Inferring Ontology Fragments from Semantic Role Typing of Lexical Variants

Mitra Bokaei Hosseini[1](✉), Travis D. Breaux[2](✉), and Jianwei Niu[1](✉)

[1] Computer Science Department, University of Texas, San Antonio, TX, USA
{mitra.bokaeihosseini,jianwei.niu}@utsa.edu

[2] Institute of Software Research, Carnegie Mellon University, Pittsburgh, USA
breaux@cs.cmu.edu

Abstract. **[Context and Motivation]** Information systems depend on personal data to individualize services. To manage privacy expectations, companies use privacy policies to regulate what data is collected, used and shared. However, different terminological interpretations can lead to privacy violations, or misunderstandings about what behavior is to be expected. **[Question/Problem]** A formal ontology can help requirements authors to consistently check how their data practice descriptions relate to one another and to identify unintended interpretations. Constructing an empirically valid ontology is a challenging task since it should be both scalable and consistent with multi-stakeholder interpretations. **[Principle Ideas/Results]** In this paper, we introduce a semi-automated semantic analysis method to identify ontology fragments by inferring hypernym, meronym and synonym relationships from morphological variations. The method employs a shallow typology to categorize individual words, which are then matched automatically to 26 reusable semantic rules. The rules were discovered by classifying 335 unique information type phrases extracted from 50 mobile privacy policies. The method was evaluated on 109 unique information types extracted from six privacy policies by comparing the generated ontology fragments against human interpretations of phrase pairs obtained by surveying human subjects. The results reveal that the method scales by reducing the number of otherwise manual paired comparisons by 74% and produces correct fragments with a 1.00 precision and 0.59 recall when compared to human interpretation. **[Contributions]** The proposed rules identify semantic relations between a given lexeme and its morphological variants to create a shared meaning between phrases among end users.

Keywords: Requirements engineering · Natural language processing
Ontology

1 Introduction

Mobile and web applications (apps) are increasingly popular due to the convenient services they provide in different domains of interest. According to a 2015 PEW Research Center study, 64% of Americans own a smart phone [1]. They found that smart phone users typically check health-related information online (62% of Americans), conduct online banking (54%), and look for job-related information (63%). To fulfill user needs and business requirements, these apps collect different categories of personal information, such as

E. Kamsties et al. (Eds.): REFSQ 2018, LNCS 10753, pp. 39–56, 2018.
https://doi.org/10.1007/978-3-319-77243-1_3

friends' phone numbers, photos and real-time location. Regulators require apps to provide users with a legal privacy notice, also called a privacy policy, which can be accessed by users before installing the app. For example, the California Attorney General's office recommends that privacy policies list what kind of personally identifiable data is collected, how it is used, and with whom it is shared [2]. Privacy policies contain critical requirements that inform stakeholders about data practices [3]. Due to different stakeholder needs, there can be disparate viewpoints regarding what is essentially the same subject matter [4]. Stakeholders use different words for the same domain, which reduces shared understanding of the subject and leads to a misalignment among the designers' intention, and expectations of policy writers and regulators [5].

Data practices are commonly described in privacy polices using hypernymy [6], which occurs when a more abstract information type is used instead of a more specific information type. Hypernymy permits multiple interpretations, which can lead to ambiguity in the perception of what exact personal information is used. To address this problem, companies can complement their policies with a formal ontology that explicitly states what kinds of information are included in the interpretations of data-related concepts. Initial attempts to build any ontology can require comparing each information type phrase with every other phrase in the policy, and assigning a semantic relationship to each pair. However, considering a lexicon built from 50 policies that contains 351 phrases, an analyst must make $(351 \times 350)/2 = 61{,}425$ comparisons, which is over 200 h of continuous comparison by one analyst.

In this paper, we describe a semi-automated semantic analysis method that uses lexical variation of information type phrases to infer ontological relations, such as hypernyms. Instead of performing paired comparisons, the analyst spends less than one hour typing the phrases, and then a set of semantic rules are automatically applied to yield a subset of all possible relations. The rules were first discovered in a grounded analysis of information types extracted from 50 privacy policies for a manual ontology construction approach [7]. To improve the semantic relations inferred using these initial set of rules, we established a ground truth by asking human subjects to perform the more time-consuming task of comparing phrases in the lexicon. We then compared the results of the semantic rules against these human interpretations, which led to identifying additional semantic rules. Finally, we evaluated the improved semantic rules using 109 unique information types extracted from six privacy policies, and human subject surveys to measure the correctness of the results produced by the semantic rules.

This paper is organized as follows: in Sect. 2, we discuss terminology and the theoretical background; Sect. 3 presents a motivating example; in Sect. 4, background and related work are discussed; in Sect. 5, we introduce our semi-automated method for discovering ontology fragments consisting of hypernyms, meronyms and synonyms; In Sect. 6, we explain the experimental setup; in Sect. 7, we present results of evaluating this technique against human subject-surveyed information type pairs, before presenting our discussion and conclusion in Sects. 8 and 9.

2 Important Terminology and Theoretical Background

In this section, we define the terminology and present the theoretical background.

2.1 Terminology

- *Hypernym* – a noun phrase, also called a superordinate term, that is more generic than another noun phrase, called the hyponym or subordinate term.
- *Meronym* – a noun phrase that represents a part of a whole, which is also a noun phrase and called a holonym.
- *Synonym* – a noun phrase that has a similar meaning to another noun phrase.
- *Lexicon* – a collection of phrases or concept names that may be used in an ontology.
- *Ontology* – a collection of concept names and logical relations between these concepts, including hypernymy, meronymy and synonymy, among others [8].

2.2 Theoretical Background on Description Logic

Description Logic (DL) ontologies enable automated reasoning, including the ability to infer which concepts subsume or are equivalent to other concepts in the ontology. We chose the DL family $\mathcal{AL}$, which is PSPACE-complete for concept satisfiability and concept subsumption. In this paper, reasoning in DL begins with a TBox T that contains a collection of concepts and axioms based on an interpretation $\mathfrak{I}$ that consists of a nonempty set $\Delta^{\mathfrak{I}}$, called the domain of interpretation. The interpretation function $.^{\mathfrak{I}}$ maps concepts to subsets of $\Delta^{\mathfrak{I}}$: every atomic concept C is assigned a subset $C^{\mathfrak{I}} \subseteq \Delta^{\mathfrak{I}}$, the top concept $\top^{\mathfrak{I}} = \Delta^{\mathfrak{I}}$ has the interpretation $^{\top}$.

The $\mathcal{AL}$ family includes operators for concept union and intersection, and axioms for subsumption, and equivalence with respect to the TBox. Subsumption is used to describe individuals using generalities, and we say a concept C is subsumed by a concept D, written $T \vDash C \sqsubseteq D$, if $C^{\mathfrak{I}} \subseteq D^{\mathfrak{I}}$ for all interpretations $\mathfrak{I}$ that satisfy the TBox T. The concept C is equivalent to a concept D, written $T \vDash C \equiv D$, if $C^{\mathfrak{I}} = D^{\mathfrak{I}}$ for all interpretations $\mathfrak{I}$ that satisfy the TBox T.

The DL enables identifying which lexicon phrases directly or indirectly share meanings, called an interpretation in DL. Each lexicon phrase is mapped to a concept in the TBox T. We express a hyponym concept C in relation to a hypernym concept D using subsumption $T \vDash C \sqsubseteq D$, and for two concepts C and D that correspond to synonyms, we express these as equivalent concepts $T \vDash C \equiv D$. For meronymy, we define a part-whole relation *partOf* that maps parts to wholes as follows: a part concept C that has a whole concept D, such that $T \vDash C \sqsubseteq (\textit{partOf}\ D)$. We express the DL ontology using the Web Ontology Language[1] (OWL) version 2 DL and the HermiT[2] OWL reasoner.

[1] https://www.w3.org/TR/owl-guide.

[2] http://www.hermit-reasoner.com/.

3 Motivating Example

We now provide an example statement from the WhatsApp privacy policy with example interpretations inferred from the statement to demonstrate the problem.

> **Statement:** You must provide certain devices, software, and data connections to use our Services, which we otherwise do not supply.

In this statement, "device" is an abstract information type that can be interpreted in many ways. Here are three example strategies for obtaining an interpretation:

1. If device is a super-ordinate concept, then we infer that mobile device is a kind device, therefore, the collection of information also applies to mobile devices.
2. If device is a kind of system with components, settings, etc., and we know that a device can have an IP address, then WhatsApp may collect device IP address. This interpretation is reached using a meronymy relationship between device and device IP address.
3. By use both strategies (1) and (2), together, we can infer that the collection statement applies to mobile device IP address, using both hypernymy and meronymy.

These interpretations are based on human knowledge and experience, and there is a need to bridge the gap between linguistic information types in privacy policies and knowledge of the world. In the above examples, mobile device, device IP address, and mobile device IP address are variants of a common lexeme: "device." We use the syntactic structure of lexical variants to infer semantics and construct lexical ontologies that are used to bridge this knowledge gap.

4 Related Work

In requirements engineering, two approaches are defined for codifying knowledge: naïve positivism, and naturalistic inquiry [9]. Positivism refers to the world with a set of stable and knowable phenomena, often with formal models. Naturalistic inquiry (NI) refers to constructivist views of knowledge that differ across multiple human observations. The research in this paper attempts to balance among these two viewpoints by recognizing that information types are potentially unstable and intuitive concepts. Our approach permits different interpretations, before reducing terminological confusion to reach a shared understanding through formal ontologies. We now review prior research on ontology in privacy.

4.1 Ontology in Security and Privacy Policy

Heker et al. developed a privacy ontology for e-commerce transactions which includes concepts about privacy mechanisms and principles from legislative documents [10]. Bradshaw et al. utilize an ontology that distinguishes between authorization and obligations for a policy service framework that forces agents to check their behavior with specifications [11]. Kagal et al. constructed an ontology to enforce access control policies in a web service model [12]. Syed et al. developed an ontology that provides a common understanding of cybersecurity and unifies commonly used cybersecurity standards [13]. Breaux et al. utilize an ontology that includes simple hierarchies for actors and information types to infer data flow traces across separate policies in multi-tier applications [14]. To our knowledge, our work is the first privacy-related lexical ontology that formally conceptualizes information types extracted from policies with their implied semantic relations. The initial version of this ontology has been used to find conflicts between mobile app code-level method calls and privacy policies [15].

4.2 Constructing an Ontology

There is no standard method to build an ontology [4], yet, a general approach includes identifying the ontology purpose and scope; identifying key concepts leading to a lexicon; identifying relations between lexicon concepts; and formalizing those relations. A lexicon consists of terminology in a domain, whereas ontologies organize terminology by semantic relations [16]. Lexicons can be constructed using content analysis of source text, which yields an annotated corpus. Breaux and Schaub empirically evaluated crowdsourcing to create corpora from annotated privacy policies [17]. Wilson et al. employed crowdsourcing to create a privacy policy corpus from 115 privacy policies [18].

WordNet is a lexical database which contains English words and their forms captured from a newswire corpus, and their semantic relations, including hypernymy and synonymy [19]. Our analysis shows that only 14% of our lexicon was found in WordNet, mainly because our lexicon is populated with multi-word phrases. Moreover, meronymy relations are missing from WordNet.

Snow et al. presented a machine learning approach using hypernym-hyponym pairs in WordNet to identify additional pairs in parsed sentences of newswire corpus [20]. This approach relies on explicit expression of hypernymy pairs in text. Bhatia et al. [21] identified and applied a set of 72 Hearst-related patterns [22] to 30 privacy policies to extract hypernymy pairs. This approach yields hypernyms for only 24% of the lexicon. This means the remaining 76% of the lexicon must be manually analyzed to construct an ontology. These approaches fail to consider the semantic relations between the morphological variants of a nominal, which may not be present in the same sentence as the nominal. Our proposed model identifies these variants with semantic relations.

5 Ontology Construction Method Overview

The ontology construction method (see Fig. 1) consists of 7 steps: (1) collecting privacy policies; (2) itemizing paragraphs in the collected privacy policies; (3) annotating the itemized paragraphs by crowd workers based on a specific coding frame; (4) employing an entity extractor developed by Bhatia and Breaux [6] to analyze the annotations and extract information types which results in an information type lexicon (artifact A in Fig. 1); (5) preprocessing the phrases in the lexicon; (6) assigning role types to each pre-processed phrase that yields information type phrases with associated role sequences; (7) automatically matching the type sequence of each phrase to a set of semantic rules to yield a set of ontology fragments consisting of hypernym, meronym, and synonym relationships. Steps

1–3 are part of a crowdsourced content analysis task based on Breaux and Schaub [17]. Our contribution in this paper includes steps 5–7 which utilizes an information type lexicon to construct an ontology.

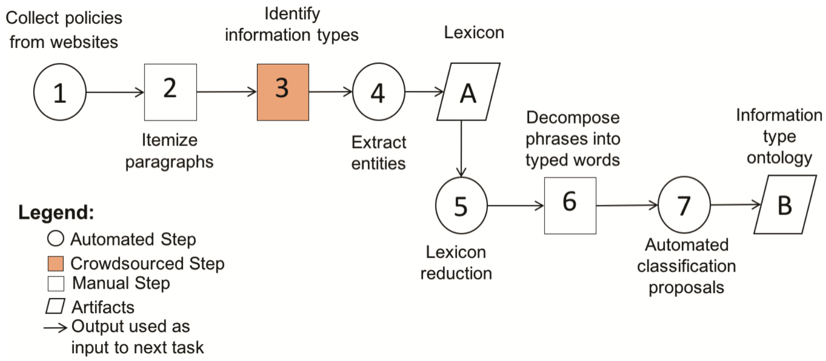

Collect policies from websites
1
2
Itemize paragraphs
Identify information types
3
4
Extract entities
Lexicon
A
5
Lexicon reduction
Decompose phrases into typed words
6
7
Automated classification proposals
Information type ontology
B
Legend:
○ Automated Step
▨ Crowdsourced Step
□ Manual Step
▱ Artifacts
→ Output used as input to next task

Fig. 1. Overview of ontology construction method

5.1 Acquiring the Mobile Privacy Policy Lexicon

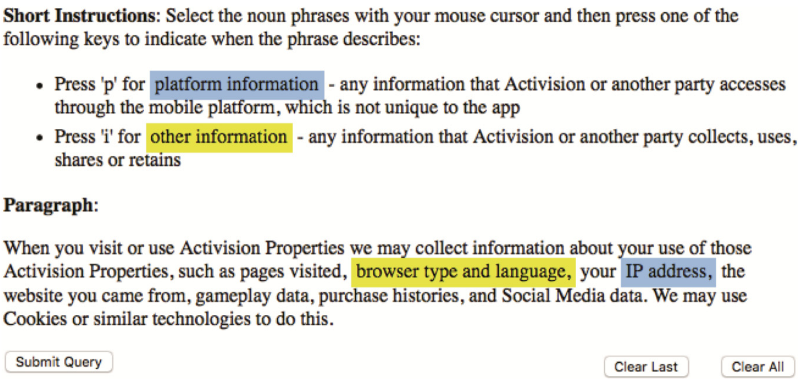
The mobile privacy policy lexicon (artifact A in Fig. 1) was constructed using a combination of crowdsourcing, content analysis and natural language processing (NLP). In step 1 (see Fig. 1), we selected the top 20 mobile apps across each of 69 sub-categories in Google Play[3]. From this set, we selected apps with privacy policies, removing duplicate policies when different apps shared the same policy. Next, we selected only policies that match the following criteria: format (plain text), language (English), and explicit statements for privacy policy; yielding 501 policies, from which we randomly selected 50 policies. In step 2, the 50 policies were segmented into ~120 word paragraphs using the method described by Breaux and Schaub [17]; yielding 5,932 crowd worker annotator tasks with an average 98 words per task for input to step 3.

In step 3, the annotators select phrases corresponding to one of two category codes in a segmented paragraph as described below for each annotator task, called a Human Intelligence Task (HIT). An example HIT is shown in Fig. 2.

- *Platform Information:* any information that the app or another party accesses through the mobile platform which is not unique to the app.
- *Other Information:* any other information the app or another party collects, uses, shares or retains.

These two category codes were chosen, because our initial focus is on information types that are automatically collected by mobile apps and mobile platforms, such as "IP address," and "location information." The other information code is used to ensure that annotators remain vigilant by classifying and annotating all information types.

[3] https://play.google.com.



Short Instructions: Select the noun phrases with your mouse cursor and then press one of the following keys to indicate when the phrase describes:

- Press 'p' for platform information - any information that Activision or another party accesses through the mobile platform, which is not unique to the app
- Press 'i' for other information - any information that Activision or another party collects, uses, shares or retains

Paragraph:

When you visit or use Activision Properties we may collect information about your use of those Activision Properties, such as pages visited, browser type and language, your IP address, the website you came from, gameplay data, purchase histories, and Social Media data. We may use Cookies or similar technologies to do this.

Fig. 2. Example HIT shown to a crowd worker

In step 4, we selected only platform information types when two or more annotators agreed on the annotation to construct the lexicon. This number follows the empirical analysis of Breaux and Schaub [17], which shows high precision and recall for two or more annotators on the same HIT. Next, we applied an entity extractor [6] to the selected annotations to itemize the platform information types into unique entities included in the privacy policy lexicon.

Six privacy experts, including the authors, performed the annotations. The cumulative time to annotate all HITs was 59.8 h across all six annotators, yielding a total 720 annotations in which two or more annotators agreed on the annotation. The entity extractor reduced these annotations down to 351 unique information type names, which comprise the initial lexicon.

In step 5, the initial lexicon was reduced as follows:

a. Plural nouns were changed to singular nouns, e.g., “peripherals” is reduced to “peripheral.”
b. Possessives were removed, e.g., “device’s information” is reduced to “device information.”
c. Suffixes “-related,” “-based,” and “-specific” are removed, e.g., “device-related information” is reduced to “device information.”

This reduced the initial lexicon by 16 types to yield a final lexicon with 335 types.

5.2 Semantic Role Typing of Lexicon Phrases

Figure 3 shows an example phrase, “mobile device IP address” that is decomposed into the atomic phrases: “mobile,” “device,” “IP,” “address,” based on a 1-level, shallow typology. The typology links atomic words from a phrase to one of six roles: (M) modifiers, which describe the quality of a thing, such as “mobile” and “personal;” (T) things, which is a concept that has logical boundaries and which can be composed of other things; (E) events, which describe action performances, such as “usage,” “viewing,” and “clicks;” (G) agents, which describe actors who perform actions or possess things; (P) property, which describes the functional feature of an agent, place or thing, such as “date,” “name,” “height;” and (α) which is an abstract type that indicates “information,” “data,” “details,” and any other synonym of “information.” In an information type ontology, the concept that corresponds to the α type is the most general, inclusive concept.

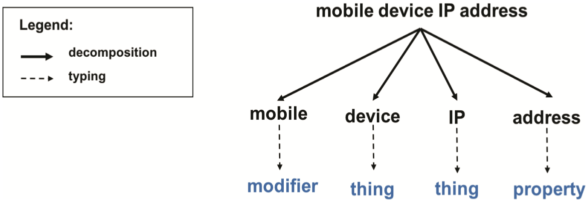


Fig. 3. Example lexicon phrase, grouped and typed

In step 6, the analyst reviews each information type phrase in the lexicon and assigns role types to each word. The phrase typing is expressed as a continuous series of letters that correspond to the role typology. Unlike the quadratic number of paired comparisons required to identify relationships among lexicon phrases, this typing step is linear in the size of the lexicon. Furthermore, word role types can be reused across phrases that reuse words to further reduce the time needed to perform this step. Next, we introduce the semantic rules that are applied to the typed phrases in the lexicon.

5.3 Automated Lexeme Variant Inference

We now describe step 7, which takes as input the typed, atomic phrases produced in step 6 to apply a set of semantic rules to infer variants and their ontological relationships, which we call *variant relationships*. Rules consist of a type pattern and an inferred ontological relationship. The type pattern is expressed using the typology codes described in Sect. 5.2. The rules below were discovered by the first and second author who classified the 335 pre-processed lexicon phrases using the typology as a second-cycle coding, which is a qualitative research method [23]. Subscripts indicate the order of same-typed phrases in asymmetric ontological relations:

Hypernymy Rules

H1. M_α implies that $M\_\alpha \sqsubseteq \alpha$, e.g., “unique information” is a kind of “information.”

H2. $M_1\_M_2\_\alpha$ implies that $M_1\_M_2\_\alpha \sqsubseteq (M_1\_\alpha \sqcup M_2\_\alpha)$, e.g., “anonymous demographic information” is a kind of “anonymous information” and “demographic information.”

H3. $M\_T_1\_T_2$ implies $M\_T_1\_T_2 \sqsubseteq (M\_\alpha \sqcup T_1\_T_2)$ and $T_1\_T_2 \sqsubseteq \mathit{partOf}\ M\_T_1$, e.g., “mobile device hardware” is a kind of “mobile information,” “device hardware,” and “device hardware” is a part of “mobile device.”

H4. M_T_α implies $M\_T\_\alpha \sqsubseteq (M\_\alpha \sqcup T\_\alpha)$, e.g., “mobile device information” is a kind of “mobile information” and “device information.”

H5. M_T_P implies $M\_T\_P \sqsubseteq M\_\alpha$ and $M\_T\_P \sqsubseteq \mathit{partOf}\, M\_T$ and $T\_P \sqsubseteq \mathit{partOf}\, M\_T$, e.g., "mobile device name" is a kind of "mobile information" and a part of "mobile device" and "device name" is a part of "mobile device."

H6. M_G_α implies that $M\_G\_\alpha \sqsubseteq (M\_\alpha \sqcup G\_\alpha)$, e.g. "aggregated user data" is a kind of "aggregated data" and "user data."

H7. T_α implies $T\_\alpha \sqsubseteq \alpha$, e.g., "device information" is a kind of "information."

H8. $T_1\_T_2\_\alpha$ implies $T_1\_T_2\_\alpha \sqsubseteq (T_1\_\alpha \sqcup T_2\_\alpha)$, e.g., "device log information" is a kind of "device information" and "log information."

H9. G_α implies that $G\_\alpha \sqsubseteq \alpha$, e.g. "user information" is a kind of "information."

H10. G_T implies that $G\_T \sqsubseteq (G\_\alpha \sqcup T)$, e.g., "user content" is a kind of "user information" and "content."

H11. G_P implies that $G\_P \sqsubseteq (G\_\alpha \sqcup P)\ \mathit{and}\ G\_P \sqsubseteq \mathit{partOf}\, G$, e.g., "user name" is a kind of "user information" and "user name" is a part of "user."

H12. E_α implies that $E\_\alpha \sqsubseteq \alpha$, e.g. "usage data" is a kind of "data."

H13. T_E implies that $T\_E \sqsubseteq (T \sqcup E \sqcup E\_\mathit{lemma})$, e.g., "page viewed" is a kind of "page," "viewed," and "view."

Meronymy Rules

M1. $T_1\_T_2$ implies $T_1\_T_2 \sqsubseteq \mathit{partOf}\, T_1\ \mathit{and}\ T_1\_T_2 \sqsubseteq T_2$, e.g., "device hardware" is a part of "device" and is a kind of "hardware."

M2. $T_1\_M\_T_2$ implies $T_1\_\mathrm{M}\_\mathrm{T}_2 \sqsubseteq \mathit{partOf}\, T_1$ and $M\_T_2\ \mathit{partOf}\, T_1$, e.g., "device unique id" is a part of "device," and "unique id" is a part of "device."

M3. T_P implies $T\_P \sqsubseteq \mathit{partOf}\, T\ \mathit{and}\ T\_P \sqsubseteq P$, e.g., "device name" is a part of "device" and a kind of "name."

M4. E_T implies that $E\_T \sqsubseteq \mathit{partOf}\, E\ \mathit{and}\ E\_T \sqsubseteq T$, e.g., "advertising identifier" is part of "advertising" and a kind of "identifier."

M5. E_P implies $E\_P \sqsubseteq \mathit{partOf}\, E\ \mathit{and}\ E\_P \sqsubseteq P$, e.g., "click count" is part of "click" and a kind of "count."

M6. T_E_α implies that $T\_E\_\alpha \sqsubseteq \mathit{partOf}\, T\ \mathit{and} T\_E\_\alpha \sqsubseteq (T\_\alpha \sqcup E\_\alpha)$, e.g., "language modeling data" is a part of "language" and a kind of "language data" and "modeling data."

M7. $M_1\_T_1\_M_2\_T_2$ implies $M_1\_T_1\_M_2\_T_2 \sqsubseteq \mathit{partOf}\, M_1\_T_1$ *and* $M_1\_T_1\_M_2\_T_2 \sqsubseteq M_2\_T_2$, e.g., "mobile device unique identifier" is a part of "mobile device" and a kind of "unique identifier."

M8. $T_1\_E\_T_2$ implies that $T_1\_E\_T_2 \sqsubseteq \mathit{partOf}\, T_1\_E$ *and* $T_1\_E\_T_2 \sqsubseteq (E\_T_2 \sqcup T_1\_\mathit{information} \sqcup T_2\_\mathit{information})$, e.g., "Internet browsing behavior" is a part of "Internet browsing" and a kind of "browsing behavior" and "Internet information" and "behavior information."

M9. T_E_P implies that $T\_E\_P \sqsubseteq \mathit{partOf}\, T\_E$ and $T\_E\_P \sqsubseteq (E\_P \sqcup \mathrm{T}\_\alpha \sqcup P)$, e.g., "website activity date" is a part of "website activity" and a kind of "activity date," "website information," and "date."

Synonymy Rules

S1. T implies $T \equiv T\_\alpha$, e.g., "device" is a synonym of "device information."

S2. P implies $P \equiv P\_\alpha$, e.g., "name" is a synonym of "name information."

S3. E implies $\equiv (E\_\alpha \sqcup E\_lemma)$, e.g., "views" is a synonym of "views information" and "view."

S4. G implies $G \equiv G\_\alpha$, e.g., "user" is a synonym of "user information."

The automated step 7 applies the rules to phrases and yields variant relationships for evaluation in two steps: (a) the semantic rules are matched to the typed phrases to infer new candidate phrases and relations; and (b) for each inferred phrase, we repeat step (a) with the inferred phrase. The technique terminates when no rules match a given input phrase. An inferred phrase can be either *explicit concept name,* which refers to an inferred phrase that exists in the lexicon, or *tacit concept nam*e referring to an inferred phrase that does not exist in the lexicon.

For example, in Fig. 3, we perform step (a) by applying the rule H5 to infer that "mobile device IP address" is a kind of "mobile information" and a part of "mobile device IP" and "device IP address" is a part of "mobile device IP." Rule H5 has the implication that $M\_T\_P \sqsubseteq M\_\alpha$, which yields an information class for M_α that includes information about things distinguished by a modifier M. In practice, these classes describe all things personal, financial, and health-related, and, in this example, all things mobile. Continuing with the example, the phrases "device IP address" and "mobile device IP" are not in the lexicon, i.e., they are potentially implied or *tacit concept names*. Thus, we re-apply the rules to "device IP address" and "mobile device IP." Rule M3 matches the "device IP address" typing to infer that "device IP address" is part of "device IP" and is a kind of "address." Since "device IP" is not in the lexicon, we re-apply the rules to this phrase. Rule M1 matches the type sequence of this phrase to yield "device IP" is a part of "device" and "device IP" is a kind of "IP." Both "device" and "IP" are *explicit concept names*. Therefore, we accept both inferences for further evaluation. We continue performing step (a) on "mobile device IP" by applying rule H3 that infers additional concept names and relations. The axioms from re-applying the rules to the explicit and tacit concepts names yield ontology fragments. We evaluate these axioms using the individual preference relationships described in the next section.

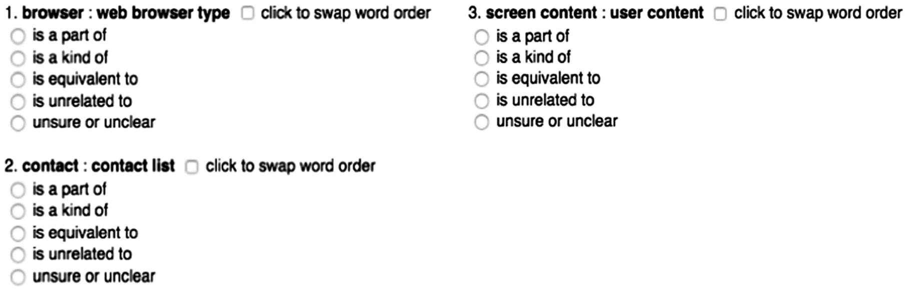
6 Experiment Setup

In psychology, preferences reflect an individual's attitude toward one or more objects, including a comparison among objects [24]. We designed a survey to evaluate and improve the ontological relationship prospects produced by step 7. We used 50 privacy policies and 335 pre-processed unique information types in a training set to improve the semantic rules. Because the prospects produced by the semantic rules all share at least one common word, we asked 30 human subjects to compare each 2,365 phrase-pair from the lexicon that shares at least one word. The survey asks subjects to classify each pair by choosing a relationship from among one of the following six options:

S: Phrase A is subsumed by phrase B in pair (A, B)

S: Phrase B is subsumed by phrase A in pair (A, B)
P: Phrase A is part of Phrase B in pair (A, B)
W: Phrase B is part of Phrase A in pair (A, B)
E: Phrase A is equivalent to phrase B in pair (A, B)
U: Phrase A is unrelated to phrase B in pair (A, B)

Figure 4 presents a survey excerpt: the participant checks one option to indicate the relationship, and they can check a box to swap the word order, e.g., in the first pair, the subject can check the box to indicate that "web browser type" is a part of "browser." We recruited 30 participants to compare each pair using Amazon Mechanical Turk, in which three pairs were shown in one Human Intelligence Task (HIT). Qualified participants completed over 5,000 HITs, had an approval rate of at least 97%, and were located in the United States. The average time for participants to compare a pair is 11.72 s.



1. **browser** : **web browser type** □ click to swap word order
○ is a part of
○ is a kind of
○ is equivalent to
○ is unrelated to
○ unsure or unclear

2. **contact** : **contact list** □ click to swap word order
○ is a part of
○ is a kind of
○ is equivalent to
○ is unrelated to
○ unsure or unclear

3. **screen content** : **user content** □ click to swap word order
○ is a part of
○ is a kind of
○ is equivalent to
○ is unrelated to
○ unsure or unclear

Fig. 4. Example survey questions to collect relation preferences

The participant results are analyzed to construct a ground truth (GT) in Description Logic. In the results, participants can classify the same phrase pair using different ontological relations. There are several reasons that explain multiple ontological relations for each pair: participants may misunderstand the phrases, or they may have different experiences that allow them to perceive different interpretations (e.g., "mac" can refer to both a MAC address for Ethernet-based routing, and a kind of computer sold by Apple, a manufacturer). To avoid excluding valid interpretations, we built a multi-viewpoint GT that accepts multiple, competing interpretations. For the entire survey results, we define *valid interpretations* for a phrase pair to be those interpretations where the observed number of responses per category exceeds the expected number of responses in a Chi-square test, where $p < 0.05$, which means there is at least a 95% chance that the elicited response counts are different than the expected counts. The expected response counts for an ontological relationship are based on how frequently participants chose that relationship across all comparisons. We constructed a multi-viewpoint GT as follows: for each surveyed pair, we add an axiom to GT for the relation category, if the number of participant responses is greater than or equal to the expected Chi-square frequency; except, if the number of unrelated responses exceeds the expected Chi-square frequency, then we do not add any axioms. We published the ground truth dataset[4] that

[4] http://gaius.isri.cmu.edu/dataset/plat17/preferences.csv.

includes phrase pairs, the ontological relation frequencies assigned by participants to each pair, and the Chi-square expected values for each relation per pair.

We measure the number of true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs) by comparing the variant relationships with the ground truth ontology to compute precision = TP/(TP + FP) and recall = TP/(TP + FN). A variant relation is a TP, if it is logically entailed by GT, otherwise, that relationship is a FP. An unrelated phrase pair in the preferences results is considered as TN, if we cannot match any inferred variant relationship with it. For all phrase pairs with valid interpretations (hypernymy, meronymy, synonymy) that do not match an inferred variant relationship, we count these as FN. We use logical entailment to identify true positives, because subsumption is transitive and whether a concept is a hypernym to another concept may rely on the transitive closure of that concept's class relationships. Next, we present results from improving the semantic rules using the training dataset and describe our approach for building the test set to evaluate the final rule set.

7 Evaluation and Results

This section presents the results for the training and testing of the approach. The training has been done in two incremental phases: (1) we first evaluated a set of 17 initial rules applied to the 335 pre-processed unique information types; (2) based on the results of phase 1 and analysis of false negatives, we extended the initial rules to 26 rules and evaluated the application of the extended rule set using the 335 pre-processed unique information types. In the testing stage, we utilized a separate 109 pre-processed unique information types to evaluate the extended rule set.

7.1 Preference Relations with Initial Rule Set

We began with a set of 17 rules that summarized our intuition on 335 pre-processed unique information types for variant relationship inference. After typing and decomposition, the technique yields 126 *explicit concept names* from the original lexicon, 182 potential *tacit concept names*, and 1,355 total axioms. Comparing the inferred relations with the individuals' preferences in the training ground truth (GT) results in 0.984 precision and 0.221 recall. Overall, the method correctly identifies 256/1,134 of related phrase pairs in the training GT. The total number of true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs) are 256, 1092, 4, and 901, respectively. To improve the results, we analyzed the FNs and extended the initial 17 rules to 26 total rules that are discussed in Sect. 5.3. Next, we report the results from applying the extended rules to the original 335 pre-processed unique information types.

7.2 Preference Relations with Extended Rule Set

The extended rule set consists of the initial and nine additional rules to improve the semi-automated technique. We also extended rules H3 and H5 with a new meronymy-inferred relationship as defined in Sect. 5.3. Using the extended rule set, the technique yields 186 *explicit concept names*, 286 potential *tacit concept names*, and 2,698 total axioms. The ontology fragments computed by applying the extended rule set can be found online in the OWL format.[5] Table 1 shows results for the semi-automated method with the initial and extended rule sets. This table also includes the number of hypernymy, meronymy, and synonymy relations that are inferred using the two rule sets. The extended rule set correctly identifies 782 preference relations out of 1,134 related pairs in the training GT. Also, the recall is improved to 0.569 with the extended rule set.

Table 1. Evaluations of relations using initial and extended rule set on training GT

	Initial rules	Extended rules
Explicit/tacit concept names	126/182	194/289
Number of inferred hypernyms	580	1,122
Number of inferred meronyms	192	535
Number of inferred synonyms	583	1041
Precision	0.984	0.996
Recall	0.221	0.569

The total number of TPs, TN, FPs, and FNs are 782, 878, 3, and 590, respectively. We observed that 477/590 of false negatives (FNs) depend on semantics beyond the scope of the 6-role typology. For example, the training GT shows the participants agreed that “mobile phone” is a kind of “mobile device,” possibly because they understood that “phone” is a kind of “device.” We observed that 22/477 of semantically related FNs exclusively concern synonyms that require additional domain knowledge, e.g., “postal code” is equivalent to “zip code,” or in the case of acronyms, “Internet protocol address” is equivalent to “IP address.” Moreover, 10/477 of semantically related FNs exclusively concern meronymy, e.g., “game activity time” is a part of “game system.” Only 1/477 of semantically related FNs is exclusively mentioned for hypernymy: “forwarding number” is a kind of “valid mobile number.” Finally, 444/477 of semantically related FNs can have multiple valid interpretations (meronymy, hypernymy, and synonymy) in the training GT.

In addition, we discovered that 53/590 of FNs were due to individual preference-errors that were inconsistent with the automated method, e.g., individual preferences identified “mobile device identifier” equivalent to “mobile device unique identifier,” which ignores the fact that an identifier is not necessarily unique. Finally, we identified 60/590 relations that can be identified by introducing new semantic rules.

The training GT also contains a special relationship identified by individuals between 40 pairs that we call *part-of-hypernymy*. For example, individuals identified “device id” as a part of “mobile device,” because they may have assumed that mobile device (as a hyponym of device) has an id. Therefore, we extended rules H3 and H5 to infer *part-of-hypernymy* in the extended rule set.

[5] http://gaius.isri.cmu.edu/dataset/plat17/variants.owl.

7.3 Method Evaluation

To evaluate our extended rule set, we randomly selected six additional privacy policies from the pool of 501 policies discussed in Sect. 5.1. We used the same approach and annotators from Sect. 5.1 to extract the unique information types and construct the test lexicon. The resulting 110 information types were reduced to 109 information types which were then typed and analyzed by the extended rule set, resulting in 76 *explicit concept names*, 139 potential *tacit concept names*, and 831 total axioms. We acquired the preference relations[6] for the test lexicon by surveying 213 phrase pairs resulting in 121 related phrase pairs included in the testing ground truth (GT) using the method discussed in Sect. 6. In further analysis, the relations in the testing GT were compared with the relations provided by the extended rule set. Overall, the extended rule set correctly identifies 79 preference relations out of 121 related pairs in the training GT. Table 2 presents the results including the precision and recall for this analysis. The ontology fragments computed using the extended rule set are online in OWL.[7]

Table 2. Evaluations of relations using extended rule set on testing GT

	Extended rules
Explicit/tacit concept names	194/289
Number of inferred hypernyms	385
Number of inferred meronyms	80
Number of inferred synonyms	366
Precision	1.000
Recall	0.593

In summary, the results show total number of 79 TPs, 80 TNs, zero FPs, and 54 FNs. We observed that 44/54 of FNs in the test set depend on semantics beyond the scope of the role typology and syntactic analysis of information types. We published a list of these concept pairs, including the human preferences.[8] Some examples include: “device open udid” as a kind of “device identifier,” “in-app page view” as a kind of “web page visited,” and “page viewed” as equivalent to “page visited.” We also observed 7/54 of FNs that require introducing six new rules. Finally, by comparing the total number of TPs and TNs with 213 phrase pairs, we can conclude that the semi-automated semantic analysis method can infer $\left(\frac{79+80}{213} \times 100\right) = 74\%$ of paired comparisons.

8 Discussion

We now discuss and interpret our results and threats to validity.

[6] http://gaius.isri.cmu.edu/dataset/plat17/study-utsa-prefs-test-set.csv.
[7] http://gaius.isri.cmu.edu/dataset/plat17/variants-test-set.owl.
[8] http://gaius.isri.cmu.edu/dataset/plat17/supplements-test-set.csv.

8.1 Interpretation of Extended Rule Set Results

Comparing the ontology fragments to preferences, we observe that preferences imply new axioms that explain a portion of the FNs in training and testing. These preferences are influenced by individual interpretations of relations between two phrases. Analyzing these FNs, we identified four cases where individuals report incorrect interpretations:

(1) The meaning of modifiers in a phrase are ignored and an equivalent relationship is identified for a pair of phrases, e.g., “unique id” and “id.”
(2) Different modifiers are interpreted as equivalent, e.g., “approximate location information” and “general location information.”
(3) The superordinate and subordinate phrase’s relationship is diminished and an equivalent relation is assumed, e.g., “hardware” and “device”, “iPhone” and “device.”
(4) Information as a whole that contains information is confused with information as a sub-ordinate concept in a super-ordinate category, e.g., “mobile application version” is both a part of, and a kind of, “mobile device information.”

One explanation for the inconsistencies is that individuals conflate interpretations when comparing two phrases as a function of convenience. Without prompting individuals to search their memory for distinctions among category members (e.g., iPhone is different from Android, and both are kinds of device), they are inclined to ignore these distinctions when making sense of the comparison. In requirements engineering, this behavior corresponds to relaxing the interpretation of constraints or seeking a narrower interpretation than what the natural language statement implies. When relaxing constraints, stakeholders may overlook requirements: e.g., if “actual location” and “physical location” are perceived as equivalent, then stakeholders may overlook requirements that serve to more closely approximate the “actual” from noisy location data, or requirements to acquire location from environmental cues to more closely approximate a “physical” location. Furthermore, this behavior could yield incomplete requirements, if analysts overlook other, unstated category members.

8.2 Threats to Validity

In this section, we discuss the internal and external validity for our approach.

Internal Validity. Internal validity is the extent to which observed causal relations actually exist within the data, and whether the investigator’s inferences about the data are valid [25]. In this method, the inferred semantic relations are highly dependent on the role typing system and any inconsistencies in the types affect the final results. For this reason, two analysts assigned roles to the phrases in the training lexicon. We used Fliess’ Kappa to measure the degree of agreement for this task [26]. Two analysts reached Kappa of 0.72, which shows a high, above-chance agreement. However, there is still a need for automating the role typing system to reduce potential inconsistencies.

External Validity. External validity is the extent to which our approach generalizes to the population outside the sample used in the study [25]. Based on our study, 7/54 of false negatives in test set evaluation require six new semantic rules. Moreover, we cannot claim that the extended rule set will cover all the information types extracted from privacy policies, since we only analyzed specific information types called platform information. To assure that the rules have saturated for information type analysis, further studies on different information types are required.

9 Conclusion and Future Work

Privacy policies contain legal requirements with which company information systems need to comply. In addition, they serve to communicate those requirements to other stakeholders, such as consumers and regulators. Because stakeholders use different words to describe the same domain concept, how these policies use abstraction and variability in concept representation can affect ambiguity and reduce the shared understanding among policy authors, app developers, regulators and consumers. To address this problem, we present results of a semi-automated, semantic analysis method to construct privacy policy ontologies that formalize different interpretations of related concepts.

The method was evaluated on 213 pairs of phrases that share at least one word from a set of 109 unique phrases in the lexicon acquired from six mobile app privacy policies. The individual preference data set contains 80/213 pairs that are identified as unrelated (37%) and 121/213 relations identified as related through hypernymy, meronymy, and synonymy in the testing GT. The technique yields 79/121 of axioms in testing GT with an average precision = 1.00 and recall = 0.59.

In future work, we envision a number of extensions. To increase coverage, we propose to formalize the rules as a context free grammar with semantic attachments using the rule-to-rule hypothesis [27]. We also envision expanding the knowledge base to include relations that cannot be identified using syntactic analysis, such as hypernymy between "phone" and "device." To improve typing, we considered identifying role types associated with part-of-speech (POS) tagging and English suffixes. However, preliminary results on 335 pre-processed phrases from the training lexicon shows only 22% of role type sequences can be identified using POS and English suffixes. Therefore, instead of relying on POS and suffix features, we envision using deep learning methods [28] to learn the features for identifying the semantic relations between phrases. Finally, we envision incorporating these results in requirements analysis tools to help detect and remediate variants that can increase ambiguity and misunderstanding.

Acknowledgement. We thank Jaspreet Bhatia, Rocky Slavin, and Xiaoyin Wang for their annotations of the 50 mobile app policies, and the CMU RE Lab for their helpful feedback. A short version of this paper, which compared a manually-constructed ontology to the initial rule set, was presented at the AAAI Fall Symposium on Privacy and Language Technologies and appears online as a non-archival technical report. This research was supported by NSF CAREER #1453139, NSA #141333, NSF #1330596, and NSF #0964710.

References

1. Smith, A.: US smartphone use in 2015. Pew Research Center, 1 (2015)
2. Harris, K.D.: Privacy on the go: recommendations for the mobile ecosystem (2013)
3. Anton, A.I., Earp, J.B.: A requirements taxonomy for reducing web site privacy vulnerabilities. Requir. Eng. **9**(3), 169–185 (2004)
4. Uschold, M., Gruninger, M.: Ontologies: principles, methods and applications. Knowl. Eng. Rev. **11**(02), 93–136 (1996)
5. Breaux, T.D., Baumer, D.L.: Legally "reasonable" security requirements: a 10-year FTC retrospective. Comput. Secur. **30**(4), 178–193 (2011)
6. Bhatia, J., Breaux, T.D.: Towards an information type lexicon for privacy policies. In: 2015 IEEE Eighth International Workshop on Requirements Engineering and Law (RELAW), pp. 19–24. IEEE (2015)
7. Hosseini, M.B., Wadkar, S., Breaux, T.D., Niu, J.: Lexical similarity of information type hypernyms, meronyms and synonyms in privacy policies. In: 2016 AAAI Fall Symposium Series (2016)
8. Martin, J.H., Jurafsky, D.: Speech and language processing. Int. Ed. **710**, 117–119 (2000)
9. Potts, C., Newstetter, W.C.: Naturalistic inquiry and requirements engineering: reconciling their theoretical foundations. In: 1997 Proceedings of the Third IEEE International Symposium on Requirements Engineering, pp. 118–127. IEEE (1997)
10. Hecker, M., Dillon, T.S., Chang, E.: Privacy ontology support for e-commerce. IEEE Internet Comput. **12**(2), 54–61 (2008)
11. Bradshaw, J., Uszok, A., Jeffers, R., Suri, N., Hayes, P., Burstein, M., Acquisti, A., Benyo, B., Breedy, M., Carvalho, M., Diller, D.: Representation and reasoning for DAML-based policy and domain services in KAoS and Nomads. In: Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 835–842. ACM (2003)
12. Kagal, L., et al.: Authorization and privacy for semantic web services. IEEE Intell. Syst. **19**(4), 50–56 (2004)
13. Syed, Z., Padia, A., Finin, T., Mathews, M.L., Joshi, A.: UCO: a unified cybersecurity ontology. In: AAAI Workshop: Artificial Intelligence for Cyber Security (2016)
14. Breaux, T.D., Smullen, D., Hibshi, H.: Detecting repurposing and over-collection in multi-party privacy requirements specifications. In: 2015 IEEE 23rd International Requirements Engineering Conference (RE), pp. 166–175. IEEE (2015)
15. Slavin, R., Wang, X., Hosseini, M.B., Hester, J., Krishnan, R., Bhatia, J., Breaux, T.D., Niu, J.: Toward a framework for detecting privacy policy violations in android application code. In: Proceedings of the 38th International Conference on Software Engineering, pp. 25–36. ACM (2016)
16. Huang, C.R. (ed.): Ontology and the Lexicon: A Natural Language Processing Perspective. Cambridge University Press, Cambridge (2010)
17. Breaux, T.D., Schaub, F.: Scaling requirements extraction to the crowd: experiments with privacy policies. In: 2014 IEEE 22nd International Requirements Engineering Conference (RE), pp. 163–172. IEEE (2014)
18. Wilson, S., Schaub, F., Dara, A.A., Liu, F., Cherivirala, S., Leon, P.G., Andersen, M.S., Zimmeck, S., Sathyendra, K.M., Russell, N.C., Norton, T.B.: The creation and analysis of a website privacy policy corpus. In: ACL, vol. 1 (2016)
19. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)
20. Snow, R., Jurafsky, D., Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. In: Advances in Neural Information Processing Systems, vol. 17 (2004)

21. Bhatia, J., Evans, M.C., Wadkar, S., Breaux, T.D.: Automated extraction of regulated information types using hyponymy relations. In: IEEE International Requirements Engineering Conference Workshops (REW), pp. 19–25. IEEE (2016)
22. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th Conference on Computational Linguistics, vol. 2, pp. 539–545. Association for Computational Linguistics (1992)
23. Saldaña, J.: The Coding Manual for Qualitative Researchers. Sage, London (2015)
24. Lichtenstein, S., Slovic, P. (eds.): The Construction of Preference. Cambridge University Press, Cambridge (2006)
25. Yin, R.K.: Case Study Research: Design and Methods. Sage publications, Thousand oaks (2009)
26. Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychol. Bull. **76**(5), 378 (1971)
27. Bach, E.: An extension of classical transformational grammar (1976)
28. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: COLING, pp. 2335–2344 (2014)