Knowledge Extraction from Professional E-mails

Nada Matta(✉), Hassan Atifi, and François Rauscher

Institute ICD/Tech-CICO, University of Technology of Troyes,
12 rue Marie Curie, CS 42060, 10010 Troyes Cedex, France
{nada.matta,hassan.atifi,francois.rauscher}@utt.fr

Abstract. Some professional e-mails contain knowledge about how actor face problem in order to realize projects. This type of knowledge is produced in cooperative activity. Representing project knowledge leads to structure link between coordination, cooperative decision-making and communication. The main objective of our work is to extract knowledge from daily work. So the main questions of our research are:

- Can we extract knowledge from professional e-mails?
- If so, which type of knowledge can be represented?
- How to link this knowledge to project memory?

We present in this paper our first work in this aim. Our hypothesis is tested on a software development application.

Keywords: Knowledge engineering · Knowledge management · Project memory · Traceability · Professional e-mails · Pragmatics analysis

1 Introduction

Currently designers use knowledge learned from past projects in order to deal with new ones. They reuse design rationale memory to face new problems. Knowledge Management provides techniques to enhance learning from the past [12]. Their approaches aim at making explicit the problem solving process in an organization. Their techniques are inherited mainly from knowledge engineering. So, we find in these approaches in one hand, models representing tasks, manipulated concepts and problem solving strategies, and in the other hand, methods to extract and represent knowledge. We note for instance MASK [14, 25] and REX [22] methods. These methods are used mainly to extract expertise knowledge and allow defining profession memories.

But, design projects involve several actors from different fields. These actors produce knowledge when interacting together and take collaborative decisions. So, it is important to also tackle this type in knowledge, which is generally volatile.

We deal, in our approach with this type of knowledge, called Project memory [24]. Project memory must represent organizational and cooperative dimension of knowledge. Current techniques used in Knowledge management, based on expert interviews are not adapted to extract these dimensions of knowledge. To tackle knowledge produced in collaborative activity, we need techniques that help to extract knowledge from

© IFIP International Federation for Information Processing 2015
E. Mercier-Laurent et al. (Eds.): AI4KM 2014, IFIP AICT 469, pp. 43–57, 2015.
DOI: 10.1007/978-3-319-28868-0_3

daily work. In this paper, we present a technique that help to extract knowledge from professional e-mails. The presented approach allows structuring extracted concepts and linking them to the project context. We use pragmatics analysis and knowledge engineering techniques for this aim.

Problem solving plays a central role in design projects. At first in the initial analysis that leads to specification documents, and then during the life-cycle of the project. This is especially true in software design projects if the development follows an agile method, with several roundtrips from design to delivery.

Project Memory focus on keeping "project definition, activities, history and results "as wrote Tourtier in [28]. Problem Solving is an essential part of design rationale as it tackles with problem definition, suggestions, and decision.

Face to face meetings are commonly used in office to manage projects and do collaborative work, but other mediated communications usages are increasing like phone, emails or instant messaging. However, information in emails is volatile, unstructured and distributed among users email boxes, making it difficult to trace and keep for corporate memory.

2 Project Memory

A project memory is generally described as "the history of a project and the experience gained during the realization of a project" [24].

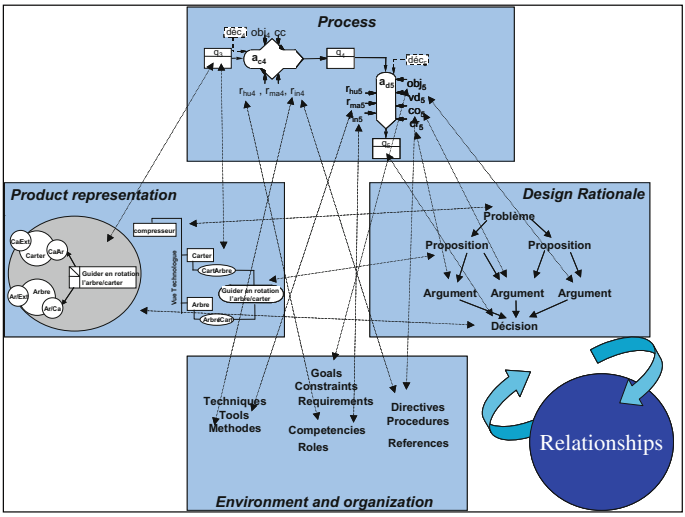


Fig. 1. Project memory

It must consider mainly (Fig. 1):

- The project organization: different participants, their competences, their organization in sub-teams, the tasks, which are assigned to each participant, etc.

- The reference frames (rules, methods, laws, …) used in the various stages of the project.
- The realization of the project: the potential problem solving, the evaluation of the solutions as well as the management of the incidents met.
- The decision making process: the negotiation strategy, which guides the making of the decisions as well as the results of the decisions.

Often, there are interdependence relations among the various elements of a project memory. Through the analysis of these relations, it is possible to make explicit and relevance of the knowledge used in the realization of the project. The traceability of this type of memory can be guided by design rationale studies and by knowledge engineering techniques.

The problem solving is part of the project design rationale memory [24]. Some technics from knowledge (e.g. REX) aims at knowledge capitalization, but others are more oriented on the traceability of the design rationale. A clear representation of the context and design rationale can be found in [3] and is presented in Fig. 2.

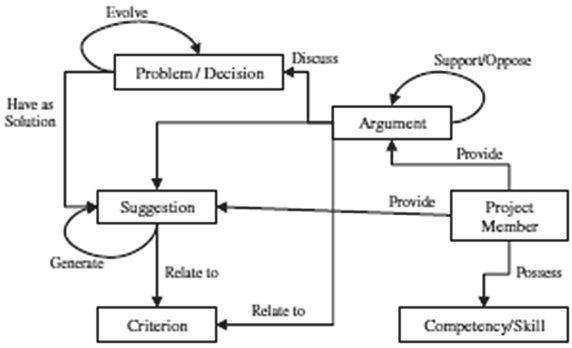


Fig. 2. Problem solving and design rationale in context.

3 Problem Solving

Theory of Human Problem Solving was developed from the work of Newell and Simon [26] and provided the basis of much problem solving research. According to Hardin in [17], "Any problem has at least three components: givens, goal and operation". This general definition from problem solving theory is bringing keys elements into light:

- Givens: information and facts presenting context;
- Goal: desired end state;
- Operations: actions to be performed to reach end state.

In our present study, related to software development, we will focus more on givens and goal, i.e. the "problem recognition" part, the operations being part of the solution. When designing software complex problem solving arise more easily, because the tasks are abstracts and often not well structured as opposed as designing a real world artifact.

Problem solving in professional project aims at transforming knowledge into business value [16]. This usually involve two types of knowledge: declarative (about facts, events, and objects) and procedural (knowing how to do things).

4 Software Development Process

Methodologies in software development evolved quickly in the last two decades from classic waterfall model to Extreme Programming and Agile methods [2].

Agile development is iterative and incremental with continuous delivery. As a side effect roundtrips between product-owner (contractor) and product-manager (developer) are more frequent, leading to increased communication and collaborative work. Typically a software design cycle in agile is divided into sprints, where the product-owner meet the product-manager (developer) and validates recent features, raises issues and express new needs. Problem Solving sequences happen on weekly (sometime daily) basis and implies all the actors of the project, not only the development team. With the new means of communication and project management methods like Kanban [19], this occurs frequently through computer mediated exchanges.

5 Pragmatic Analysis

The act of request has been extensively studied in the field of theoretical linguistics (Searle 1969), intercultural and inter-language pragmatics [4], NLP community on automated speech act identification in emails [7, 22] etc. However, as pointed by De Felice et al. [11] there is very little work concerned with data other than spoken language and few researches seem to fully respond to requirements of being sufficiently general, non-domain specific, and easily related to traditional speech acts. In addition, few researchers have focused their research requests in business written discourse (workplace email communication). Lampert et al. [20] try to create tools that assist email users to identify and manage requests contained in incoming and outgoing email. Atifi et al. [1] analyze email effectiveness from the professional's point of view by mixing two kinds of analysis: a content analysis of interviews of professionals and a pragmatic and conversational analysis of emails. De Felice et al. [11] propose a global classification scheme for annotating speech acts in a business email corpus based on traditional speech act theory described by Austin and Searle [27].

6 Related Works on Email Analysis

Several approaches study how to analyze e-mails as a specific discourse. We note for instance, tagging work [29], in which Yelati presents techniques that help to identify topics in e-mails, or the use of zoning segmentation in [21]. Other works use natural language processing in order to identify messages concerning tasks and commitment [18]. They parse verbs and sentences in order to identify tasks and they track messages between senders and receivers.

Even there is lot of work on pragmatics, which study dialogue and distinguish techniques in order to identify speech intention (Patient/doctor dialogue analysis [17]), coding dialogue scheme [8], etc. Pragmatics analysis of e-mails uses only some of these methods like ngrams analysis by Carvalho in [8], Verbal Response Mode scheme by Lampert in [21] or a custom coding scheme like De Felice et al. [11].

Techniques studying e-mails, often do not consider the context of discussions, which is important to identify speech intention. We deal with our work with professional e-mails, extracting from projects. So, we mix pragmatics analysis and topic parsing and we link this type of analysis to project context (skill and role of messages senders and receivers, project phases, and deliverables, etc.) in order to keep track of speech intention. As pragmatics analysis shows, there is not only one grid to analyze different types of speech intention. In project memory, we look for problem solving, design rationale, coordination, etc. In this study, we focus on problem solving and we build an analysis grid for this purpose.

7 Project Knowledge Extraction from Emails

The main objective of our work is to extract knowledge from daily work. So the main questions of our research are:

- Can we extract knowledge from professional e-mails?
- If so, which type of knowledge can be represented?
- How to link this knowledge to project memory?

To answer these questions, we analyze professional e-mails related to projects. In last studies, we identify a structure to analysis coordination messages [23]. Based on pragmatics analysis, we defined a grid to structure coordination messages based on the main act to do (inform, request, describe, etc.) and the objects of coordination (task, role, product, etc.). In this paper, we will go ahead and define an approach that helps to extract knowledge from professional e-mails. So, we identify firstly step by step how to isolate important messages and how to analyze them. Knowledge from e-mails, as knowledge produced in daily work, cannot be very structured. It is related closely to context. In our work, we focus on knowledge produced during project realization. We will show in our method how information from project organization help in e-mails knowledge extraction.

7.1 Classification of E-mails

Firstly, we have to identify important messages (Fig. 3). For that, we have to gather messages in subjects. Then, we can identify the volume of messages related to each subject. Then we analyze only messages that heave more than 4 answers; we believe that knowledge can be extracted based on interaction. Finally, we link the messages to be analyzed to project phases.

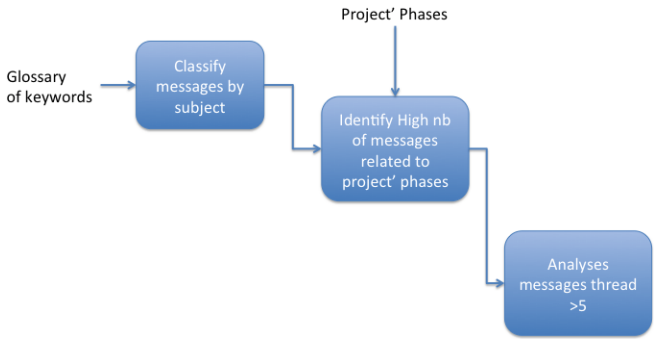


Fig. 3. First e-mails analysis

7.2 Messages Analysis

For each message thread (message and answers), we identify (Fig. 4):

- Information to be linked to organization:
- Authors, To whom, In Copy.
- Information about phases:
- Date and hour of messages and answers.
- Information about product:
- Topic and joined files.
- Information about message intention:
- Main speech act and intention of message.

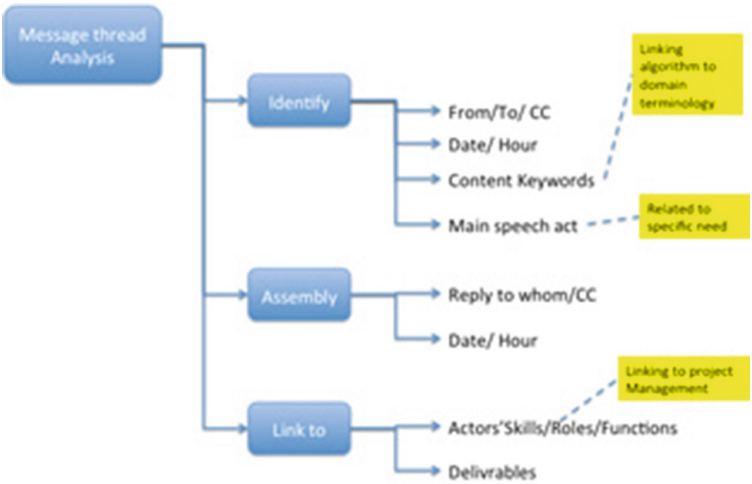


Fig. 4. Analysis of messages

By linking messages to project organization, we help in making sense of interactions between actors. In fact, the role and skill of messages' senders and receivers help to analyze the role of the message in problem solving and the nature of the content (solution answering a problem, proposition discussions, coordination messages, etc.). In the same way, linking messages to phases help to identify main problems to deal in each phase of the same type of projects.

As first work, we focus our speech act analysis on problem solving by identifying request and solution. So, we identify first speech acts that help to localize a request in a message (Fig. 4). Then, we study the organization of related messages thread in order to identify the solution proposed (if it exists) to the request. Our analysis is based first on pragmatics in order to characterize request speech act, and that by identifying request verbs and forms. In the present study we limited our research to the analysis of the act of requesting in problem solving sequences.

From a pragmatic point of view, a request is a directive speech act whose purpose is to get the hearer to do something in circumstances in which it is not obvious that he/she will perform the action in the normal course of events [27]. By introducing a request, the speaker believes that the hearer is able to perform an action. Request strategies are divided into two types according to the level of interpretation (on the part of the hearer) needed to understand the utterance as a request. The two types of requests include direct request and indirect requests. The request can be emphasized either projecting to: 1- the speaker (Can I do X?) or 2- the hearer (Can you do X?). A direct request may be use an imperative, a performativity, obligations and want or need statements.

An Indirect request may use query questions about ability, willingness, and capacity etc. of the hearer to do the action or use statements about the willingness (desire) of the speaker to see the hearer doing x. At last, for us, a grammatical utterance corresponds to only one speech act as in Table 1.

Table 1. Grid of request speech act

Request form	Linguistic form	Examples
Direct request	Imperative	Do x
	Performative	I am asking you to do x.
	Want or Need statements.	I need/want you to do x
	Obligation statements	You have to do x
	Query questions about ability of the hearer to do X	Can you do x?
		Could you do x?
	Query questions about Willingness of the Hearer to do X	Would you like to do x?
	Statements about the willingness (desire) of the speaker	I would like if you can do X
		I would appreciate if you can do X

Then, we complete our analysis by from one side identifying answers verbs and from another side, linking answers to actors' role and skills and also joining files. The date of answers can be an indicator of several elements in the organizations: engagement, difficulty of time spending of solution, stress and multi-responsibilities, etc. We aim at analyzing in the future the frequency of answers.

8 Example

8.1 Example Description

INFOPRO Business Publishing Company asked a software Company to develop a workflow tool that helps journalists to edit their articles and to follow the modification of the journal. Due to geographical constraints, nearly all the communications and negotiations during specification, implementation, tests and delivery were done through email. The development method was mixed agile, with weekly deliveries after initial analysis. The period of the project was more than one year. In this project, the actors were:

- SRA: an editing responsible (skill: law and management, Role: Contractor).
- JBJ: Information System Manager (Skill: Information system, Role: Contractor).
- FX: Information System Developer (Skill: Software Engineering, Role: Development manager).
- CV: Prototyping (Skill: Human Machine Interface, Role: User Interface Modeling).
- RT: Information System Developer (Skill: Software Engineering, Role: Sub-contractor).

Principles phases of the project can be found in Table 2:

Table 2. Phases of the project

	Q1 09	Q2 09	Q3 09	Q4 09	Q1 10	Q2 10	Q3 10
XML Import							
DocumentDatabasespecification and development							
Workflow Specification and development							
User Interface							
Export to magazine and website							
Web service specification and development							
Application test							

8.2 E-mails Analysis

As first step, we identify messages topics based on e-mails subjects. In our project, we identify main discussions topics based on keywords:

- XML: structuration, tag, tree, xsd, dtd, schema.
- BDD: Data base, table, editing part, code part.
- Interface workflow: UI, Workflow, User Interface, login, user management.
- Code: Insurance Code, Legifrance, auto code, vehicle code, mutu code, chapter, article.
- Document: new collection, construction, document.
- Export paper: Indesign, layout, mapping, tag indd, indd template.
- Export site: export web, web tag, web format, dtd web.
- Export Author: word, author, xslt author.
- Services: update legi, Legifrance update, FTP.
- Word: macro word, addin, web service, word 2007, wordlink.

Business emails collected from a project in their raw form are very redundant. In case of multiple replies or forward, several parts of the messages are repeated (e.g. quoted reply content). This occurs typically in long threads, mediated equivalents to spoken conversations, which are especially interesting for our study. Some preprocessing steps have to be performed in order to prepare messages and threads for analysis. We chose a deliberately simple method similar to Carvalho [6]. The steps involved are:

- Remove all previous message text from reply;
- Keep previous message in case of first reply of a thread or forwarded email cause it carries context information;
- Remove signatures and disclaimers when possible (identity of sender and receivers are kept in email metadata).

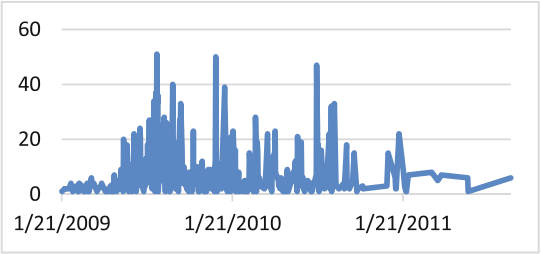
This leaves us with a corpus of messages and threads without too much duplicated or useless information. For some treatments, the granularity at message level is not sufficient, and it's relevant to split the messages into sentences. Here again, we use a standard approach and split according to punctuation and paragraph signs.

8.3 Frequency

Our corpus represent 3080 messages/14987 sentences in 801 threads between 30 projects actors. Sizes of message are relatively uniform, very long message are not frequent, being not suitable for email efficiency. On the average, threads length is between 2 and 7 messages with some exception at 17 or 21 messages. Usually threads are spread over 3 or 4 days, with higher messages frequency in the beginning.

We identify 10 main actors during this project that account for more than 80 % of the messages. Also the daily frequency in Table 3 show 3 relevant spikes of activity matching critical time of the project: the first delivery and second delivery and a new features addition. We will reduce our investigation to the first spike between 06/2009

Table 3. Daily message frequency.



and 09/2009 where a lot of exchanges occurs and focus on long threads showing the presence of a dialog.

As an additional information, it is to be noted that the Development Manager was the one receiving in TO (direct receiver) the higher number of messages, the Chief Editing Manager was the one sending message the most and the Information System Manager was the one receiving the most message in CC. This global numbers are matching their roles in the project, respectively executing, requesting and supervising.

8.4 Topics

We decide to make a very straightforward and knowledge oriented classification of messages and sentences. This steps is necessary in order to assert to deal with messages concerning pure software functionality knowledge and to filter project coordination emails.

Our approach is to create a keywords dictionary for the main topics of the project. This dictionary can be built from the following sources:

- Project phasing and specifications documents;
- an expert;
- domain ontology if available.

As in project memory context, we choose not to rely on statistical NLP clustering like in Cselle's approach [10] but to use existing context knowledge. This dictionary is voluntarily kept simple and have the form:

$$\text{Topic1: keywords1, keywords2...keywordsn.}$$

Using this dictionary we classify messages into weighted topics vector (same technic is applied to sentences for a fine granularity analysis). In order to do that we use a cosine similarity based algorithm. We compute a Lucene [14] ranking between our message and each topics in order to identify main topics of messages (boosting email subject importance compared to email body (Figs. 5 and 6). This give us a topics matrix T where T_{ij} represents weight of topic j in message i.

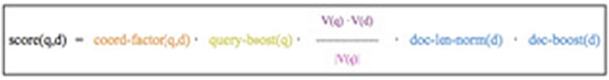


Fig. 5. Lucene scoring formula

As a side remark, keywords chosen in topics shall not overlap too much to keep the results significant. In Fig. 6, one can notice the amount of emails increased as the project is approaching its first milestone, but the topics are not always directly correlated to the phase of the project. In fact the project team members are often exchanging emails and dealing with problem before the phase really start or after (when the phase is supposed to be finished and some problems remained unsolved).

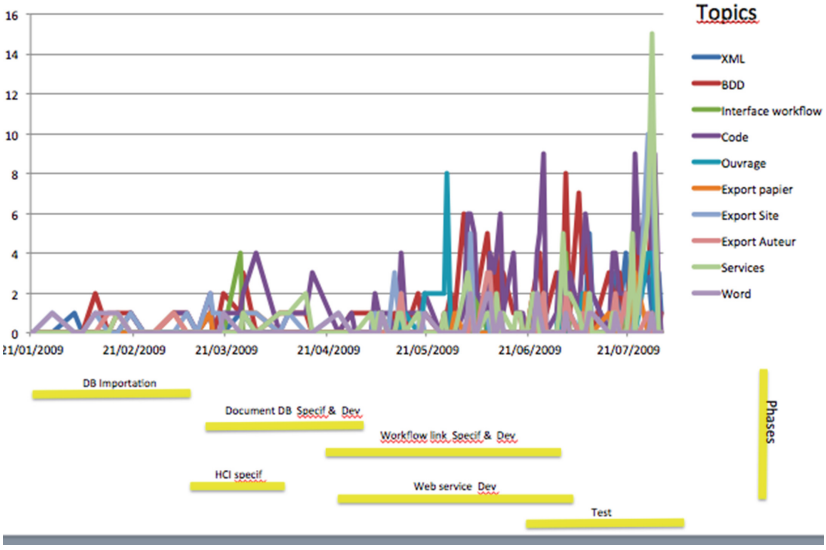


Fig. 6. Topics analysis/project phases

Figure 7 shows first step of analysis of these messages; in which we show senders and receivers and their skills, topics of messages and date of messages. Some patterns of communication are emerging, for instance, the Information System Manager (JBJ) is very often in CC of every message because of its supervisor role.

To analyze messages text, we use pragmatics in order to identify problem and solution discussions. For that, we identify Request messages based on Request speech acts. Then, we identify related answers messages. In these messages we look for sender's skills and joined files. So, we identify for the "Annexes" topic, in which there are 23 messages, related topics are XML and code. Messages were during 12 days from 5th until 17th June. They concern workflow development phase. Based on the Request-Answer grid and role actors, we analyze messages, in order to identify

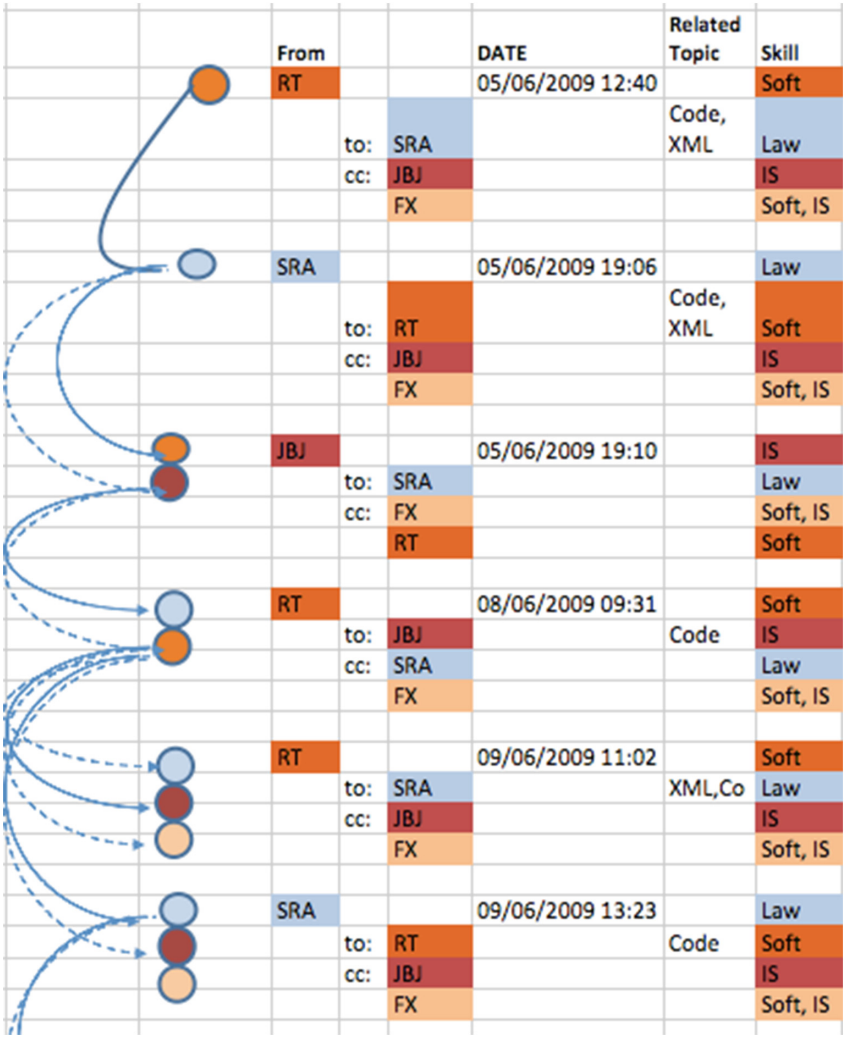


Fig. 7. First analysis of messages: representing of Senders/Recievers/Copy, date and actors role and skill

problem-solving intentions. So, we identify for instance, the problem Insurance Text extraction. SRA; the editing responsible (contractor) asks FX to extract Insurance text in a good format. When FX; the Information System Developer (Development manager) answer him, we suppose that as an answer, based on the role of sender of message and the main topic. We consider also joined files as part of this answer. Figure 8 shows this example.

From			Date	Sentence elements	Related Topic	Function
SRA			2009-06-05 12:40:46.	I put in "Bold", what **I need**:		Request
	to:	FX		1- *Insurances*		
	cc	JBJ		2- Text without tags Text in XML files	Code	
		CV		3- Tag Pb : Text outside tag in XML	XML, Code	
		RT		4- Tag Pb <b> is opened and not closed, as same as, tag is badly formed		
FX			2009-06-05 19:06:34.			Answer
	to:	SRA		1- *Insurances*		
	cc:	JBJ		I propose to convert: Xpress format in XML	XML	
		CV		Beware, the text will contain a lot of error blank, "enter" and image	Code	
		RT		I can transform it on enriched XML	XML	
				It contains a lot of references, so we have to compose with links		

Fig. 8. Example of messages analysis

9 Conclusions

The aim of our study is to identify knowledge from daily work. In this paper, we show that it is possible to study professional e-mails for this aim. We consider e-mails as specific discourse. So we use pragmatics generally used to analyze discourse and to categorize it to identify knowledge from professional e-mails. Our hypothesis is can we identify a grid as guide to analyze professional e-mails? If so, can the result be relevant as project knowledge?

Based on this hypothesis, we know that pragmatics intention must be based to context. So, we consider the project context from different aspect: organization and environment. We believe that this context is very helpful to clarify ambiguity of

sentence analysis. We show in the example how sender/receiver role can identify problem-solving answer. Adding this analysis to the identification of keywords of messages, as topics can be a first step, towards a structuring of knowledge: Problem related to a topic, possible answers.

We will continue to validate this work on other type of projects. This work can open to identify other grid analysis like: engagement of actors, design-rationale, coordination [23], etc. Our current work objective is to explore various techniques from machine learning to implement support algorithm for the projection of our features vectors (topics, pragmatics and context) to problem- solving knowledge model. Although related to the works of Cleland-Huang et al. [5] on Requirement traceability in software design, we will focused more on functional testing and design detection.

Finally, this study is a part of our work on project memory: Keeping track and structuring knowledge in daily work realization of project. We developed techniques to extract knowledge from project meetings [13] and to identify occurrences in order to identify concepts in project memory.

References

1. Atifi, H., Gauducheau, N., Marcoccia, M.: The effectiveness of professional emails: representations and communicative practices. In: Proceedings of the 13th Conference of the International Association for Dialogue Analysis, Dialogue and Representation, Montréal (2011)
2. Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Thomas, D.: Manifesto for agile software development (2001)
3. Bekhti, S., Matta, N.: Project memory: an approach of modelling and reusing the context and the de design rationale. In: Proceedings of the International Joint of Conferences of Artificial Intelligence (IJCAI 2003), Workshop on Knowledge Management and Organisational Memory, Acapulco (2003)
4. Blum-Kulka, S., House, J., Kasper, G. (eds.): Cross-Cultural Pragmatics: Requests and Apologies. Ablex Publishing, Norwood (1989)
5. Cleland-Huang, J., Settimi, R., Zou, X., Solc, P.: The detection and classification of non-functional requirements with application to early aspects. In: 14th IEEE International Conference on Requirements Engineering, pp 39–48 (2006)
6. Carvalho, V., Cohen, W.: Learning to extract signature and reply lines from email. In: Proceedings of the Conference on Email and Anti-spam, Palo Alto, CA (2004)
7. Carvalho, V., Cohen, W.: On the collective classification of email "speech acts". In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 345–352. Association for Computing Machinery, New York (2005)
8. Carvalho, V., Cohen, W.: Improving "email speech acts" analysis via n-gram selection. In: Proceedings of the HLT-NAACL 2006 Workshop on Analyzing Conversations in Text and Speech (ACTS 2009), pp. 35–41. Association for Computational Linguistics, Stroudsburg (2006)
9. Core, M.G., Allen, J.: Coding dialogs with the DAMSL annotation scheme. In: AAAI Fall Symposium on Communicative Action in Humans and Machines (1997)

10. Cselle, G., Albrecht, K., Wattenhofer, R.: BuzzTrack: topic detection and tracking in email. In: Proceedings of the 12th International Conference on Intelligent User Interfaces, pp. 190–197 (2007)
11. De Felice, R., Darby, J., Fisher, A., Peplow, D.: A classification scheme for annotating speech acts in a business email corpus. ICAME J. **37**, 71–105 (2013)
12. Dieng, R., Corby, O., Giboin, A., Ribière, M.: Methods and tools for corporate knowledge management. In: Proceedings of the KAW 1998, Banff, Canada (1998)
13. Ducellier, G., Matta, N., Charlot, Y., Tribouillois, F.: Traceability and structuring of cooperative Knowledge in design using PLM. Knowl. Manage. Collab. Spec. Issue Int. J. Knowl. Manage. Res. Pract. **11**(1), 53–61 (2013)
14. Ermine, J.L.: La gestion des connaissances. Hermès Sciences Publications, Paris (2002)
15. Gospodnetic, O., Hatcher, E.: Lucene in Action. Manning Publications, Greenwich (2004)
16. Gray, P.H.: A problem-solving perspective on knowledge management practice. Decis. Support Syst. **31**(1), 87–102 (2001)
17. Hardin, L.E.: Problem solving concepts and theories. J. Vet. Med. Educ. **30**(3), 227–230 (2002)
18. Kalia, K.A.: Identifying Business Tasks and Commitments from Email and Chat Conversations. Technical report, HP Labs (2013)
19. Ladas, C.: Scrumban-essays on kanban systems for lean software development (2009). http://Lulu.Com
20. Lampert, A., Dale, R., Paris, C.: Detecting emails containing requests for action. In: Proceedings of the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL), pp. 984–992. Association for Computational Linguistics (2010)
21. Lampert, A., Dale, R., Paris, C.: Classifying speech acts using Verbal Response Modes. In: Proceedings of the 2006 Australasian Language Technology Workshop (ALTW 2006), pp. 34–31 (2006)
22. Malvache, P., Prieur, P.: Mastering corporate experience with the REX method. In: Proceedings of the International Symposium on Management of Industrial and Corporate Knowledge (ISMICK 1993), Compiegne (1993)
23. Matta, N., Atifi, H., Sediri, M. Sagdal, M.: Analysis of interactions on coordination for design projects. In: IEEE Proceedings of the 5th International Conference on Signal-Image Technology and Internet Based Systems, Kula Lumpur (2010)
24. Matta, N., Ribière, M., Corby, O., Lewkowicz, M., Zacklad, M.: Project Memory in Design, Industrial Knowledge Management - A Micro Level Approach. Springer, Verlag (2000)
25. Matta, N., Ermine, J.-L., Aubertin, G., Trivin, J.-Y.: Knowledge capitalization with a knowledge engineering approach: the MASK method. In: Dieng-Kuntz, R., Matta, N. (eds.) Knowledge Management and Organizational Memories, pp. 17–28. Kluwer Academic Publishers, New York (2002)
26. Newell, A., Simon, H.A.: Human Problem Solving. Prentice-Hall, Inc., Englewood Cliffs (1972)
27. Searle, J.R.: Speech Acts: An Essay in the Philosophy of Language. Cambridge University Press, Cambridge (1969)
28. Tourtier, P.A.: Analyse préliminaire des métiers et de leurs interactions. In: Rapport intermédiaire, project GENIE, INRIA-Dassault-Aviation (1995)
29. Yelati, S., Sangal, R.: Novel approach for tagging of discourse segments in help-desk e-mails. In: 2011 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), vol. 3, pp. 369–372 (2011)