Towards Supporting the Analysis of Online Discussions in OSS Communities: A Speech-Act Based Approach

Itzel Morales-Ramirez[1,2](✉), Anna Perini[1], and Mariano Ceccato[1]

[1] Software Engineering Research Unit, Fondazione Bruno Kessler - IRST,
Via Sommarive 18, 38123 Trento, Italy
{imramirez,perini,ceccato}@fbk.eu
[2] University of Trento, Trento, Italy

Abstract. Open-Source Software (OSS) community members report bugs, request features or clarifications by writing messages (in unstructured natural language) to mailing lists. Analysts examine them dealing with an effort demanding and error prone task, which requires reading huge threads of emails. Automated support for retrieving relevant information and particularly for recognizing discussants' intentions (e.g., suggesting, complaining) can support analysts, and allow them to increase the performance of this task. Online discussions are almost synchronous written conversations that can be analyzed applying computational linguistic techniques that build on the speech act theory. Our approach builds on this observation. We propose to analyze OSS mailing-list discussions in terms of the linguistic and non-linguistic acts expressed by the participants, and provide a tool-supported *speech-act* analysis method. In this paper we describe this method and discuss how to empirically evaluate it. We discuss the results of the first execution of an empirical study that involved 20 subjects.

Keywords: Online discussions · Intentions extraction · Speech act theory

1 Introduction

The increasing participation of stakeholders to online discussions of Open-Source Software (OSS) is turning these discussions into an attractive source of information, although still costly to exploit. OSS is usually produced by distributed collaborative communities composed of heterogeneous and diverse stakeholders (including users, developers, and analysts [1]). Such stakeholders extensively rely on online communication channels such as open forums or mailing lists, to elaborate solution design, code writing, software deployment, maintenance and evolution. Mailing-list discussions are highly exploited by all kind of stakeholders to provide bug reports, feature requests or simply to ask for clarifications. Wherein discussants express their arguments mainly as unstructured natural

S. Nurcan and E. Pimenidis (Eds.): CAiSE Forum 2014, LNBIP 204, pp. 215–232, 2015.
DOI: 10.1007/978-3-319-19270-3_14

language (NL) text. The immediacy that email offers to its users makes it the preferred channel of communication, as reported in [2]. But this can result in huge threads of emails that the analysts need to carefully check in order to identify information that could be important for software development tasks. Similarly, open forum is a communication channel typically chosen by users of software applications to discuss about tips, bugs or features related to such an application, still using unstructured NL text.

Analysts who aim at recognizing feature requests or bug identification by reading the resulting huge threads of messages or emails, face an effort demanding and error prone task. This motivates research on techniques for automating the extraction of relevant information from online discussions. Our research has the ultimate goal of lighten the burden of analyzing online discussions related to software development by supporting developers or requirements analysts to identify discussants' intentions (such as suggesting or complaining). We take the inspiration from the Speech Act Theory (SAT), originally formulated by Austin and Searle [3,4], whose core idea is captured in the following quotation: "by saying something, we react by doing something". Indeed, a speaker may aim at persuading, inspiring or getting a hearer to do something. Concretely, speech acts are classified according to specific performative verbs, such as *suggest*, *recommend*, and *advise*, among other verbs. Since online discussions are considered almost synchronous written conversations, we propose to analyze them in terms of *speech-acts* including linguistic acts, which corresponds to the *speech-act* types as per the SAT, and non-linguistic acts that are commonly used in such type of discussions, e.g. log files or URL links. We define a tool-supported method at use of analysts during the processing of online discussions. This tool aims at facilitating the recognition of *speech-acts* frequently used in sentences that describe bug identifications, features or clarifications requests, by supporting *speech-acts* annotation of online discussions. The tool builds on a natural language processing (NLP) framework [5] and the SAT along with its application in computational linguistic [6].

In this paper, we describe our method and the first phase of an empirical evaluation plan, which aims at providing evidences about the method's effectiveness. Indeed, to empirically evaluate the approach we designed a three-phase plan: the first phase is devoted to the investigation of how non-trained humans perform the activity of annotating sentences; the results are used in the second phase as ground truth, as well as input for improving the classification rules exploited by the tool; and the third phase is aimed at evaluating if the tool-supported identification of bug and feature requests will be effective in a realistic setting with expert analysts. As said before, here we focus on the first phase whose research question is: *RQ: How difficult is for non-trained human annotators to recognize speech-acts in online discussions?* We present the design of an empirical study and report the results of the analysis of the first execution with 20 subjects, who were requested to annotate 20 OSS online discussions containing 1685 sentences in total. The results allow us to estimate the effort required to manually annotate sentences. Moreover, the study provided interesting suggestions on how to improve the study design towards building a ground truth to be used for evaluating the performance of our proposed tool-supported method.

The remainder of the paper is structured as follows. In Sect. 2 we give some background on SAT and on the NLP framework used in our approach. In Sect. 3 our tool-supported approach for analyzing online discussions is presented. The design of an empirical study for the first phase of the evaluation plan is presented, together with a description of its execution in Sect. 4. The discussion of the results is given in Sect. 5. The related work is presented in Sect. 6 and the conclusion in Sect. 7.

2 Background: Speech Act Theory and the NLP Framework

The Speech Act Theory (SAT) was developed by Austin and Searle in the field of philosophy of language [3,4]. In a nutshell, the theory claims that when a person says something she/he attempts to communicate certain things to the addressees, which affect either their believes and/or their behavior. So, for instance, if I say "I'll bring you a chocolate", this utterance expresses my intention (technically named *illocutionary act*) to make you aware that I'm committing to bring you a chocolate, and the effect (technically named *perlocutionary act*), is that you get convinced about my intention and expect to receive a chocolate. According to the classification proposed by Bach and Harnish in [7], this type of *speech-act* is called *Commissives* since it expresses the speaker's intention to commit to do something for the benefit of the hearer. Other types are: the *Constantives* type, which expresses the speaker's belief and her intention or desire that the hearer forms a like belief; the *Directives* type, which expresses the speaker's attitude toward some prospective action that should be performed by the hearer; and the *Acknowledgements* type that expresses the speaker's intention to satisfy a social expectation.

The NLP framework used is GATE (General Architecture for Text Engineering) which is a Java suite of tools [5], developed by the University of Sheffield in UK, for building and deploying software components to process human language. GATE can support a wide range of NLP tasks for Information Extraction (IE). IE refers to the extraction of relevant information from unstructured text, such as entities and relationships between them, thus providing facts to feed a knowledge base [8]. GATE is widely used both in research and application work in different fields (e.g. cancer research, web mining, law). This tool is composed of three main components for performing language processing tasks, namely the *Language Resources* component that represents entities such as lexicons, corpora or ontologies; the *Processing Resources* component, which contains a library of executable procedure, such as parsers, generators or ngram modelers; and the *Visual Resources* component that provides visualization and editing functions that are used in GUIs.

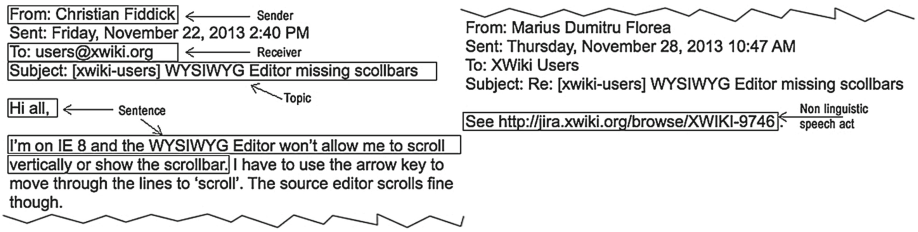
3 Approach to Analyse Online Discussions

In order to analyze online discussions through mailing lists, as those used in OSS development, we apply the concepts related to SAT and a communication ontology that we have described in a previous work [9]. Specifically, we first identify which linguistic and non-linguistic acts can be used to model such online discussions, we define a suitable *speech-act* taxonomy and, based on it, a proposal for analyzing online discussions.

The concepts are: a *sender*, a *receiver* and an expression (typically a *sentence* or *proposition*). The expression is written in a given language and within certain context that is determined by a *topic*. For example, by analyzing the directive *speech-act* - "Open the door, please!"- results in: the *sender* is me; the *receiver* is you; the *sentence* or *proposition* is "Open the door, please!"; in English; and the context could be a situation in which we are exiting the office and I'm carrying a heavy box.

Mailing-list discussions are organized as emails threads. A thread is initiated by a member of the mailing list, who proposes a topic to be discussed (i.e., the field *Subject:* of such an email). The discussion develops as a thread of replies by interested people who give their contribution, writing NL text. Different behaviors of the participants emerge in a conversation: someone asks about a topic, or states problems related with it; others provide suggestions, answer questions or simply add details.

To analyze these conversations we first apply the concepts mentioned previously, as shown in the excerpt depicted in Fig. 1: the *sender* corresponds to the *writer* who is specified in the field *From*; the *receiver* is the *addressee* who is the person in the field *To* (it can be also addressees); each *proposition* is a *sentence* in the email body; the language is English, with terms that may be typically used in the context; and the context is determined by the *topic* in the field *Subject*.



From: Christian Fiddick
Sent: Friday, November 22, 2013 2:40 PM
To: users@xwiki.org
Subject: [xwiki-users] WYSIWYG Editor missing scollbars

Hi all,

I'm on IE 8 and the WYSIWYG Editor won't allow me to scroll vertically or show the scrollbar. I have to use the arrow key to move through the lines to 'scroll'. The source editor scrolls fine though.

From: Marius Dumitru Florea
Sent: Thursday, November 28, 2013 10:47 AM
To: XWiki Users
Subject: Re: [xwiki-users] WYSIWYG Editor missing scollbars

See http://jira.xwiki.org/browse/XWIKI-9746.

Fig. 1. Excerpt of an online discussion in OSS.

In terms of SAT concepts we can characterize emails of discussion threads as a set of linguistic acts (*speech-acts*) and non-linguistic acts, such as email attachments, URL links, fragments of code, etc. For the sake of simplicity we will use the term *speech-act* to name both types of acts. We find that *speech-acts*, hence, are composed of verbal, syntactic and semantic aspects that reflect the *intention* of a writer. In this paper we consider that an intention is found in a sentence and it is reified by a sequence of specific words, e.g. the sentence "Please help me ..." makes an addressee recognize the speaker's *intention* of requesting something. In the right part of Fig. 1 the sentence in the rectangle shows a non-linguistic *speech-act* with the *intention* to make the addressee give a look at an URL link.

3.1 Taxonomy of Speech Acts

We have elaborated a taxonomy[1] of categories and subcategories of *speech-acts*, shown in Table 1. Column *Category* refers to the main types of *speech-acts* found in the literature, the column *Subcategory* refers to the specific types of *speech-acts*. The column *Analysis category* is the aggregation of *speech-acts* in a reduced number of categories. Finally, the column *Some definitions* presents some *speech-acts* definitions and examples. For instance, the category *Constantives* is specialized into seven subcategories, from which *Assertives*, *Confirmatives* and *Concessives* are aggregated into the analysis category *Assertives*. The *speech-act Assertives* is considered as a strong belief and intention by a sender who maintains his/her belief about something.

Table 1. Categories of *speech-acts*.

Category	Subcategory	Analysis category	Definition (excerpt)(see Footnote 1)
Constantives	Informatives	Not used	*Assertives: speech-act* that is considered as having a strong belief and intention by a sender who maintains his/her belief about something, e.g., “I know the chocolate is good for your health...”. *Suppositive: speech-act* conveying that is worth considering the consequences of something regardless of whether it is true, e.g. “I suppose the configuration file ...” *Requestive: speech-act* expressing sender's intention that the receiver take the expressed desire as reason to act, e.g., “I kindly ask you to provide me ...”
	Assertives	Assertives	
	Confirmatives		
	Concessives		
	Suggestives	Responsives	
	Suppositives		
	Responsives		
Directives	Requestives	Requestives	
	Questions		
	Requirements		
Expressives	Thank	Not used	
	Accept	Accept	
	Reject	Reject	
	Negative opinion	Negative opinion	
	Positive opinion	Positive opinion	
Attach (non-linguistic)	URL link	Attach	
	Code line		
	Log file		

3.2 Automated Tagging of Speech Acts

The procedure we followed to build our method includes a gathering of seed words and the computation of their frequencies. This was done in order to have evidence of a presence of performative verbs in messages of OSS community

[1] Taxonomy and definitions have been adapted from the work of Bach and Harnish [7].

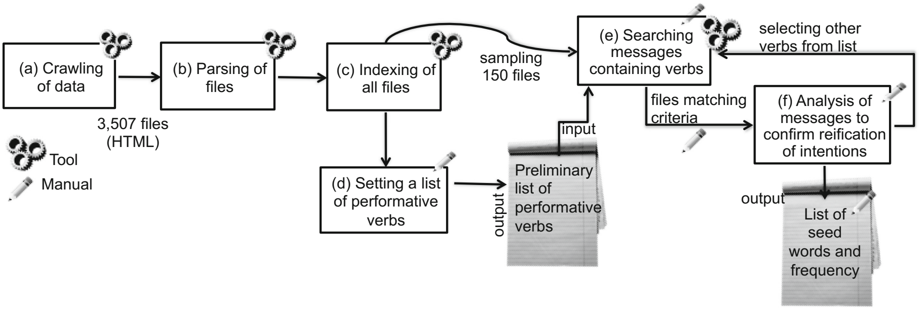

(a) Crawling of data
3,507 files (HTML)
(b) Parsing of files
(c) Indexing of all files
sampling 150 files
(e) Searching messages containing verbs
selecting other verbs from list
files matching criteria
(f) Analysis of messages to confirm reification of intentions
input
(d) Setting a list of performative verbs
output
Preliminary list of performative verbs
output
List of seed words and frequency
Tool
Manual

Fig. 2. Process to select the seed words.

discussions. The resulting seed words compose sets of words that are used later to elaborate rules for tagging *speech-acts*. In the following we explain the steps of the procedure.

Selection of Seed Words for the OSS Context. To find out what are the words reifying the intentions that we want to identify, we examined crawled messages from Apache OpenOffice bugzilla[2] to find regularities in written messages. We analyzed empirical data from other OSS project to not be biased in defining the kind of words regularly used by participants in online discussions. Following a supervised selection of seed words, we executed the process depicted in Fig. 2, from step (a) to (f):

(a) Crawling of data: we used a tool for crawling data, called Teleport Ultra[3]. We first executed a simple search on Apache OpenOffice bugzilla platform using a single word, i.e. *feature*, in the searching box. Then we took the web link appearing and used it for the crawling tool.
(b) Parsing of data: we applied an algorithm to parse them into files with the extension *properties*, by using Jsoup[4]. We obtained 3,507 files.
(c) Indexing files: we indexed all the files, using Lucene[5], to have a corpus of messages.
(d) Setting a reference list or verbs: we then elaborated a preliminary reference list of performative verbs taken from the suggested list described by Bach and Harnish [7].
(e) Searching messages containing verbs: for this step we sampled 150 files that we selected randomly and we took the reference list to search for the performative verbs contained in the messages, using Lucene. The output is a set of messages that were analyzed.
(f) Analyzing messages: we read each one of the messages from the resulting set of files matching the searched verbs (output from step (e)). By reading and

[2] https://issues.apache.org/ooo/.
[3] http://www.tenmax.com/Teleport/ultra/home.htm.
[4] http://www.jsoup.org.
[5] http://lucene.apache.org/core/.

interpreting the whole message wrt. the intentions described in our taxonomy, see Table 1, we classified the sentence containing the verb in the corresponding subcategory. A given set of seed words accompanying the verbs that appear in the message become candidate indicators for the identified subcategory if we found at least three occurrences of the same set of seed words in the same subcategory. We returned to step (e) till we finish the reference list of verbs.

For example, the final list of seed words and their occurrence contains the intention *Requirements*, whith the following three instances of seed words: "I want" with a frequency of 11 messages, "I would like to have" found in 7 messages, and "It would be nice" in 65 messages.

Design of the Speech-Acts Tagging Tool. Our tool is based on a knowledge-heavy approach [10], this means the use of a part-of-speech (POS) tagger, java annotation patterns engine (JAPE) rules, a tokenizer, a lemmatizer and gazetteers (list of performative verbs). Gazetteers and JAPE rules have been tailored to annotate the intentions applying some tags. These tags are used to annotate fragments of text, the tags are the subcategories of *speech-acts* defined in Table 1. To adapt the JAPE module we have formulated lexico-syntactic rules, by using a set of words inspired from the examples given by Jurafsky [11] and by the obtained seed words from the previous step. The Gazetteers used in our approach are the lists of verbs and seed words for each subcategory of *speech-act*. Some JAPE rules use the Gazetteers to annotate intentions. The linguistic analysis is executed on discussion threads in the format of TXT files, which are the input. The tool is used to annotate sentences on the text messages of each thread. After this, the files annotated with intentions are parsed to extract the intentions found in each message. Finally, an analysis of intentions is performed, following an analysis model that we are elaborating.

Examples of design of JAPE rules are illustrated below in Table 2, the full set of rules are available online[6]. The first column, *Category*, refers to the category of *speech-act*. The second column, *Tag*, refers to the name for tagging sentences. The third column, *Rule*, shows the rule for tagging the *speech-act*, for example, the category *Constantives* has two tags, namely, *Suggestives* and *Responsives*. Along these lines, the tag *Suggestives* presents two rules to annotate. As it can be seen there are POS tags and seed words that are used by the tool to annotate the *speech-acts*. The POS tags $< PRP >$ and $< MD >$ refers to the initial set of words to annotate, the $< Keyword >$ refers to the list of verbs or seed words defined in the Gazetteer[7] modules and that are used by the JAPE rules.

We manually designed the rules considering some characteristics for extracting the intentions, such as preceding and succeeding words, length of the words, root of the words, special types of verbs, using the seed words, syntax and the

[6] JAPE files are available at http://selab.fbk.eu/imramirez/JAPErulesSep2014/files.zip.

[7] Gazetteer files are available at http://selab.fbk.eu/imramirez/GazetteerSep2014/files.zip.

Table 2. Example of rules for tagging *speech-acts Directives* and *Constantives.*

Category	Tag	Rule
Directives	Questions	$< WRB > + < PRP > + < content > +$ "?" $< MD > + < PRP > + < content > +$ "?"
Constantives	Suggestives	$< PRP > + (< MD >)* +$ ("try" | "check") $< PRP > + (< MD >)* +$ ("suggest" | "recommend")
	Responsives	$(< PRP >)* +$ "[Hh]ope" $+ < content > +$ "help"

codification of the POS tagger used by GATE. The tag is used to label a text fragment when one of the corresponding rule matches it. Each rule is formulated as a regular expression. The regular expressions $< content >$, $(< MD >)*$ and [Hh], for example, make reference to a set of words in the middle of two keywords or POS tags, to the presence or absence of the POS tag and to the uppercase or lowercase of the first letter of a word, respectively. More details can be found our previous work [12].

Analysis Model. We are building an analysis model of the intentions in a discussion thread that can be performed at different levels of granularity, see Table 3. At the sentence level we can identify single and nested *speech-acts*.

Table 3. Granularity of analysis.

Granularity level	Aggregation of intentions	Example
Sentence	Single intention	$\overbrace{"I\ suggest\ you"}^{Suggestives}$
	Nested intentions	$\overbrace{"Why\ \underbrace{don't\ you\ try}_{Suggestives}?"}^{Questions}$
Message	Compound intentions	$\text{Bug indicator} = \begin{cases} \underbrace{"There\ is\ a\ problem"}_{Negative\ opinion} \\ \underbrace{"Can\ anyone\ help\ me?"}_{Questions} \end{cases}$ $\text{Feature indicator} = \begin{cases} \underbrace{"I\ really\ like\ the\ application"}_{Positive\ opinion} \\ \underbrace{"It\ would\ be\ nice"}_{Requirement} \end{cases}$

For instance, in the sentence "I suggest you to make a copy of your data", the single intention of suggesting is triggered by the sequences of words "I suggest you...", which refers to the *speech-act Suggestives*. An example of nested intentions is expressed in the sentence "Why don't you try to use the wizard?". In this case there are two *speech-acts*, one is "Why... ?", and the other one is "don't you try", representative of the intentions of questioning and suggesting, respectively. At the message level the occurrences of pairs of intentions is analyzed, called compound intentions, and we claim can be indicators of *Bug*, *Feature*, or *Clarification* requests. For example, a combination of *speech-act Negative opinion* ("There is a problem") and *Question* ("Can anyone help me?") can be an indicator of a bug. Therefore, a set of nested or compound linguistic and non-linguistic acts can be considered as indicators of bugs, features, etc.

4 Empirical Evaluation Phase 1: Human Annotation of *Speech-Acts*

4.1 Overall Plan

Our plan for empirically evaluating the proposed approach consists of three phases. In the first phase we investigate how non-trained humans perform when annotating *speech-acts*, for which we have formulated the following research question: *RQ1. How difficult is for non-trained human annotators to recognize speech-acts in online discussions?* For the second phase we would use part of the annotated database for improving the rules of our tool and to evaluate the performance of it. Then, we want to investigate *RQ2. What is the accuracy of the tool for annotating speech-acts in terms of precision and recall?* The third phase is aimed at evaluating if the tool-supported identification of bug and feature requests will be effective in a realistic setting, possibly with the participation of expert analysts. In the following subsections we describe the details of the first execution of phase one.

4.2 Context

The context of the study is the following: the *subjects* are people playing the role of a receiver of messages that must interpret the predominant intention in each sender's sentence. The *objects* are 20 discussions from an OSS project, namely the XWiki project whose data is publicly available[8]. The 20 discussions are split into 1685 sentences in total. We sent email invitations to 38 people to participate in the empirical study. The people invited have a position either as a PhD student, Post-doc or technician. Their field of expertise is in Computer Science, Software Engineering or Biology. All of them are from different countries (e.g., China, The Netherlands, Mexico, Brazil, Germany, among others). We informed them that the activity should have been performed through an online platform and that it should have required approximatively 1 hour and 30 min to be completed. We did not specify time constraints, although we expressed our expectation to collect data after a week. Only 20 subjects accepted the invitation, we grouped them in pairs (labeled as $G_1 \ldots G_{10}$) but they worked individually. The members of a group were selected randomly. Each group was assigned with two online discussions to annotate.

[8] XWiki is an OSS generic platform for developing collaborative applications, http://www.xwiki.org/xwiki/bin/view/Main/WebHome.

4.3 Metrics

Given the research question *RQ1. How difficult is for non-trained human annotators to recognize speech-acts in online discussions?* We collect these (dependent) metrics:

- $Time_i$ = seconds spend by subject to annotate the sentence i;
- $Effort_p = sum(Time_i)/\#\ of\ sentences$ [for the participant p];
- $AgreementN$ = Kappa(i,j) between subject i and j (on the same discussion) with N number of classes.

The first two metrics measure respectively the *Time* required to annotate a single sentence, and the *Effort* to annotate an entire session, as the average time per sentence in the session. The last metric represents the *Agreement* between a pair of subjects in annotating the same sentences, computed with the statistical measure Cohen's Kappa (see Analysis, Sect. 4.5).

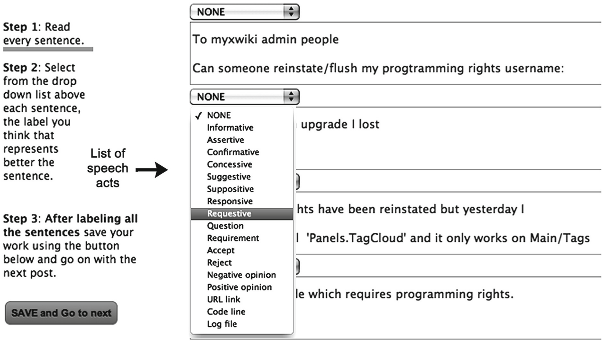
Moreover, based on a profiling questionnaire, we also measure independent factors that possibly influence our dependent metrics, such as:

1. *Working field:* we have classified the annotators according to three fields, namely, Biology, Computer Science and Software Engineering.
2. *Years of experience:* according to the answers we created 4 ranges of years of experience (i.e., 1–3, 4–5, 6–8, 9–15).
3. *Current position:* another possible factor could be if the participant is a PhD student, Post doc or technician.
4. *Distributed collaboration:* we have asked to the participants their experience in working collaboratively in a distributed setting.
5. *Channel of communication:* we also wanted to know which is their preferred channel of communication.
6. *Knowledge about OSS:* the participants' knowledge about OSS is also one possible factor.

4.4 Experiment Material and Procedure

In this first execution we exploited an online platform that allows us to set up annotation tasks on preprocessed sets of OSS online discussions, involving distributed annotators[9]. We gave the subjects instructions about how to perform the assigned task, by sending individual emails including: a password and a URL link to access the online platform; and a PDF document containing a short guide about the *speech-act* annotation. The guide briefly introduces the goal of the annotation task, it lists *speech-acts*, gives some hints of what is meant by the speaker's intention expressed through a *speech-act*, and illustrates some screenshots of the platform they would have been presented along the basic steps. A screenshot is depicted in Fig. 3. Each text box, from the top to the bottom, shows a sentence in a discussion. Above each text box there is a drop down

[9] Similar to crowdsourcing platforms, such as CrowdFlower http://www.crowdflower.com/.



Step 1: Read every sentence.

Step 2: Select from the drop down list above each sentence, the label you think that represents better the sentence.

Step 3: **After labeling all the sentences** save your work using the button below and go on with the next post.

To myxwiki admin people

Can someone reinstate/flush my progtramming rights username:

upgrade I lost

hts have been reinstated but yesterday I

I 'Panels.TagCloud' and it only works on Main/Tags

le which requires programming rights.

Fig. 3. Interface of the online platform to annotate sentences.

menu containing the list of the labels for the different *speech-act* subcategories. After reading each sentence the subjects should select the *speech-act* label that represents the intention of the sentence. The annotation session is saved by clicking the button *SAVE* before being given a new set of sentences (varying the # of sentences).

The list of *speech-acts* labels to be used for the annotation study was reduced from the 18 presented in Table 1, column *Subcategory*, to 17. Indeed, the *speech-act Thank* was ignored since it is trivially identifiable by the word "thank" and its variants. We added a label "NONE" to describe a nonsense sentence in case the participants were not satisfied with any other label. However, we did not consider it as well as the default *speech-act Informative* in the analysis of the data, reported below.

4.5 Analysis

For the analysis we have used descriptive statistics and ANOVA tests to make our interpretations of how difficult is for human annotators to identify expressed intentions in online discussions. Since each group was assigned with different number of sentences, for the purpose of our analysis, and taking into account space limits, we have selected the first 100 sentences that each participant annotated[10]. We have computed descriptive statistics such as the mean, median and standard deviation. We have applied the ANOVA test of time and ANOVA test of time by participant profile to analyze possible influencing factors. We present some plots for cases where the statistical significance is reached with a confidence of 95%, i.e. with a *p-value* < 0.05. We have used the R tool to compute the descriptive statistics and ANOVA tests.

We computed the Cohen's Kappa coefficient [13] to obtain the percentage of agreement of a pair of annotators. There is a perfect agreement when $k = 1$ and no agreement when $k = 0$. We interpret the quality of agreement, thereby quality

[10] The complete analysis is described in the technical report available at http://selab.fbk.eu/imramirez/TR_CAiSEDec2014.pdf.

of the data, according to two different scales, namely Landis and Koch [14] and Green [15].

Eventually, in order to understand how difficult is for non-trained human annotators to recognize *speech-acts* in online discussions, we present our interpretations based on the empirical evidence.

5 Results

In this section we describe the results of the measurements applied and the interpretation of the ANOVA tests.

Analysis of Time. We observed that the time for annotating has the following distribution: a mean of 35 s, a median of 18 and a standard deviation of 56 s.

The ANOVA test of time shows that the time for annotating a sentence is influenced by the order of the sentences, i.e. there is a learning effect. Another factor that influences the time is the participant profile, whose ANOVA test is shown in Table 4 (see Footnote 10).

Table 4. ANOVA of time by participants' profile.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Field	1	28461.79	28461.79	9.72	**0.0019**
Years	3	180122.91	60040.97	20.50	<**0.01**
Position	1	16577.79	16577.79	5.66	**0.0175**
Distributed	1	156474.19	156474.19	53.43	<**0.01**
Channel	1	11737.71	11737.71	4.01	**0.0455**
OSS	1	4165.79	4165.79	1.42	0.2332
Residuals	1454	4257992.54	2928.47		

As it can be observed the participant's field of expertise, years of experience, position (PhD vs. Post-doc), the past experience in working with a distributed team and the preferred communication channel are co-factors that influence the time to annotate a sentence. Therefore, the difficulty (in terms of time) that each participant experienced during the annotation of sentences varies according to these factors. Focusing on the years of experience, we see that for experienced participants it took 25 s per sentence, while for participants with less experience it took 10 s per sentence. Our interpretation is that with more years of experience, participants become more meticulous in analyzing something, paying more attention in performing the assigned tasks. Instead, less experienced participants take risks and act more by instinct or common sense. However, we cannot exclude that participants who spent more time in annotating just experienced a higher difficulty.

Analysis of Effort. While the analysis of time is focused on the answers of the questionnaire, the analysis of effort involves the complete annotation session. Regarding the distribution of effort spent by subject, we observed a mean of 36 s per sentence, a median of 33 and a standard deviation of 19 s. We computed the ANOVA of effort with factors such as the participant's group, the kappa agreement for 8-type *speech-act* categories and the agreement for 16-type *speech-act* categories. Also, we computed the ANOVA of effort by participant profile. We found in the ANOVA of effort that the agreement for 16-categories has a slight influence on the effort, and that the experience in working in a distributed setting can be a factor on the effort. Figure 4 shows that participants who exchange many emails per day have spent around 10 s per sentence, which can be interpreted as the annotation task is less difficult due to their high experience in working collaboratively, differently from other participants who exchange emails less frequently.

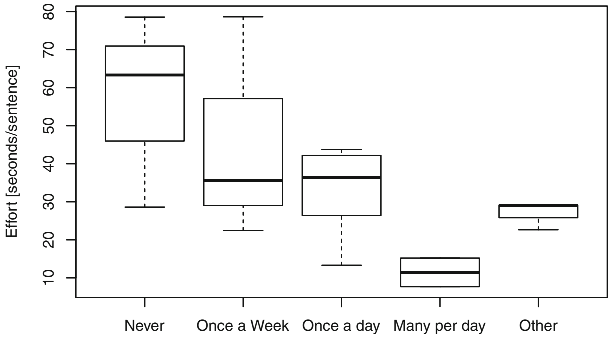


Fig. 4. Boxplot of effort by collaborative working frequency.

Analysis of Agreement. The third metric refers to the Cohen's Kappa value, we computed the k-values by considering the 16-categories (called *Subcategory* in Table 1) and the 8-categories, shown in Table 5. The first column reports the number of types of *speech-acts* . The following 10 columns present the k-value of each group ($G_{\#}$). The second row reports the k-value computed on the 16-categories and the third row on the 8-categories. These 8-categories correspond to the aggregated subcategories of *speech-acts*, ignoring Default and Thank; and including the single categories Accept, Reject, Negative and Positive opinion.

According to the Landis scale, values of k from 0.0 to 0.2 correspond to *Slight*, from 0.2 to 0.4 to *Fair*, from 0.4 to 0.6 to *Moderate*, from 0.6 to 0.8 to *Substantial* and from 0.8 to 1.0 to *Perfect*. While for Green's scale values of k ranging from 0.0 to 0.4 are considered *Low*, from 0.4 to 0.75 *Fair/Good*, and from 0.75 to 1.0 *High*. The best agreement is between the participants of group G_6 with 0.51 for 16-categories and 0.66 with 8-categories. The interpretation are *Moderate* and *Substantial* on the Landis scale and *Fair/Good* for both cases in the Green scale. The slight correlation between the agreement on 16-categories

Table 5. k-value per group for the *speech-acts* subcategory and analysis category (*#speech-act*). Gray-colored cells are the highest k-values.

#speech-act	$\mathbf{G}_1$	$\mathbf{G}_2$	$\mathbf{G}_3$	$\mathbf{G}_4$	$\mathbf{G}_5$	$\mathbf{G}_6$	$\mathbf{G}_7$	$\mathbf{G}_8$	$\mathbf{G}_9$	$\mathbf{G}_{10}$
16	0.38	0.31	0.15	0.33	0.22	0.51	0.41	0.28	0.29	0.28
8	0.49	0.44	0.29	0.34	0.23	0.66	0.48	0.43	0.38	0.39

and effort can be seen in the plot of their correlation in Fig. 5. This can be interpreted as the participants who put more effort in annotating sentences, indeed, take more time to analyze the sentences. Therefore, there is an increase in the agreement and probably the quality of the task also increases. Thus the annotation task becomes difficult because there could be a cognitive overload while understanding all the categories and select only one for a given sentence.

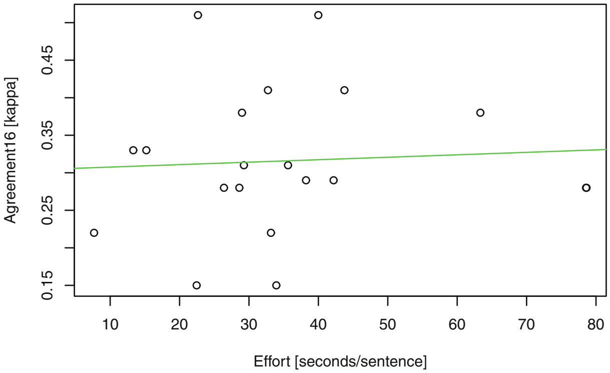


Fig. 5. Correlation of effort and kappa agreement for the 16-type *speech-act* categories.

5.1 Threats to Validity and Observations

We discuss the threats to validity as recommended by the literature in empirical software engineering [16]. *Conclusion validity* threats concern issues that affect the ability to draw the correct conclusion on the observed phenomenon. The evidence obtained so far does not show a trend, thus we need replications and more data to confirm the difficulty experienced by the human annotators. *Internal validity* threats concern the possible confounding elements that may hinder a well performed experiment. On one side, since the participants did not receive any training we consider that they were not biased while performing the activity. On the other side, variables as different background, not enough attention or low understanding of the task and that participants are non-native English speakers could have affected the results. *Construct validity* threats concern the relationship between theory and observation. As we requested to participants to annotate sentences using 16 *speech-acts* labels (based on the theory) and then for the analysis we aggregated the classes and reduced them to 8, we see that there might not be a clear disambiguation of all the classes and for further experiments we will reduce the number of labels. *External validity* threats concern extending the validity of observations outside the context. *Speech-acts* are not limited to a certain domain and are expressed by people mainly in spoken language, but the NLP research field exploits transcripts of telephone conversations to find *speech-acts*. We claim that *speech-acts* can be found in any type of written document, specially but not uniquely, in archives of online discussions.

Observations. The execution of the empirical evaluation brought to our attention important aspects to be considered in future empirical evaluations, which did not emerge in a pilot execution with one of the authors acting as annotator. We shortly discuss them in the following. We need to consider *speech-acts* as an aggregated category. Indeed, we have recognized that there were misunderstandings between *speech-acts* such as Question and Requestive. Some participants were labeling as a Requestive act a Question act that other participant was annotating. For example, the participants 19 and 20 with the sentences: *Is SQL Server 2005 a hard requirement?* and *Which errors do you meet?* Based on this, we acknowledge that the tool will likely fail since we rely on a lexico-syntactic approach for annotating. We need to train people in this matter for the software domain, in line with what reported by Plaza [17], where a similar recommendation is discussed with reference to the maritime domain. The tutorial must be revised to include feedback from the annotators in order to improve their understanding of the task and increase their motivation to perform it at best. We are planning to increment the number of annotators per group, i.e. by having at least three annotators per group.

6 Related Work

The analysis of NL text messages in online forums, bug-tracking systems or mailing lists has been addressed by research works in HCI, computer-mediated business conversation analysis, and more recently in software engineering. We briefly recall relevant works in the following. Ko et al. [18] perform a linguistic analysis of titles of bug reports to understand how people describe software problems. In their approach they use a part-of-speech tagger to identify nouns, verbs, adjectives, etc. and obtain their frequency. They make reference to an intent of a sentence or *grammatical mood* that is indicated by the verbs, which can help in classifying problem reports from requests, but they only analyze the titles and conclude that the use of such a mood concepts need further investigation. Contrary, we analyze speech acts in the body of the emails of a discussion thread. An automated identification of intentions is presented by Twitchell et al. [19]. This investigation proposes a tool that is based on SAT, dialogue acts and fuzzy logic to analyze transcripts of telephone conversations. The goal of this research is to derive participant profiles based on a map of the intentions expressed in the conversation. Our work aims at supporting automated identification of discussants' intentions in OSS communities, such as requesting for features, reporting bugs, or others. The classification of emails using speech acts is investigated by Carvalho et al. [20]. They are interested in classifying emails regarding office-related interactions as negotiation and delegation of tasks. They introduce the term *email acts* which follows a taxonomy of verbs and nouns and they highlight the fact that sequential email acts in a thread of messages contain information useful for a task-oriented classification. Moreover they consider non-linguistic acts as *deliver*. Analogously, we define speech acts to characterize the communication actions in the context of mailing-list discussions in OSS development and consider non-linguistic acts as attachments. The investigation of speech acts by Ravi et al. [21] on thread of discussions in student forum aims at identifying unanswered questions, to be assigned to an instructor for their resolution. They present some patterns of interaction found in the threads, the patterns correspond to the acts *Responsive* and *Question*.

With reference to Requirements Engineering tasks, Knauss et al. [22] analyze discussion threads for requirements elicitation purposes. They focus on the content of communication between stakeholders to find patterns of communication used by stakeholders when they are seeking clarification on requirements. Their approach is based on a Naive Bayesian classifier, a classification scheme of clarification and some heuristics, with interesting results. Worth mentioning is the work of Galvis Carreño et al. [23] that aims at analyzing messages, or *comments*, from users of software applications. Information extraction techniques and topic modeling are exploited to automatically extract topics, and to provide requirements engineers with a user feedback report, which will support them in identifying candidate new/changed requirements. A similar approach is the one of Guzman et al. [24] where App reviews are the input of an automated tool that supports the tasks of filtering, aggregating and analyzing the reviews by applying topic modeling and sentiment analysis. All the above mentioned research works in the area of Requirements Engineering use NL text messages or documents to discover patterns, relevant topics or identify domain key terms, but none of the them consider SAT based techniques to understand stakeholders' intentions behind their messages. We consider that the application of SAT in Requirements Engineering can be a powerful strategy to understand stakeholder's intentions, thus supporting the analysis of the messages they exchange in current distributed collaboration and deriving requirements knowledge.

7 Conclusion

In this paper we proposed a tool-supported method approach to analyze OSS mailing-list discussions in terms of linguistic and non-linguistic acts. We described how the method builds on the idea that the recognition of *speech-acts* expressed in these conversations is key to reveal discussants' intentions, such as suggesting, or complaining. We introduced a three-phase plan to empirically evaluate it and discussed a first execution of phase one, which involved 20 human annotators that were doing the activity of annotating sentences with intentions. The result of this execution gave us a first dataset of *speech-acts* annotations in the domain of OSS online discussions, which may represent a valuable resource for the research community in itself. So far, the interpretation of the results indicates that human annotators might have experienced difficulties when identifying intentions in sentences, mainly if their expertise is not in Software Engineering. But a replication of the experiment must be performed to draw a stronger conclusion. We are working on improving the study design to build a ground truth. This data would be used for evaluating the effectiveness of the proposed method, which is part of the second phase of our empirical evaluation. We are also using the dataset with machine learning algorithms to identify patterns of speech acts. The long-term objective is to use our tool for supporting the classification of messages as bug reports, feature requests or clarifications. We are also considering to collect online discussions of non OSS projects to evaluate the generalizability of the proposed approach in different distributed development settings.

References

1. Castro-Herrera, C., Cleland-Huang, J.: Utilizing recommender systems to support software requirements elicitation. In: RSSE, pp. 6–10. ACM (2010)
2. Camino, B.M., Milewski, A.E., Millen, D.R., Smith, T.M.: Replying to email with structured responses. Int. J. Hum. Comput. Stud. **48**(6), 763–776 (1998)
3. Searle, J.R.: Speech Acts: An Essay in the Philosophy of Language, vol. 626. Cambridge University Press, Cambridge (1969)
4. Wilson, D., Sperber, D.: Relevance theory. In: Horn, L., Ward, G. (eds.) Handbook of Pragmatics. Blackwell, Oxford (2002)
5. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M.A., Saggion, H., Petrak, J., Li, Y., Peters, W.: Text Processing with GATE (Version 6) (2011)
6. Stolcke, A., Coccaro, N., Bates, R., Taylor, P., Van Ess-Dykema, C., Ries, K., Shriberg, E., Jurafsky, D., Martin, R., Meteer, M.: Dialogue act modeling for automatic tagging and recognition of conversational speech. Comput. Linguist. **26**(3), 339–373 (2000)
7. Bach, K., Harnish, R.M.: Linguistic Communication and Speech Acts. MIT Press, Cambridge (1979)
8. Cowie, J., Lehnert, W.: Information extraction. Commun. ACM **39**(1), 80–91 (1996)
9. Morales-Ramirez, I., Perini, A., Guizzardi, R.: Providing foundation for user feedback concepts by extending a communication ontology. In: Yu, E., Dobbie, G., Jarke, M., Purao, S. (eds.) ER 2014. LNCS, vol. 8824, pp. 305–312. Springer, Heidelberg (2014)
10. Ahrenberg, L., Andersson, M., Merkel, M.: A knowledge-lite approach to word alignment. In: Véronis, J. (ed.) Parallel Text Processing. Text, Speech and Language Technology, vol. 13, pp. 97–116. Springer, Dordrecht (2000)
11. Jurafsky, D., Shriberg, L., Biasca, D.: Switchboard SWBD-DAMSL shallow-discourse-function annotation coders manual, draft 13. Technical report, University of Colorado at Boulder Technical Report 97–02 (1997)
12. Morales-Ramirez, I., Perini, A.: Discovering speech acts in online discussions: a tool-supported method. In: Proceedings of the CAiSE 2014 Forum, pp. 137–144 (2014)

13. Cohen, J.: A coefficient of agreement for nominal scales. Educ. Psychol. Measur. **20**(1), 37–46 (1960)
14. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics **33**, 159–174 (1977)
15. Green, A.M.: Kappa statistics for multiple raters using categorical classifications. In: Proceedings of the Twenty-Second Annual SAS Users Group International Conference (online), March 1997
16. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., Wesslén, A.: Experimentation in Software Engineering - An Introduction. Kluwer Academic Publishers, Norwell (2000)
17. Plaza, S.M.: Teaching performative verbs and nouns in eu maritime regulations. Procedia Soc. Behav. Sci. **141**, 90–95 (2014)
18. Ko, A.J., Myers, B.A., Chau, D.H.: A linguistic analysis of how people describe software problems. In: VLHCC 2006, pp. 127–134. IEEE Computer Society (2006)
19. Twitchell, D.P., Nunamaker, J.F.: Speech act profiling: a probabilistic method for analyzing persistent conversations and their participants. In: International Conference on System Sciences, p. 10. IEEE Computer Society Press (2004)
20. Carvalho, V.R., Cohen, W.W.: On the collective classification of email speech acts. In: ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 345–352. ACM (2005)
21. Ravi, S., Kim, J.: Profiling student interactions in threaded discussions with speech act classifiers. Front. Artif. Intell. Appl. **158**, 357–364 (2007)
22. Knauss, E., Damian, D., Poo-Caamano, G., Cleland-Huang, J.: Detecting and classifying patterns of requirements clarifications. In: RE, pp. 251–260. IEEE (2012)
23. Carreño, L.V.G., Winbladh, K.: Analysis of user comments: an approach for software requirements evolution. In: ICSE, pp. 582–591. IEEE (2013)
24. Guzman, E., Maalej, W.: How do users like this feature? a fine grained sentiment analysis of app reviews. In: RE, pp. 153–162. IEEE (2014)