Moving Disambiguation of Regulations from the Cathedral to the Bazaar

Manasi Patwardhan, Richa Sharma, Abhishek Sainani, Shirish karande, Smita Ghaisas
TCS Research, 54-B Hadpasar Industrial Estate, Pune 411013, India
manasi.patwardhan@tcs.com

Abstract

Regulatory compliance is critical to the existence, continuity, and credibility of businesses. Regulations, however, are ridden with ambiguities that make their comprehension a challenge that seems surmountable only by experts. Experts' involvement in understanding regulatory requirements for every software development project is expensive and not scalable. Having software engineers perform disambiguation of such requirements would be a great value addition. We present our design of a 3-step crowdsourcing workflow that aims to convert the task of disambiguation into a series of micro-tasks to be performed by a crowd of software engineers. We demonstrate that the outcome of this workflow is at par with the expert-enabled disambiguation at 4.5 times lower cost.

Introduction

Since regulations aim to safeguard the wellbeing of citizens, they are written with a great rigor and discipline to minimize incidents of violations. However, their diction is so highly specialized that it is almost incomprehensible to business communities, who need to ensure regulatory compliance. Mechanisms to assure and demonstrate regulatory compliance have been researched for a long time (Breaux, Vail, and Anton 2006). However, researchers have noted that the ambiguities in regulations pose a challenge to requirements engineers and thus the process of deriving system requirements tends to be error prone.

Massey et.al. have created a legal ambiguity taxonomy for identifying and classifying ambiguities in regulations that govern software systems (Massey et al. 2014). In their experiments involving software engineers (undergraduate and graduate students) in resolving ambiguities, they found that the engineers could identify ambiguous terms or phrases in a regulation statement, but were not able to agree on a consistent rationale. The authors therefore suggest that software engineers need expert inputs to validate their interpretations of ambiguities (Massey et al. 2015). Involving legal experts in every software project is expensive and therefore not scalable. In our work, we explore this line of research further by involving a crowd of professional software engineers to not only identify ambiguities; but also to disambiguate regulations, with an aim to find viable and scalable alternative to the current expensive mode of disambiguation.

We conduct a series of pilot crowdsourcing experiments that help us design a 3-step workflow composed entirely of micro-tasks. Micro-task crowdsourcing has a potential which is yet to be fully explored in the field of software engineering (Adriano and van der Hoek 2016; Weidema et al. 2016; Zhao and van der Hoek 2015; LaToza and van der Hoek 2016). We employ micro-tasking to break down the complex task of disambiguation into smaller chunks of tasks, sequentially executed as the steps of the workflow, causing less cognitive load, and resulting in a better quality and scalability. We use an already proven method of peer-evaluation as a part of crowdsourcing workflow to produce reliable data (Goto, Ishida, and Lin 2016; Ambati, Vogel, and Carbonell 2012; Hansen et al. 2013; Huang and Fu 2013). The outcome of the micro-task executed in the i^{th} step of the workflow is peer-evaluated in the $(i+1)^{th}$ step, ensuring successive and incremental enhancement in quality.

For other complex tasks, such as tasks in linguistics (Hong and Baker 2011) and the medical domain (Zhai et al. 2013), use of lay crowd to replace experts is proven to be a feasible option leading to more scalable and less costlier solution for data collection. On the similar lines, in this work, we prove that the crowd annotations we receive for ambiguity detection and disambiguation, upon reaching consensus, match with those made by the experts, providing a clear indication that the wisdom of software engineers' can equate experts'. We demonstrate that our approach moves this highly specialized task of disambiguation *From the Cathedral to the Bazaar* (Raymond 1999) and leads to 4.5 times reduction in cost of experts.

Disambiguation

There are six distinct types of regulation ambiguities defined by Massey et al (Massey et al. 2014), viz. 1. Lexical 2. Syntactic 3. Semantic 4. Incompleteness 5. Vagueness and 6. Referential. As a part of this study, we have focused on the first three types of ambiguities. A term /phrase in a regulation statement is lexically ambiguous if it has multiple dictionary meanings. Disambiguation here would mean explicating the exact meaning. Syntactic ambiguity points at multiple word associations leading to multiple parse trees and disambiguation here amounts to clarifying the scope of the word association. Semantic ambiguity occurs if a statement is not self-contained and disambiguation would mean providing the additional contextual information for interpretation. Table 1 illustrates examples of regulation state-

Ambiguity	Regulatory Statement (marked term in bold)	Question	Answers (valid answers in bold)
Lexical	Implement hardware, software, and/or procedural mechanisms that **record** and examine activity in information systems that contain or use electronic protected health information.	In the given sentence what is the meaning of word 'record'?	a) **to put in writing or digital form for future use.** b) information stored on a computer. c) best performance. d) **to make a permanent or official note of.** e) a piece of evidence from the past.
Syntactic	Implement policies and procedures to address the **final disposition of** electronic protected health information, and/or the hardware or electronic media on which it is stored.	In the given sentence the phrase 'final disposition of' refers to?	a) **electronic protected health information** b) policies c) **hardware** d) address e) **electronic media**
Semantic	Implement hardware, software, and/or procedural mechanisms that record and **examine activity** in information systems that contain or use electronic protected health information.	What does "examine activity" mean?	a) Keep a log of what was done b) Notify admin that something was done c) Stop/block what is being done d) **Identify what was done** e) **Classify what was done**

Table 1: Ambiguity Examples

ments per ambiguity type, with an ambiguous term/ phrase marked. A question posed on the term would highlight the source and type of ambiguity and a list of valid explanatory answers to the question would result in disambiguation. For our study, we sought ground truth inputs from three experts who have worked with Health Insurance Portability and Accountability Act (HIPAA) regulations (Dwyer III, Weaver, and Hughes 2004) for more than 3 years. We asked experts to select 5 regulation statements from HIPAA, each having terms/phrases depicting the three types of ambiguities.

Pilot Tasks

To conducted pilot crowdsourcing experiments with a specific intent to evaluate the design trade-offs w.r.t. cognitive load, scalability and quality. Our experiments consisted of 3 crowdsourcing tasks to collect regulation disambiguation data for 5 regulation statements. We targeted a crowd of 30 professional software engineers with 3 to 4 years of experience (henceforth referred to as crowd workers). They were asked to perform this task during their working hours. In the first task, we tried to achieve disambiguation in a single step. We presented regulation statements and asked the crowd workers to either write their own policy statement(s) in response to the regulations or produce policies from credible sources which comply with the regulations. These policy statements would serve as explanatory texts for disambiguation. We achieved a very low participation (3 out of 30 crowd workers) with 27% error rate (incorrect/spam inputs) and completion time of average 3 minutes per regulation statement indicating a high cognitive load. To address this issue, as a part of second pilot task, we designed the disambiguation as a two-step process: (i) Pose questions about the ambiguities and (ii) Provide answers. We still got a low participation (4 out of 30) with a small reduction in the error rate 24% and completion time (average 2.5 minutes per regulation statement). Thus, the reduction in cognitive load was not significant enough.

Both these pilot tasks sought textual inputs, leading to high cognitive load. Furthermore, algorithmically evaluating consensus is a challenge. To address this challenge, as the third pilot task, we decided to seek discrete responses rather than textual ones. We presented regulation statements and supplementary text in the form of policy statements extracted from university websites which publish their HIPAA policies (NYU). The crowd provided binary annotations indicating whether a given policy in response to a regulation seemed to implement what was intended by the regulation statement. We received an increased participation (24 out of 30) with reduced completion time (average 1 minute per task) alleviating cognitive load. However, 74% of the responses were incorrect. Moreover, the design of this pilot task is not scalable as it requires collection of policy statements for every regulation statement from web sources.

For all the three pilot tasks, we noted that the tasks involved comprehension of the regulation and strategizing for compliance. The comprehension was subjective because our crowd consisted of software engineers working in different domains. Accordingly, their foci while formulating or selecting policy statements and/or posing questions as responses were different. This led to a lot of variations in the responses, making it an impossible task to draw a consensus on the source of ambiguity. To address this challenge, we needed to direct their attention to specific ambiguities in the regulation statements, which are indicated by specific terms or phrases.

Workflow Design

With the observations made from our pilot studies, we arrived at the following conclusions: (1) The complex task of disambiguation has to be divided into smaller chunks of micro-tasks, so that, the reduction in cognitive load would achieve better participation and quality of inputs (2) The micro-task design should be (i) amenable to achieve scalability, (ii) lead to discrete set of responses, which eases the process of achieving consensus, (iii) highlight source of ambiguity in a regulation statement, alleviating the problem of varying focal points. (3) There is a need to design a workflow which consists of a sequence of micro-tasks, such that the solicited crowd responses in the i^{th} step of the workflow, are reviewed and validated by other set of crowd workers (peers), in the $(i+1)^{th}$ step, followed by providing responses on validated inputs. Such peer-evaluation

would ensure successive and incremental enhancements in disambiguation without expert involvement and also would achieve crowd engagement since they are required to *rationalize* their validations by providing responses. The resultant workflow is described below.

Workflow Step 1: Marking Ambiguous Terms and Posing Questions In this micro-task a crowd worker is (i) presented with a regulation statement, (ii) asked to mark a (set of) term(s) and/or phrase(s) in the statement, which are ambiguous, and (iii) to pose a (set of) question(s) to every term or phrase marked, the answer to which would cause disambiguation. We apply majority voting to find consensus on terms/phrases. Thus, the outcome of this micro-task is a set of regulation statements with valid set of ambiguous terms/phrases and a set of corresponding questions for each term.

Workflow Step 2: Validating Questions and Providing Answers The outcome of the prior micro-task is used as an input here. A crowd worker is (i) presented with a regulation statement, along with a validated ambiguous term or phrase and the corresponding set-of questions (ii) asked to validate each question for its meaning, grammar,and applicability (if the answer actually leads to disambiguation), by providing binary input (Valid/Invalid). (iii) For all the questions marked as valid, (s)he needs to provide a succinct answer to the question, which would cause disambiguation. We ensure that the set-of crowd workers attempting this micro-task are different than those who have worked on Step 1; or if they are the same set of workers, they do not get to work on their own set of responses (questions) from the earlier step. We use majority voting for consensus on the valid set of questions. Thus, the outcome of this step is set of regulation statements containing a term and/or a phrase marked as ambiguous for which at least one question is validated. In addition, each of these questions is accompanied by a (set-of) answer(s) provided by the crowd.

Workflow Step 3: Validating Answers The outcome of the prior micro-task serves an input to this micro-task. A crowd worker is asked to (i) read the regulatory statement along with the marked term or phrase, (ii) read the question posed on the marked term, and (iii) choose any subset of the answers as valid answers to the posed question, considering the context of the regulation statement. Her response to an answer would be 'yes' if she thinks that the answer is valid; otherwise it would be 'no'. We follow the same strategy as discussed in the prior step to select and allocate micro-tasks to the crowd workers. We use majority voting for consensus on the valid set of answers.

Data Collection and Analysis

For step 1 in the workflow, we selected five regulation statements from our expert annotated data. We remind that each statement contains terms or phrases that demonstrate all the three types of ambiguities. 15 crowd workers marked 46 unique set of terms. Of these, 19 terms were majority-voted as ambiguous. For Step 2 in the workflow, for the same set of five regulation statements, we selected 3 terms/phrases (one per ambiguity type) for which crowd consensus was achieved in step 1 and were in agreement with expert annotations. We also included 3 randomly chosen questions posed by the crowd workers to each of these terms in the prior micro-task. For each of these 45 micro-tasks (5 regulatory statements * 3 terms * 3 questions) we received inputs from a distinct set of 15 crowd workers. After majority voting, we had 35 valid questions and 10 invalid questions. For Step 3, we selected the same 5 regulation statements with the same set of 3 ambiguous terms. For each term, we randomly selected 1 majority-voted question that matched with that from experts. For every question, we randomly selected 5 answers provided in the earlier step by crowd workers. Thus, we had a total of 75 micro-tasks. For each answer, we expected 5 binary responses from 15 crowd workers. After majority voting, we had 40 answers marked as valid and 35, invalid. The crowd consensus (majority voting) results were validated by experts. The results of all three micro-tasks are illustrated in table 2.

Crowd Consensus		Expert Annotations			P	R	F
		Valid	Invalid	Total			
Step 1	Valid	17	2	19	89%	85%	87%
	Invalid	3	24	27			
	Total	20	26	46			
Step 2	Valid	30	5	35	86%	97%	91%
	Invalid	1	9	10			
	Total	31	14	45			
Step 3	Valid	31	7	38	82%	91%	86%
	Invalid	3	34	37			
	Total	34	41	75			
P: Precision, R: Recall, F: F-score							

Table 2: Confusion Matrix for the Workflow

To complete these micro-tasks the crowd workers took 1 to 5 minutes, which is of the same order of the time taken for completing the pilot-tasks. However, we received a 100% participation, with higher quality inputs. This shows that the micro-tasking and the workflow have reduced the cognitive load and achieved a higher crowd engagement.

Projected Cost Analysis HIPAA has about 5000 regulation statements. A crowd of software engineers working for an hour daily at the rate of 4 USD, spending 3 minutes per task per worker would cost USD 86.5 Kfor HIPPA annotations. On the other hand, legal experts working for 200 USD per hour (rate validated by legal experts in a personal communication) at the rate of 1.5 minutes per task would cost USD 395 K (4.5 times that of software engineers).

Conclusion and Future Work

We have early indications of success for disambiguating regulations by utilizing a crowd consisting of software engineers. Our approach could lead to a 4.5 fold reduction in cost compared to employing legal experts. In future, we intend to extend this work to other ambiguity types: referential, incompleteness and vagueness and employ techniques (such as adaptive task allocation, online Expectation Maximization, active learning, etc.) which could help acquire annotations on a large scale, so that, machine/ deep learning algorithms can be trained to provide automated disambiguation.

References

Adriano, C. M., and van der Hoek, A. 2016. Exploring microtask crowdsourcing as a means of fault localization. *arXiv preprint arXiv:1612.03015*.

Ambati, V.; Vogel, S.; and Carbonell, J. 2012. Collaborative workflow for crowdsourcing translation. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, 1191–1194. ACM.

Breaux, T. D.; Vail, M. W.; and Anton, A. I. 2006. Towards regulatory compliance: Extracting rights and obligations to align requirements with regulations. In *Requirements Engineering, 14th IEEE International Conference*, 49–58. IEEE.

Dwyer III, S. J.; Weaver, A. C.; and Hughes, K. K. 2004. Health insurance portability and accountability act. *Security Issues in the Digital Medical Enterprise* 72(2):9–18.

Goto, S.; Ishida, T.; and Lin, D. 2016. Understanding crowdsourcing workflow: modeling and optimizing iterative and parallel processes. In *Fourth AAAI Conference on Human Computation and Crowdsourcing*.

Hansen, D. L.; Schone, P. J.; Corey, D.; Reid, M.; and Gehring, J. 2013. Quality control mechanisms for crowdsourcing: peer review, arbitration, & expertise at familysearch indexing. In *Proceedings of the 2013 conference on Computer supported cooperative work*, 649–660. ACM.

Hong, J., and Baker, C. F. 2011. How good is the crowd at real wsd? In *Proceedings of the 5th linguistic annotation workshop*, 30–37. Association for Computational Linguistics.

Huang, S.-W., and Fu, W.-T. 2013. Enhancing reliability using peer consistency evaluation in human computation. In *Proceedings of the 2013 conference on Computer supported cooperative work*, 639–648. ACM.

LaToza, T. D., and van der Hoek, A. 2016. Crowdsourcing in software engineering: Models, motivations, and challenges. *IEEE software* 33(1):74–80.

Massey, A. K.; Rutledge, R. L.; Antón, A. I.; and Swire, P. P. 2014. Identifying and classifying ambiguity for regulatory requirements. In *Requirements Engineering Conference (RE), 2014 IEEE 22nd International*, 83–92. IEEE.

Massey, A. K.; Rutledge, R. L.; Antón, A. I.; Hemmings, J. D.; and Swire, P. P. 2015. A strategy for addressing ambiguity in regulatory requirements. Technical report, Georgia Institute of Technology.

New york university hipaa policies.

Raymond, E. 1999. The cathedral and the bazaar. *Knowledge, Technology & Policy* 12(3):23–49.

Weidema, E. R.; López, C.; Nayebaziz, S.; Spanghero, F.; and van der Hoek, A. 2016. Toward microtask crowdsourcing software design work. In *CrowdSourcing in Software Engineering (CSI-SE), 2016 IEEE/ACM 3rd International Workshop on*, 41–44. IEEE.

Zhai, H.; Lingren, T.; Deleger, L.; Li, Q.; Kaiser, M.; Stoutenborough, L.; and Solti, I. 2013. Web 2.0-based crowdsourcing for high-quality gold standard development in clinical natural language processing. *Journal of medical Internet research* 15(4).

Zhao, M., and van der Hoek, A. 2015. A brief perspective on microtask crowdsourcing workflows for interface design. In *Proceedings of the Second International Workshop on CrowdSourcing in Software Engineering*, 45–46. IEEE Press.