theories underpins the design of the architecture which consists of Ontology Learning Component, Management Component, Semantic Email Com-ponent and the Client Side Plugin. The architecture depicts the bridge between the ontology learning system and the existing Email clients (MS Outlook). We adopted the SSD and the NEPOMUK as the semantic desktop environment and reuse the NMO ontologies to model Emails. We also suggest several existing ontologies to extend NMO and the categories of the Emails into different domains. The implemen-tation of this will integrate the ontology learning module which supports SSD and NEPOMUK with Email client application. The final deliverable results of the imple-mentation from the end users' view point will be an MS Outlook plugin which will help the users automatically to annotate the Emails and track the activities that are based on the content of the Emails.

REFERENCES

[1] S. Whittaker and C. Sidner, "Email overload: exploring personal information management of email," presented at the Proceedings of the SIGCHI conference on Human factors in computing systems: common ground, Vancouver, British Columbia, Canada, 1996.

[2] H. Khosravi and Y. Wilks, "Routing email automatically by purpose not topic," Nat. Lang. Eng., vol. 5, pp. 237-250, 1999.

[3] T. Berners-Lee, et al. (2001) The Semantic Web. Scientific American. 34 - 43.

[4] D. Brickley and R. V.Guha, "RDF Vocabulary Description Language 1.0: RDF Schema". (B. McBride, Ed.)W3C Recommendation. W3C, 2004. Retrieved from http://www.w3.org/TR/rdf-schema/

[5] D.L. Mcguinness and F. V. Harmelen, "OWL Web Ontology Language Overview", W3C recommendation, vol. 10, pp 2004-03, 2004

[6] E. Prud'hommeaux and A. Seaborne. "SPARQL query language for RDF". Technical report, W3C, April 2006. W3C Candidate Recommendation, URL http://www.w3.org/TR/rdf-sparql-query/

[7] A. Maedche and S. Staab, "Ontology learning for the Semantic Web," Intelligent Systems, IEEE, vol. 16, pp. 72-79, 2001.

[8] A. Maedche, et al., "Bootstrapping an ontology-based information extraction system," in Intelligent exploration of the web, S. S. Piotr, et al., Eds., ed: Physica-Verlag GmbH, 2003, pp. 345-359.

[9] J. Turmo, et al., "Adaptive information extraction," ACM Comput. Surv., vol. 38, p. 4, 2006.

[10] I. Frommholz, "Email Classification and Information Extraction Methods: The Inbox Sce-nario", University of Duisburg-Essen, 2006

.

[11] P. Lambrix and H. Tan, "SAMBO-A system for aligning and merging biomedical ontologies," Web Semant., vol. 4, pp. 196-206, 2006.

[12] R. Poli, et al., Theory and Applications of Ontology: Computer Applications: Springer Publishing Company, Incorporated, 2010.

[13] T. R. Gruber, "Towards Principles for the Design of Ontologies Used for Knowledge Sharing," presented at the Formal Ontology in Conceptual Analysis and Knowledge Representation, Deventer, The Netherlands, 1993.

[14] L. McDowell, et al., "Semantic email," presented at the Proceedings of the 13th international conference on World Wide Web, New York, NY, USA, 2004.

[15] K. Taghva, et al., "Ontology-based classification of email," in Information Technology: Coding and Computing [Computers and Communications], 2003. Proceedings. ITCC 2003. International Conference on, 2003, pp. 194-198.

[16] H. Yang and J. Callan, "Ontology generation for large email collections," presented at the Proceedings of the 2008 international conference on Digital government research, Montreal, Canada, 2008.

[17] S. Youn and D. McLeod, "Efficient Spam Email Filtering using Adaptive Ontology," in Information Technology, 2007. ITNG '07. Fourth International Conference on, 2007, pp. 249-254.

[18] J.R. Searle, "A classification of illocutionary acts," Language in Society 5(01), 1976.

[19] V. R. Carvalho and W. W. Cohen, "On the collective classification of email "speech acts"," presented at the Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, Salvador, Brazil, 2005.

[20] U. V. Riss, et al., "E-mail in Semantic Task Management," in Commerce and Enterprise Computing, 2009. CEC '09. IEEE Conference on, 2009, pp. 468-475

[21] S. S. B. D. S. Handschuh, "The path towards Semantic Email: Summary and Outlook," presented at the The AAAI 2008 Workshop on Enhanced Messaging, 2008.

[22] S. Scerri, et al., "Semanta - Semantic Email Made Easy," presented at the Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications, Heraklion, Crete, Greece, 2009

[23] L. Sauermann, et al., "PIMO - A Framework for Representing Personal Information Models," presented at the Proceedings of I-MEDIA '07 and I-SEMANTICS '07 International Conferenceson New Media Technology and Semantic Systems as part of TRIPLE-I 2007, 2007.

[24] R. Shaw, et al., "LODE: Linking Open Descriptions of Events," presented at the Proceedings of the 4th Asian Conference on The Semantic Web, Shanghai, China, 2009.

[25] M. Shamsfard and A. A. Barforoush, "The state of the art in ontology learning: a framework for comparison," Knowl. Eng. Rev., vol. 18, pp. 293-316, 2003.

[26] L. Zhou, "Ontology learning: state of the art and open issues," Information Technology and Management, vol. 8, pp. 241-252, 2007.

[27] I. Bedini and B. Nguyen, "Automatic ontology generation: State of the art," In PRiSM Laboratory Technical Report. University of Versailles, 2007.

[28] C. Manning, et. al., "Introduction to Information Retrieval". Cambridge University Press, 2008.

[29] P. S. Keila and D. B. Skillicorn, "Structure in the Enron Email Dataset," Comput. Math. Organ. Theory, vol. 11, pp. 183-199, 2005

[30] J. Ulrich, et. al., "A Publicly Available Annotated Corpus for Supervised Email Summarization", AAAI08 EMAIL Workshop, Chicago, USA, 2008.