Helping Users Understand Privacy Notices with Automated Query Answering Functionality: An Exploratory Study

Kanthashree Mysore Sathyendra, Abhilasha Ravichander,
Peter Garth Story, Alan W. Black, Norman Sadeh

CMU-ISR-17-114R

Also available as Language Technologies Institute Technical Report CMU-LTI-17-005

December 2017

School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

This work has been supported by the National Science Foundation as part of the Usable Privacy Policy Project (www.usableprivacy.org) under Grant No. CNS 13-30596. The US Government is authorized to reproduce and distribute reprints for Governmental purposes not withstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the National Science Foundation or the US Government.

Abstract

Privacy notices are the default mechanism used to inform users about the data collection and use practices of technologies (e.g., websites, mobile apps, Internet of Things devices) and processes with which they interact. The length of these policies and their often convoluted language have been shown to discourage most users from reading them. Recent progress in natural language processing and machine learning has opened the door to the development of technologies that are capable of automatically extracting statements (or "annotations") from the text of privacy policies. These technologies could help users quickly identify those elements of a privacy notice they care about - without requiring them to read the full text of the notice.

In this article, we review the requirements associated with the development of Query Answering functionality that would enable users to ask questions about specific aspects of privacy notices (e.g. Does this app share my location with third parties? Am I able to review the information this website collects about me? Can I delete my account? For how long is my information going to be retained by this company?). We discuss different possible approaches to supporting such functionality and how they relate to recent advances in automatically annotating privacy notices. Initial results obtained with different machine learning/natural language processing techniques are presented, suggesting that Query Answering functionality could be a particularly promising approach to informing users about privacy practices. In particular, in contrast to automated annotation techniques that aim to extract detailed statements from the text of privacy notices, Query Answering functionality could be configured to return short text fragments extracted from privacy notices and rely on the user (rather than the computer) to interpret some of the finer nuances of the text found in these fragments. Such an approach could potentially prove more robust than fully automated annotation techniques, which at least at this time struggle with the interpretation of finer nuances.

This article also includes a brief discussion of opportunities and challenges associated with possible extensions of Query Answering functionality in the form of privacy assistants capable of entertaining dialogues with users to clarify some of their questions and help them understand to what extent their concerns are explicitly addressed (or not) by the text of privacy notices. Such functionality could provide for yet greater robustness and usability than fully automated annotation techniques, and could eventually also leverage models of what the user already knows and/or cares about.

Contents

1 Introduction

Privacy notices (aka "privacy policies") are intended to inform people about the data collection and use practices of technologies and processes with which they interact. In practice these notices often come in the form of long and complicated documents. It has been shown that average Internet users would in fact require an impractical amount of time to read the privacy policies of all the online services they access [24]. Although people are generally concerned about their privacy and would like to understand how their data is collected and used, they are simply not willing to spend the time that would be necessary to read the text of all these privacy notices. What is needed is technology that could help them quickly zoom in on those issues they care about. Ideally, this would come in the form of funtionality capable of answering questions the user has about the collection and use of his or her data (e.g., what data is collected, for what purpose, for how long it will be retained, whether it will be shared with third parties and more). Because privacy notices are often incomplete and ambiguous, ideally such question answering functionality should also be able to tell users that a privacy notice is silent or unclear about some of their questions.

Over the past several years, research combining crowdsourcing, machine learning and natural processing has shown that is possible to semi-automatically extract a variety of annotations from privacy policies, some more successfully than others [44, 43, 37, 30]. These efforts show the promise of machine learning and natural language processing when it comes to helping users make sense of long privacy policies. At the same time, they continue to be imperfect, as these techniques are not fully accurate and as certain types of annotations are proving more difficult to automatically extract than others. An alternative is to rely on the ability of users to better interpret some of the finer nuances found in the language of privacy policies, and to focus on just extracting concise text fragments that are likely to contain the answer(s) a user is looking for. This approach is the focus of the present report. With the emergence of voice assistants and voice-enabled devices such as Amazon Alexa, recent advances in Natural Language Processing (NLP) and the emergence of privacy assistant technology [18], exploring such an approach seems particularly timely. The techniques presented in this technical report can easily be deployed on platforms such as Alexa or Google Home, for instance.

The ultimate objective of the research presented in this report is the development of a privacy assistant capable of engaging in dialogues with users to help them understand the data collection and use practices disclosed in privacy notices. Figure 1 outlines possible components of such an assistant. They would likely include policy retrieval functionality to determine and retrieve the one or more privacy notices corresponding to a user question, a module to identify relevant text fragments in these privacy notices, a dialogue manager responsible for determining how to most effectively present the resulting text to the user (e.g., presenting all the text, or possibly engaging in a dialogue with the user to more specifically identify the parts of the text the user is likely interested in). The dialogue manager could also be responsible for helping clarify elements of the user's question upfront. Ultimately, one would want to develop privacy assistants capable of answering a wide variety of more or less articulate questions from users. This would range from simple questions about the data

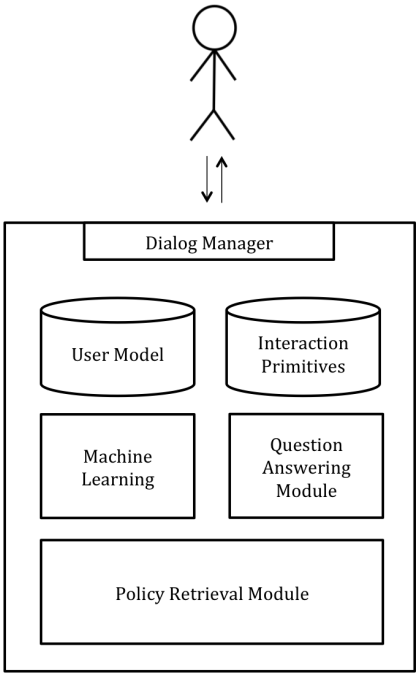

Figure 1: Schematic Diagram of Privacy Assistant

a particular technology (e.g. website or mobile app) is collecting to more complex questions requiring the analysis and/or comparison of multiple privacy notices (e.g., "Can you identify a mobile app that does not share my location with advertisers?"). It might eventually also include the ability to answer more open-ended questions such as "Are there practices that this website is particularly ambiguous about," or even "Is there anything this website does that would surprise me?" Answering such questions would require building models of people's privacy expectations and what they already know. The initial work presented in this report is significantly less ambitious and focuses specifically on answering user questions that pertain to one particular privacy notice, namely the functionality identified as "question answering" in Figure 1.

In this preliminary study, we distinguish between two broad approaches to answering questions: "Closed Question Answering" and "Open Question Answering". Under the "Closed Question Answering" model, we assume that user questions can generally be mapped onto a predefined taxonomy of questions and rely on automated annotation techniques to extract answers to the one or more predefined questions that best match the user's query. Under the "Open Question Answering" approach, we make no such assumption and instead rely on information retrieval techniques and neural network models to identify text fragments that seem to best match the user's query. As we explore these two approaches, we consider different sets of techniques and evaluate them on different corpora of privacy notices and user queries.

The remainder of this technical report is organized as follows. Section 2 provides a discussion of related work in both privacy and NLP/ML. Section 3 provides

an overview of the datasets used in this study Section 4 introduces the closed and open Query Answering models considered in this study. Specific techniques developed for each of these models are presented along with results obtained with these techniques. Concluding remarks, including a brief discussion of future work, are presented in Section 8.

2 Related Work

Privacy policies, being long, complicated documents full of legal jargon, are sub-optimal for communicating information to individuals [7, 6, 38, 24, 36]. As described in [24], they are 'hard to read, read infrequently, and do not support rational decision making'. There has been a wealth of research on techniques to make these policies more accessible and interpretable for consumers. Vail et al. study different ways in which information from privacy policies can be presented to consumers, and discuss some of the tradeoffs associated with these different types of presentation [42]. The Platform for Privacy Preferences (P3P) introduced browser agents that can automatically check whether a given privacy policy aligns with a user's specified privacy preferences [8]. Kelley et al. proposed a 'nutrition label' approach, where they explore presenting relevant sections of each privacy policy to a user in a standardized format and find that this approach can have a positive influence on the user's experience and can help motivate users to pay closer attention to privacy policies. Micheti et al. aim to develop *guidelines* to follow while constructing privacy policies with the end goal of making them simpler to understand; they analyze what factors make policies accessible to teenagers and children in order to create the guidelines [25]. These and related efforts can inform the composition and presentation of privacy policies. Yet adoption has been fairly slow and, as already mentioned, policies generally remain long and difficult for users to read and understand. By developing question answering functionality based on the text of privacy policies and allowing users to directly submit questions of interest to them, our objective is to increase the accessibility and usefulness of these policies.

The potential for the application of NLP and information retrieval techniques to legal documents has been studies by a number of researchers [22], with multiple efforts applying NLP techniques to legal documents. [3] uses multi-layer sequence learning model and integer linear programming to learn logical structures of paragraphs in legal articles. [13] presents a hybrid approach to summarization of legal documents, based on creating rules to combine different types of statistical information about text. [29] investigates the peculiarities of the language in legal text with respect to that in ordinary text by applying shallow parsing. [10] utilizes WordNet and chunk-based dependency parsing to extract rules from legal texts. [19] modelled the language of vagueness in privacy policies using deep neural networks.

Over the past several years, there has been extensive research on using Natural Language Processing to understand the content of privacy policies as part of the Usable Privacy Policy Project [37]. [30] extracts user choices in privacy policies, focusing in particular on opt-out choices.[34] introduces an unsupervised model for the automatic alignment of privacy policies and shows that Hidden Markov Models are more

effective than clustering and topic models. [20] uses topic models to show meaningful mappings of text segments onto a collection of ten categories of data collection and use practices identified together with privacy practitioners and privacy law experts. [45] automatically analyzes the privacy policies of mobile apps to check for discrepancies between an app's privacy policy and the code of the app. [11] examines the automatic creation of an information-type ontology in privacy policies. [15] desribes a set of heurestics that can be used to construct an ontology taking into account hypernymy, meronymy and synonymy relations. [44] describes the creation of a dataset of privacy policies from websites and an annotation scheme desribing the data collection and use practices of the different sections in a privacy policy.

Previous attempts have been made to build question answering systems for legal documents (e.g., [28, 33]). These approaches are based on information retrieval for legal documents and have primarily been applied to juridical documents.[16] explores answering true or false questions from Japanese bar exams using Convolutional Neural Networks, by first identifying relevant articles and training an entailment model to predict whether the article entails the question. [21] attempts to find relevant Taiwanese legal statutes for a natural language query. [5] uses an n-gram language model with several lexical and morphological features to answer yes/no questions. [23] describes a Hidden Markov Model (HMM) approach to retrieving relevant legal documents. [9] utilizes a Ranking Support Vector Machine (SVM) model and a Deep Convolutional Neural network to answer legal questions, by retrieving legal articles and ranking them in order of relevance to the question. A number of authors have also described domain-specific knowledge engineering approaches combining ontologies and knowledge bases to answer questions (e.g., [27, 12]).

Despite the above efforts, there has been very limited work on answering questions based on the text of privacy policies. The work presented herein is closest to that of Harkous et al. who describe conversational privacy bots (PriBots) that build on machine learning techniques to automatically annotate the text of privacy policies [14]. Some of the datasets used in this research are the same as those used by Harkuous et al., with some datasets originating from their group and the OPP1-115 corpus originating from our group. Some of the techniques presented herein are also similar. In contrast to the work of Harkous et al, the work presented herein explores both closed domain and open domain approaches and, in the latter case, explores different ranking techniques to determine which text segments to include in answers.

Other related research also includes that of Ammar et al who use automatic text categorization to answer some simple questions about privacy policies [2]. They utilize logistic regression to predict the presence or absence of a limited set of practice statements within a privacy policy. As already indicated, this work was later extended in the Usable Privacy Policy Project (e.g.,[37, 30, 20]). Oltramari et al. [31] introduced an ontology of data collection and use practices that allows one to submit SparQL queries against a corpus of annotated privacy policies. The annotations used in this work were crowdsourced. The same ontology could also be used to interpret automatically annotated privacy policies.

In this report, we describe preliminary work aimed at exploring the application of Question Answering (QA) and Natural Language Processing (NLP) techniques to the problem of answering user privacy questions based on the text of available privacy

policies. Because of the lack of an available question answering dataset, we focus on semi-supervised and unsupervised techniques. We also look at techniques to address the problem of open QA where the question space is not restricted to a predefined set of questions.

3 Datasets

There are few datasets of privacy policies, which limited our options for training our question answering systems. The datasets we used were the following:

- OPP-115 Corpus, namely a corpus of 115 privacy policies with manual annotations associated with a number of different data collection and use practices [44]
- 35,000 privacy policies shared with us by researchers working on the PriBot project [14]
- Dataset of privacy related questions submitted via Twitter, also shared with us by the PriBot project [14]
- 6,000 privacy policies crawled from the web

The OPP-115 Corpus consists of 115 website privacy policies and associated annotations of *data practices*. A data practice is a statement about the website operator's data handling practices. Each data practice consists of a selection of a a data practice category (e.g., is the data practice corresponding to data collected or used by the website operator, namely "First Party Collection/Use," does it pertain to a "User Choice/Control," or does it pertains to "Data Retention"), a set of values for attributes specific to the category (e.g. the particular type of data being collected by the first party, or the length of time over which the data will be retained) , and text spans from the policy associated with the practice and the value selections [44].

The second dataset, consisting of 35,000 privacy policies which were crawled from the web by researchers working on the PriBot project [14]. These websites were raw unannotated HTML files which were further segmented.

The third dataset consisted of privacy related questions scraped from Twitter, and was also provided by the researchers working on the PriBot project [14]. The dataset's questions were collected by searching for tweets to which an organization replied with a reference to their privacy policy (see Annex 6.1 for sample questions in this dataset).

The last dataset of 6,000 privacy policies was obtained by crawling top Alexa-ranked websites[1] and automatically retrieveing the text of their privacy policies. These were also raw unannotated HTML files which we subsequently segmented.

Note that of the four datasets, only the Twitter dataset contains questions. This dataset also includes the privacy policies corresponding to the websites referred to in the questions. The dataset however does not include answers to the questions. Also, it should be clear that the text of the policy identified to answer each question may not

[1] `http://www.alexa.com/topsites`

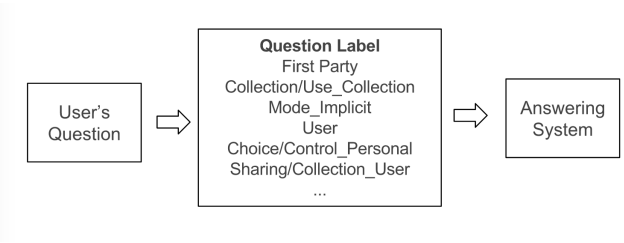

Figure 2: Schematic Diagram of Closed Domain Systems

contain an answer to that question. To overcome the lack of question-answer pairs in our four datasets, we will introduce techniques we used to synthesize question-answer pairs for each of these datasets. In Section 5, we discuss the value of creating a dataset specifically designed for the purpose of training privacy question answering systems.

4 Proposed Approaches

In this project we looked at Closed QA and Open QA approaches. These techniques can be deployed on platforms such as Alexa, Google Home etc., which also support dialog based conversations. Given a privacy related question, the system refers to the particular privacy policy and presents the closest segment from the privacy policy that best answers the users question. In this report, we discuss preliminary work using the following approaches:

- **Closed QA System** The main assumption here is that users' questions can always be mapped onto one or more of the preexisting categories of questions or question labels. In other words, this approach can only handle a limited set of questions.

- **Open QA System** Under this approach, the user can ask any question and the system tries to retrieve an answer that best matches the language of the question. We experiment with both neural and non-neural approaches, as well as expore the utility of domain-specific word embeddings in this task.

4.1 Closed QA Systems

In the Closed QA systems, each question asked by the user is mapped onto an intermediate question label. These question labels are determined based on the annotation scheme of the OPP-115 Corpus. The main assumption here is that, all user questions broadly fall into one of the categories of the annotation scheme of the OPP-115 Corpus[44]. Furthermore, if the user's question can not be mapped onto any of the question labels with sufficiently high confidence, the system could respond in two ways. The system could either continue the conversation by asking clarifying questions so that it is easier to map the user's question to a preexisting question label, or respond with a generic answer saying the system is unable to identify an answer to the question.

In this tech report, we have investigated this approach with the assumption that the user's question falls into one of the preexisting question labels.

Our Closed QA system has two main components: the Question Mapping system and the Answering System. This is illustrated in Figure 2, where, in a first step, the user's question is mapped onto one of the nine data collection and use practices available in the OPP-115 annotation scheme, and, in a second step, the text of the paragraphs associated with these labels is provided to an answer system responsible for generating the actual answer.

In the OPP-115 annotation scheme, labels correspond to ⟨category, attribute, value⟩ triples, where attributes are category-specific, with some attributes being mandatory (i.e., a label will not be created unless all mandatory attributes can be identified) and others potentially being optional[44]. For instance, a User Choice/Control label (namely a label describing choices and/or control mechanisms potentially available to users) comes with four required attributes: Choice Type, Choice Scope, Personal Information Type, and Purpose. Each such attribute comes with its own set of possible values. Given the vagueness of many privacy policies, these values typically include an "unspecified" value - to indicate that the policy is silent on a particular issue. Such values can be very useful in helping answer user questions, as they enable one to say with confidence that the policy is silent on a given issue.

A triple associated with a User Choice/Control practice and Choice Type attribute could be of the form: ⟨ User Choice/Control, Choice Type, Opt Out Via Contacting Company ⟩ (i.e., a company or entity that allows users to opt out of some practice by directly contacting the company). Another triple regarding third party sharing and collection could come in the form: ⟨ Third Party Sharing/Collection, Third Party Entity, Unspecified ⟩ (i.e., an entity that indicates it may share some unspecified information with some unspecified third parties).

To achieve question mapping, we use pre-trained Glove/Google Word Embeddings of 300 dimensions. [26][32]. These word embeddings are trained on 100 billion words based on Google News and Wikipedia articles. These word embeddings are vectors trained to capture semantic similarity of words. Words with similar meanings have embeddings that are closer in the vector space. Google Word2Vec uses neural networks to train whereas Glove uses statistical methods and is based on co-occurrences of words in large bodies of text.

4.1.1 Bag of Word Cluster Representation

Approach

In this approach, each piece of text (questions/annotated text spans) is represented using a bag of word cluster representation. For this, we first clustered word embeddings into 300 clusters using k-means clustering algorithm. This ensures that semantically similar words fall in the same cluster. Then, each query is represented as a bag-of-word cluster, i.e, each word was replaced by its representative cluster number. A vector of the size equal to the number of clusters was used to represent the query. A similar representation was used to represent annotated text spans for each of the question labels in the OPP-115 Corpus. Then, cosine similarity was used as the similarity metric

Approach	Value Level	Attribute Level	Category Level
Bag of Word Clusters	1/21	4/21	12/21
Bag of Words	0/21	1/21	7/21

Table 1: Results for the Question Mapping System. The values in each cell indicate the number of correct question label assignments at each level of granularity.

to compute the similarity between question and question labels and the question was assigned the question label for which the similarity was highest.

Preliminary Results

We selected 21 questions (see 6.1 for the list of questions) based on questions asked by users on privacy from the Twitter dataset and manually mapped these questions to question labels. We evaluated the performance of the question mapping system for this set of questions using the manually mapped target question labels. The results are presented in Table 1. As can be seen in the table, the Bag of word cluster representation works better than the Bag of Words representation to assign question labels to questions.

4.1.2 Discussion

We computed the results for the question mapping system for both Bag of Word Clusters and Bag of Words methods at three different levels of granularity: Value Level (finest level), Attribute Level (intermediate level), and Category Level (coarsest level). We found that Bag of Word Clusters performs better than Bag of Words at all three levels of granularity. In addition, we found that the accuracies of both the approaches were higher at the coarsest level - the category level, followed by the attribute level and the value level. This aligns with our expectation that it is easier to map questions to the coarser levels than the finer levels.

The exercise of manually mapping questions to question labels also showed that many questions asked by users did not map to question labels. We present some examples of such questions which do not exactly map to question labels here:

> "Can you tell me about privacy?"
>
> "How do we know u won't sell our information?"
>
> "Is making a note in Momento, or adding a picture, part of 'submitting to the Service' in the Terms of Service?"

These examples seem to suggest that this problem arises in particular when dealing with rather vague or ill-formulated privacy questions.

Questions that did not map at the Value level sometimes mapped at either the attribute or the category level. This would largely increase the set of possible question labels, which in turn could lead to significantly longer answers that include many details users do not care about. This finding prompted us to explore Open QA systems where users are allowed to ask any free form questions and questions are not expected to necessarily fall within a predefined set of categories.

Another challenge associated with the closed QA model outlined above is that it also requires developing an Answering system. In situations where Question Labeling ends up returning a large number of text spans, the Answering system would have to either summarize the matching text or find some meaningful way of organizing the text to avoid overwhelming the user (e.g., organizing it in such a way that the user can interactively drill down different paths to obtain the answer he or she is looking for).

4.2 Open QA Systems

Open QA Systems allow users to ask any free form natural language question. These questions need not map onto a specific set of categories and use unsupervised techniques. In this section, we describe two models to perform open question-answering., We also discuss a methodology for constructing word embeddings specifically for the privacy domain which help us improve the performance of Open QA systems.

4.2.1 Priv2Vec - Word Vectors for the Privacy Domain

As part of this project, we trained word embeddings on large corpuses of unannotated privacy policy text. The language used in privacy policies is very different from that used in generic English corpuses. For example, the words/phrases such as 'information' or 'data collection' might have different meanings in the privacy domain as compared to the generic English language. Most publicly available word embeddings are trained on large corpuses of generic English language text, such as the Wikipedia Corpus or the Google News corpus. As such they may not be optimal for work in the privacy domain. This raises the question of whether training word embeddings specifically for the privacy domain could help improve performance. Further, easy access to a large number of online privacy policies enables training of word embeddings specifically identified for privacy-related text.

Training Word Vectors for Privacy

We used the unannotated datasets mentioned in Section 3, for a total of 41,000 privacy policies. These privacy policies were in the form of raw HTML files. To train the word vectors, we first preprocessed these HTML files to remove unwanted HTML content such as JavaScript content, navigation bars, advertisements, images etc. We further extracted just the text from these HTML files, removing punctuation symbols and non-letters using BeautifulSoup [2]. We further removed all privacy policies which were not written in English, and privacy policies with less than 400 characters. We further replaced words with a word frequency of less than 5 with the <unk>token representing

[2] https://www.crummy.com/software/BeautifulSoup/bs4/doc/

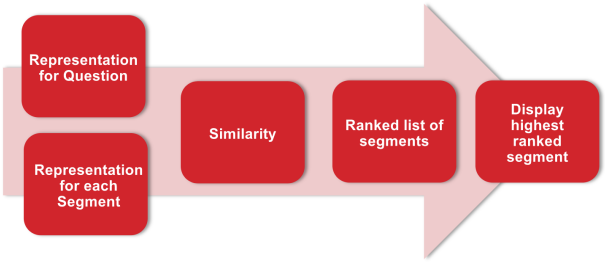

Figure 3: Schematic Diagram of IR based approach

unknown words. We were left with a set of 24,000 privacy policies which contained approximately 54 million words, with a vocabulary size of 103,708. We trained the word vectors using the Word2Vec skip-gram model [26] using Gensim[35]. We used these word embeddings in the approaches we present in later sections.

Although our experiments show that the word embeddings are indeed reasonable, the quality of these word embeddings could formally be evaluated using various methods. Some methods are presented in [39]. These evaluations provide a measure of how good the word embeddings are and could prove to be very useful in training domain specific word embeddings using unsupervised methods. It is also interesting to note that generic English word embeddings are trained on corpuses of very large sizes (in the order of billions of words). However, our experiments indicate that even with a relatively small corpus, the word embeddings trained for privacy seem to work well for the task of question answering, as discussed below.

4.2.2 BM25

Approach

The main idea here is to identify the segment in the text of the privacy policy that is most relevant to the user's question. The sequence of steps followed to determine the correct answer are as follows:

1. The privacy policy is first divided into segments. These segments are the answer candidates for the user's question. For the purpose of this project, we have used the segmentation technique introduced in the OPP-115 Corpus [44].

2. The user's query is then stemmed and expanded.

3. Segments from the privacy policy are ranked based on BM25 scores of the segments with the user's query.

4. The top ranked segment is presented as the answer. To handle very long segments, a segment length reduction strategy is used which is discussed below. This also handles the cases where the answer to the query is hidden in long segments of text.

	MRR	Avg #words	Avg #sentences
Query expansion	0.57	122.66	6.66
No Query expansion	0.21	46	2.66

Table 2: Results for IR based Approach, with and without Query Expansion

Stemming and stop word removal In traditional IR methods, stemming and stop word removal normalizes text. Stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form: generally a written word form. Words such as 'inform', 'information', or 'informing' are reduced to their root word, namely 'inform'. We also removed stop words such as 'a', 'an' etc. to prevent these words from influencing performance.

Query Expansion Each word in the user's query was first expanded to include synonymous terms and then stemmed using the NLTK stemmer [4]. To obtain synonymous terms, we used Priv2Vec word vectors. Each word in the query was expanded based on the 10 most similar words (obtained based on cosine similarity) based on the trained word embeddings.

Length Reduction Strategy Sometimes, the length of the answer segment is excessive. Furthermore, the actual answer could be hidden in the middle of the identified answer segment. To overcome this, we used a simple strategy. We first considered the top most similar segment based on BM25 scores after query expansion and stemming. This was the 'answer segment'. This answer segment was then broken down into sentences. These sentences were again ranked and reordered based on BM25 similarity with the question. Sentences with negative scores were removed. A threshold of 3 sentences was set for the answers and anything beyond three sentences was not presented in the answer by the dialog agent in the first go. The dialog agent would present the first three sentences and would then ask the user if he or she wanted to see/hear more. If the user answered yes, the agent would go on to present the rest of the answer. This was implemented by passing context variables through the "Session Attributes" field in the Alexa Response template [1]. Furthermore, context variables were also maintained for the current policy and passed through the response templates. The response templates contained all the context that was required to answer context-based questions.

Results

The results for the BM25-based approach with and without query expansion are shown in Table 2. As shown in the table, the MRR with query expansion is higher indicating that query expansion is useful for presenting the relevant segment as the answer. The table also shows the average number of words and the average number of sentences returned in the responses identified by each method. The evaluation was performed based on manually preparing a set of questions. In the future, we would like to build a UI interface and crowdsource evaluation.

Discussion

An illustrative example of how this technique works is provided below:

Query	Do you collect my financial information?
Expanded Stemmed Query	pii bill person information why credit/debit cardhold credit automatically your information debit receive card/debit financi collect card, debit/credit data payment card date cvv gather inform technology collect
Answer	Information you provide directly. Some Services enable you to give us information directly. If you order a product or paid service from us, we may ask for your name, contact information, shipping and billing address(es), and credit card information in order to process your order. Some of our Services enable you to communicate with other people. Those communications will be transmitted through and stored on our systems.

As seen in the above example, the answer presented by the system does not contain any words overlapping with the query. Yet, the query expansion of the word 'financial' helped in retrieving the answer segment that contained the words 'credit card'. Based on our preliminary analysis, this approach seems to work relatively well on questions that are somewhat broad in nature, but does not necessarily perform as well on more specific questions.

4.2.3 Deep Neural Models

We also investigated training deep neural network models for answer selection. Our work and network architectures were greatly inspired by [41]. In their paper, the authors use BiLSTM based deep neural models with attention for answer selection. In particular, the deep networks are trained to predict the similarity between two pieces of text. The question and answer candidates are passed through layers of BiLSTM to obtain their respective representations. We have used a very similar model as in [41]. The question asked shapes the attention vector which in turn determines what kind of words to focus on in the answer candidate for predicting answer similarity. This is further discussed below.

Approach

For training this model, we need question and answer pairs. However, due to the lack of training data, we generated questions and answers by leveraging the structure of privacy policies. Most privacy policies are divided into multiple sections and each section contains a header. For training our models, question answer pairs were generated by considering each section heading as a question and the paragraph within the ensuing

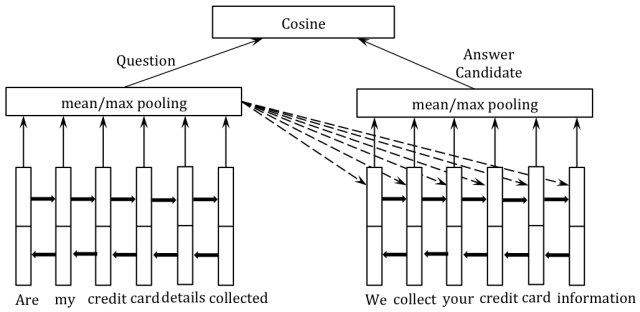

Figure 4: Attention Based Neural Networks to learn similarity. Note: This image is an adaptation of a similar diagram in [41]

section as the correct answer to that particular question. Furthermore, a set of bad answers were also generated by picking segments under different sections within the same policy. We then added question indicators such as "why/when/how/what" to these questions to allow the network to learn to handle questions with "wh" words. The neural network was then trained to output a similarity of 1 for matching question and answers and an output of 0 for bad question and answers. After the question generation phase, we were left with around 3,00,000 question answer pairs. We divided them into them into a train set and a test set, with 80% of the pairs in the train set and 20% in the test set. Further, we set aside 10% of the train data as validation data and the best epoch for testing was picked based on the model's performance on the validation data.

For our experiments, each question and answer was encoded using bidirectional LSTMs. Furthermore, attention vectors were predicted based on the question representation and this was used to attend to specific parts of the answer to predict the similarity. This illustrated in Figure 4.

As in [41], we use a structured hinge loss for training the model as defined below:

$$L = max(0, M - cosine(q, a_+) + cosine(q, a_-))$$

where M is a constant, a_+ is the true answer and a_- is a randomly-sampled wrong answer. We treat any question with more than one ground truth as multiple training examples, each for one ground truth. For our experiments, the ADAM optimizer was used [17]. The dropout regularization parameter was set at 0.5 and M was set to be 0.05.

Intuitively, bidirectional LSTMs are similar to LSTMs in encoding text. They further have the advantage that they encode the text in both forward and backward directions, thus capturing context in both directions. In contrast, LSTMs encode text in one direction. Hence for this approach, bi-LSTMs were used. The attention vector is predicted based on the question and this vector learns the words in the answer candidates to attend to. This allows the same network to handle questions belonging to different categories.

	Top-1 Precision	MRR
Train	1.0	1.0
Test	0.73	0.85

Table 3: Results for Neural Models.

Results and Analysis

The results based on automatic evaluation of the neural models are displayed in table 3. The evaluation is based on the question answer pairs generated as described in the previous section. For each good answer, we randomly sample one of the other segments as a negative example. A total of 131571 question-answer pairs were used to train the models and 30543 question-answer pairs were used for evaluation. We have reported scores for both train and test splits. As seen in the results, the Precision@1 and MRR are both quite high, suggesting a decent level of performance. At the same time, we acknowledge that this preliminary analysis should be supplemented with some manual vetting of the answers and that ultimately one should evaluate their usefulness with actual users.

5 Conclusion & Future Work

In this preliminary work, we have identified and started to explore different approaches for building privacy question answering systems. In particular, we have looked at Closed QA and Open QA approaches. In Closed QA, we tried to map users' natural language questions onto predefined categories of questions or question labels. We explored three levels of granularity for these categories - Value Level (Finest), Attribute Level, Category Level (Coarsest), using annotation the OPP-115 Corpus and its annotation scheme. Our experiments indicated that it was not always possible to map all user questions onto one of the predefined categories. This finding motivated our exploration of Open QA techniques, where the user's question does not need to map onto a predefined data practice category. We also trained Priv2Vec word embeddings - word vectors specifically build for the privacy domain using the Word2Vec skip-gram model.

For Open QA, we first explored an approach where we divided a privacy policy into small segments. Each of these segments was treated as a candidate answer by our system. We showed that the performance of the QA system was better when query expansion was performed as opposed to when it was not. We also devised a length reduction strategy to handle long segments.

We further explored the possibility of training Deep Neural Networks to select the correct answer from candidate answers. We built a Bi-LSTM Attention-based Deep Neural Model to predict similarity between the user's question and candidate answers and used this metric to identify the best answers.

Clearly, a lot of additional work is required before one can hope to field practical Privacy Question Answering functionality such as the one discussed here. Some obvi-

ous next steps in exploring the techniques introduced in this initial report include the following:

- **Improved datasets for privacy question answering:** Of the four datasets we used, only the Twitter dataset contained actual questions from users. As described in Section 3, because this dataset was collected using automated techniques, sampling bias might be a problem. The ideal dataset for privacy question answering would contain questions from users from a more representative sample of the general population.

- **Determining if answers are not present in the privacy policy:** We did not attempt to identify when answers to users' questions are not present in privacy policies. However, it will be important to address this problem in order to provide a good user experience. Users would likely lose confidence in a QA system that does not properly handle omission and ambiguity, given the level of omission and ambiguity typically found in many privacy policies. Ideally, the QA system should be able to differentiate between when the answer is not present in the policy and when the system has difficulty locating the answer.

- **Decrease reliance on statically defined segments:** The quality of our answers depends directly on the quality of the policy segmentation technique, since segments are candidate answers. However, the boundaries between segments are somewhat arbitrary, since they are defined by the policy authors' use of HTML tags, which don't necessarily follow a common convention. The coherence of our answers might possibly be improved if we analyzed policy text at the sentence level.

- **Coreference resolution:** Our current system does not perform coreference and anaphora resolution. Thus, it is possible that certain segments of the privacy policy contain pronouns that cannot be resolved within the text of the segment. This issue can be addressed by applying NLP techniques such as coreference and anaphora resolution. The need for these techniques would be even more important if we switched from analyzing statically defined segments to analyzing sentences, given that a single sentence is less likely to be self-contained than a segment, which typically comprises multiple sentences.

- **Improve answers by using dialogue:** Currently, our QA system accepts a single question and returns a list of candidate answers from the text of the privacy policy. Dialogue could be used to refine this list of candidate answers and help identify the information the user is most interested in. For example, if the user asks why a company collects their location information, the system might find several relevant passages in the privacy policy. The system could ask a followup question, such as "Are you interested in how the company uses your location, or with whom they share your location ?" to determine the passage which is most relevant to the user's interests.

- **Multimodal QA** Usability research has shown that there is value in pairing voice assistants with companion apps [40]. Companion apps can help users discover

which commands are supported by a voice assistant, and can reduce the need for assistants to speak for extended periods of time. It is easy to imagine that users who are unwilling to read privacy polices might be even less willing to listen to voice assistants reciting lengthy passages from the text of privacy policies. An easy solution to this problem would be for the voice assistant to deliver the relevant passages to a companion app that would allow the user to skim through and identify the specific passage(s) he or she is interested in.

6 Appendix

6.1 21 Questions selected from the Twitter dataset used for Bag of Word Cluster Approach

1. @asiaelle @graceishuman I like Evernote for some things but I worry about data security. Who can see my pages ?
2. @Kenshoo I know I can just check your website, but are you taking any personal data while you are looking for our search queries?
3. do you have a warrant canary statement that youve never provided users address books to authorities? If not, can you?
4. @creditkarma Does cancelling an account also delete all associated data (especially SSN) from your system? Want to know before I sign up. :)
5. @NorthumbrianH2O thanks. Do you pass on customer addresses to 3rd parties? Got interiors catalogue addressed to me here. How did they know?
6. @floatapp how secure are your servers? Can you direct me to the security info on your website please?
7. @TTChelps Yes, how will you manage my travel records and contact info, under what circumstances will you release to 3rd parties?
8. @evernotehelps are my notes being saved encrypted on your servers per default? Or is only manually encrypted text encrypted?
9. @fundbox Not too comfy with opening my books. Is my data shared with anyone, and can that change without notice?
10. @fitbit @FitbitSupport Where can I go to see who you sold my private health data too? http://t.co/Rd64dKWGFb
11. With the rapid rise in so called encrypted messaging apps, how do you feel @viber competes on security? #cgc14
12. .@AngiesList so do you sell your mailing list to everyone??? My junk email has increased exponentially since joining. #sheesh
13. @MailChimp Does Mail Chimp retain our list and use or sell them elsewhere ?
14. Hey @SparkNZ where do I find your statemenrt that outlines when our mobile usage data is provided by you to third parties?
15. My money is on marketing plus hubris. @bankofireland, how confident are you that data wont leak?
16. Latest @bankofireland iPhone app update wants constant background access to my location. For security, marketing, or something else?

17. @angrybirds is this true? http://t.co/gCBoFIcZ you send people's contacts to 3rd parties without permission?

18. @onavo Can I ask - why is it free? and how do u guarantee that data is anonymised? Ta :)

19. Also, @HotDocOnline you make no mention on your site of how patient data is secured. Would you like to elaborate in public?

20. Sean: What Mobile Apps Know & Transmit About You: @AngryBirds sends my contacts to third parties? #WTF #FAIL http://t.co/IKVYc6l7

21. @SagiGidali Hi Sagi, what's your stance on keeping customer logs, and where is your company/customer data based for legal reasons?

6.2 Integration with Amazon Alexa

As part of this project, we also integrated our QA systems with Amazon Alexa, Amazon's voice assistant for the Echo family of devices. We developed our service as a 'skill' to integrate with Alexa. Amazon provides a easy to use Amazon Alexa Skill Development Toolkit which offers an easy way to integrate dialog systems with the Amazon Echo devices [1].

Amazon Alexa platform allows developers to develop 'skills' (analogous to mobile apps) for different services which work on the Echo devices. To make building skills possible, they allow developers to define 'intents'. Every utterance by the user is mapped to an 'intent' based on sample utterances that are provided while building the skill. Developers can then define handlers to handle different intents. Alexa provides only the mapped intents and does not explicitly provide access to users' utterances. Hence, we devised a method to gain access to the ASR output, and thus the utterance, which could be useful for many developers to build their skills. The idea is to define intents and slots in such a way that all the words in the vocabulary are captured by the utterance. Amazon Alexa Skill Kit offers an easy to use, develop and test platform for dialog systems. We hosted our dialog system as a web service. We then developed a simple skill which would query this service by providing the user's query. The service would then respond to the skill with the answer that was obtained and the skill converts the text to speech. In this project, we used Amazon Alexa as a speech-to-text and a text-to-speech interface.

References

[1] Amazon Alexa. "Request and Response JSON Reference". In: *Alexa Skills Kit* (2017).

[2] Waleed Ammar, Shomir Wilson, Norman Sadeh, and Noah A Smith. "Automatic categorization of privacy policies: A pilot study". In: *Technical Report* (2012).

[3] Ngo Xuan Bach, Nguyen Le Minh, Tran Thi Oanh, and Akira Shimazu. "A two-phase framework for learning logical structures of paragraphs in legal articles". In: *ACM Transactions on Asian Language Information Processing (TALIP)* 12.1 (2013), p. 3.

[4] Steven Bird. "NLTK: the natural language toolkit". In: *Proceedings of the COLING/ACL on Interactive presentation sessions*. Association for Computational Linguistics. 2006, pp. 69–72.

[5] Danilo S Carvalho, Minh-Tien Nguyen, Chien-Xuan Tran, and Minh-Le Nguyen. "Lexical-Morphological Modeling for Legal Text Analysis". In: *JSAI International Symposium on Artificial Intelligence*. Springer. 2015, pp. 295–311.

[6] Fred H Cate. "The limits of notice and choice". In: *IEEE Security & Privacy* 8.2 (2010), pp. 59–62.

[7] Lorrie Faith Cranor. "Necessary but not sufficient: Standardized mechanisms for privacy notice and choice". In: *J. on Telecomm. & High Tech. L.* 10 (2012), p. 273.

[8] Lorrie Faith Cranor. "P3P: Making privacy policies more useful". In: *IEEE Security & Privacy* 99.6 (2003), pp. 50–55.

[9] Phong-Khac Do, Huy-Tien Nguyen, Chien-Xuan Tran, Minh-Tien Nguyen, and Minh-Le Nguyen. "Legal Question Answering using Ranking SVM and Deep Convolutional Neural Network". In: *arXiv preprint arXiv:1703.05320* (2017).

[10] Mauro Dragoni, Serena Villata, Williams Rizzi, and Guido Governatori. "Combining NLP Approaches for Rule Extraction from Legal Documents". In: *1st Workshop on MIning and REasoning with Legal texts (MIREL 2016)*. 2016.

[11] Morgan C Evans, Jaspreet Bhatia, Sudarshan Wadkar, and Travis D Breaux. "An Evaluation of Constituency-based Hyponymy Extraction from Privacy Policies". In: *Requirements Engineering Conference (RE), 2017 IEEE 25th International*. IEEE. 2017, pp. 312–321.

[12] Anette Frank, Hans-Ulrich Krieger, Feiyu Xu, Hans Uszkoreit, Berthold Crysmann, Brigitte Jörg, and Ulrich Schäfer. "Question answering from structured knowledge sources". In: *Journal of Applied Logic* 5.1 (2007), pp. 20–48.

[13] Filippo Galgani, Paul Compton, and Achim Hoffmann. "Combining different summarization techniques for legal text". In: *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data*. Association for Computational Linguistics. 2012, pp. 115–123.

[14] Hamza Harkous, Kassem Fawaz, Remi Lebret, Florian Schaub, Kang G Shin, and Karl Aberer. "Polisis: Automated Analysis and Presentation of Privacy Policies Using Deep Learning". In: *arXiv preprint arXiv:1802.02561* (2018).

[15] Mitra Bokaei Hosseini, Sudarshan Wadkar, Travis D Breaux, and Jianwei Niu. "Lexical Similarity of Information Type Hypernyms, Meronyms and Synonyms in Privacy Policies". In: 2016.

[16] Mi-Young Kim, Ying Xu, and Randy Goebel. "Applying a Convolutional Neural Network to Legal Question Answering". In: *JSAI International Symposium on Artificial Intelligence*. Springer. 2015, pp. 282–294.

[17] D.P. Kingma and J.L. Ba. "ADAM: A Method for Stochastic Optimization". In: *International Conference on Learning Representations*. ILCR. 2015.

[18] Bin Liu, Mads Schaarup Andersen, Florian Schaub, Hazim Almuhimedi, SA Zhang, Norman Sadeh, Alessandro Acquisti, and Yuvraj Agarwal. "Follow my recommendations: A personalized privacy assistant for mobile app permissions". In: *Symposium on Usable Privacy and Security*. 2016.

[19] Fei Liu, Nicole Lee Fella, and Kexin Liao. "Modeling Language Vagueness in Privacy Policies Using Deep Neural Networks". In: *2016 AAAI Fall Symposium Series*. 2016.

[20] Frederick Liu, Shomir Wilson, Florian Schaub, and Norman Sadeh. "Analyzing Vocabulary Intersections of Expert Annotations and Topic Models for Data Practices in Privacy Policies". In: *2016 AAAI Fall Symposium Series*. 2016.

[21] Yi-Hung Liu, Yen-Liang Chen, and Wu-Liang Ho. "Predicting associated statutes for legal problems". In: *Information Processing & Management* 51.1 (2015), pp. 194–211.

[22] Lars Mahler. *What Is NLP and Why Should Lawyers Care?* `http://www.lawpracticetoday.org/article/nlp-lawyers/`. 2015.

[23] K. Tamsin Maxwell and Burkhard Schafer. "Concept and Context in Legal Information Retrieval". In: *Proceedings of the 2008 Conference on Legal Knowledge and Information Systems: JURIX 2008: The Twenty-First Annual Conference*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2008, pp. 63–72. ISBN: 978-1-58603-952-3. URL: `http://dl.acm.org/citation.cfm?id=1564008.1564016`.

[24] Aleecia M McDonald and Lorrie Faith Cranor. "Cost of reading privacy policies, the". In: *ISJLP* 4 (2008), p. 543.

[25] Anca Micheti, Jacquelyn Burkell, and Valerie Steeves. "Fixing broken doors: Strategies for drafting privacy policies young people can understand". In: *Bulletin of Science, Technology & Society* 30.2 (2010), pp. 130–143.

[26] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. "Distributed representations of words and phrases and their compositionality". In: *Advances in neural information processing systems*. 2013, pp. 3111–3119.

[27] Diego Mollá and José Luis Vicedo. "Question answering in restricted domains: An overview". In: *Computational Linguistics* 33.1 (2007), pp. 41–61.

[28] Alfredo Monroy, Hiram Calvo, and Alexander Gelbukh. “NLP for shallow question answering of legal documents using graphs”. In: *Computational Linguistics and Intelligent Text Processing* (2009), pp. 498–508.

[29] Simonetta Montemagni, Wim Peters, and Daniela Tiscornia. *Semantic Processing of Legal Texts*. Springer, 2010.

[30] Kanthashree Mysore Sathyendra, Shomir Wilson, Florian Schaub, Sebastian Zimmeck, and Norman Sadeh. “Identifying the Provision of Choices in Privacy Policy Text”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, 2017, pp. 2774–2779.

[31] Alessandro Oltramari, Dhivya Piraviperumal, Florian Schaub, Shomir Wilson, Sushain Cherivirala, Thomas B Norton, N Cameron Russell, Peter Story, Joel Reidenberg, and Norman Sadeh. “PrivOnto: A semantic framework for the analysis of privacy policies”. In: *Semantic Web* Preprint (2017), pp. 1–19.

[32] Jeffrey Pennington, Richard Socher, and Christopher D Manning. “Glove: Global Vectors for Word Representation.” In: *EMNLP*. Vol. 14. 2014, pp. 1532–1543.

[33] Paulo Quaresma and Irene Pimenta Rodrigues. “A Question Answer System for Legal Information Retrieval.” In: *JURIX*. 2005, pp. 91–100.

[34] Rohan Ramanath, Fei Liu, Norman Sadeh, and Noah A. Smith. “Unsupervised Alignment of Privacy Policies using Hidden Markov Models”. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Baltimore, Maryland: Association for Computational Linguistics, 2014, pp. 605–610.

[35] Radim Řehůřek and Petr Sojka. “Software Framework for Topic Modelling with Large Corpora”. English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. `http://is.muni.cz/publication/884893/en`. Valletta, Malta: ELRA, May 2010, pp. 45–50.

[36] Joel R Reidenberg, Travis Breaux, Lorrie Faith Cranor, Brian French, Amanda Grannis, James T Graves, Fei Liu, Aleecia McDonald, Thomas B Norton, and Rohan Ramanath. “Disagreeable privacy policies: Mismatches between meaning and users’ understanding”. In: *Berkeley Tech. LJ* 30 (2015), p. 39.

[37] Norman Sadeh, Alessandro Acquisti, Travis D Breaux, Lorrie Faith Cranor, Noah A Smith, Fei Liu, Florian Schaub, et al. “The Usable Privacy Policy Project”. In: *Technical Report* (2013).

[38] Florian Schaub, Rebecca Balebako, Adam L. Durity, and Lorrie Faith Cranor. “A Design Space for Effective Privacy Notices”. In: *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*. Ottawa: USENIX Association, July 2015, pp. 1–17. ISBN: 978-1-931971-249.

[39] Tobias Schnabel, Igor Labutov, David M Mimno, and Thorsten Joachims. “Evaluation methods for unsupervised word embeddings.” In: *EMNLP*. 2015, pp. 298–307.

[40] Max Silverman and Katie Quehl. “UX Research for Oath”. OATH: Tech Talk. 2017.

[41] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. “LSTM-based deep learning models for non-factoid answer selection”. In: *arXiv preprint arXiv:1511.04108* (2015).

[42] Matthew W Vail, Julia B Earp, and Annie I Antón. “An empirical study of consumer perceptions and comprehension of web site privacy policies”. In: *IEEE Transactions on Engineering Management* 55.3 (2008), pp. 442–454.

[43] S. Wilson, F. Schaub, F. Liu, K.M. Sathyendra, S. Zimmeck, R. Ramanath, F. Liu, N. Sadeh, and N.A. Smith. “Analyzing Privacy Policies at Scale: From Crowdsourcing to Automated Annotations”. In: *ACM Transactions on the Web* (2017). Submitted.

[44] S Wilson, F Schaub, A Dara, F Liu, S Cherivirala, P G Leon, M S Andersen, S Zimmeck, K Sathyendra, N C Russell, T B Norton, E Hovy, J R Reidenberg, and N Sadeh. “The Creation and Analysis of a Website Privacy Policy Corpus”. In: *Annual Meeting of the Association for Computational Linguistics, Aug 2016*. ACL. 2016.

[45] Sebastian Zimmeck, Ziqi Wang, Lieyong Zou, Roger Iyengar, Bin Liu, Florian Schaub, Shomir Wilson, Norman Sadeh, Steven M Bellovin, and Joel Reidenberg. “Automated analysis of privacy requirements for mobile apps”. In: 2017.