Towards Change Detection in Privacy Policies with Natural Language Processing

Andrick Adhikari
Computer Science
University of Denver
Denver, USA
andrick.adhikari@du.edu

Rinku Dewri
Computer Science
University of Denver
Denver, USA
rinku.dewri@du.edu

***Abstract*—Privacy policies notify users about the privacy practices of websites, mobile apps, and other products and services. However, users rarely read them and struggle to understand their contents. Due to the complicated nature of these documents, it gets even harder to understand and take note of any changes of interest or concern when the policies are changed or revised. With advances in machine learning and natural language processing, tools that can automatically annotate sentences of policies have been developed. These annotations can help a user identify and understand relevant parts of a privacy policy. In this paper, we present our attempt to further such annotations by also detecting the important changes that occurred across sentences. Using supervised machine learning models, word-embedding, similarity matching, and structural analysis of sentences, we present a process that takes two different versions of a privacy policy as input, matches the sentences of one version to another based on semantic similarity, and identifies relevant changes between two matched sentences. We present the results and insights of applying our approach on 79 privacy policies manually downloaded from Facebook, WhatsApp, Twitter, Google, LinkedIn and Snapchat, ranging between the period of 1999 to 2020.**

***Index Terms*—Change Detection, Usable Privacy Policy, Semantic Similarity, Security and privacy**

I. Introduction

A privacy policy is a statement or a legal document (in privacy law) that discloses some or all of the ways a party gathers, uses, discloses, and manages a client's data. ISO/IEC 29100 establishes 11 privacy principles for competent privacy management in an organization, namely consent and choice; purpose legitimacy and specification; collection limitation; data minimization; use, retention and disclosure limitation; accuracy and quality; openness, transparency and notice; individual participation and access; accountability; information security; and privacy compliance [1]. These principles should be adequately reflected in the respective privacy policy of an organization. Various legal regimes around the world require that website operators, app publishers, data processors, and service providers post a notice on how they gather and share users' information [2]. This requirement results in a large number of privacy policy documents that most users are unlikely to read due to their incomprehensible nature. As a matter of fact, if users start reading policies for each of the services they use, it is estimated that it would cost them at least 181 hours per year [3]. Further, whenever there is a change in the data collection and use practices followed by an organization, the respective policy is revised and modified to reflect that change. Even if a user is well-versed in the organization's privacy policy (unlikely), it is a tedious task to become aware of the critical changes through perusal. The sheer number of privacy policies (a policy for each of the used websites or services) further demotivates users to be well informed about changes in those policies. Hence, most users remain in the dark whenever a change is introduced and are unable to make informed decisions related to their privacy. The goal of this work is thus to work towards a tool that facilitates the comparison of two versions of a privacy policy with ease, and help users learn about important changes between the two

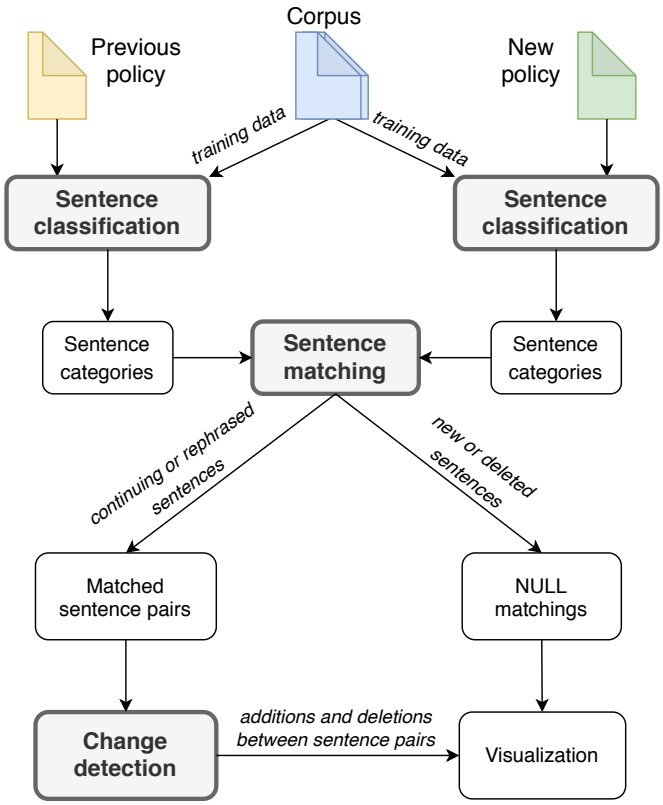

Fig. 1. Schematic of tasks for automated change detection in privacy policies.

documents. We bring together a number of existing methods in machine learning and natural language processing, and prior work in usable privacy policies, and apply them to a problem in privacy assessment, namely that of automatically detecting privacy-related changes in an organization's privacy policy. We present a workflow culminating into a prototype tool that extracts and presents the changes between two versions of a policy in a comprehensible format. The workflow prunes redundant comparisons by performing change detection through a sequence of three filtering tasks, namely sentence classification guided by a corpus of 115 privacy policies that have been manually annotated for fine-grained data practices and categories, followed by sentence matching driven by similarity measures on word-embeddings, and finally change detection using dependency tree analysis (Figure 1). In this study, we focus on identifying changes in nouns and their context verb to provide the user with a minimalistic, yet relevant change information. We exemplify and bring to light some observations learned from using this workflow on 79 policies from six global technology organizations (Facebook, WhatsApp, Twitter, Google, LinkedIn and Snapchat), as they went through changes from the year 1999 to 2020.

The remainder of the paper is organized as follows. Section II presents related work on usable privacy policies and change detection. This is followed by a description of the data we use and our approach in Sections III and IV, including discussion on method choices. Section V presents results and insights highlighted through example sentences from privacy policies. Finally, we conclude in Section VI.

II. Related Work

Natural language privacy policies are the primary medium used by organizations to communicate information addressing user data collection and use, and expectations thereof [4]. Nevertheless, the lengthy and complex nature of privacy policies often hinders their usefulness, and therefore they are neglected by users [5]. McDonald et al. viewed that the time required to read privacy policies is equivalent to a form of payment, where users pay with their time to research policies in order to secure their privacy [3]. Even when users attempt to understand the practices described in the policies, there is often a disagreement between the actual meaning and the user's interpretation [6].

Vail et al. conducted an empirical study of the different ways of presenting information from privacy policies to consumers, and discussed how the mode of presentation impacts a user's perception [7]. Micheti et al. developed guidelines for constructing privacy policies that are simple enough for children and teenagers to interpret with relative ease [8].

Under the Usable Privacy Policy Project, there are extensive research works on using natural language processing to understand the content of privacy policies [9]. Liu et al. presented advances in using supervised learning to automatically annotate paragraphs and sentences in a policy with expert-identified categories of policy content [10]. Ramanath et al. approached the annotation of privacy policy segments as an alignment problem by using hidden markov models [11]. Ammar et al. utilized logistic regression to predict the presence or absence of content pertaining to transparency on law enforcement requests and a user's right to terminate an account [12]. Wilson et al. studied the accuracy of crowd sourced privacy policy annotations, the levels of granularity in annotations that are feasible for automatic analysis of privacy policies, and formulated guidelines that can enhance crowd worker productivity in annotation tasks [13]. Sathyendra et al. and Habib et al. focused on extracting user choices in privacy policies, particularly opt-out choices [14], [15]. Sathyendra et al. explored a query answering approach that would enable users to ask questions about specific aspects of privacy policies, and receive answers in the form of short text fragments from a policy [16]. A similar work is presented by Harkous et al. describing conversational privacy bots (PriBots) that build on machine learning techniques to automatically annotate the text of privacy policies [17]. Cherivirala et al. presented a website that facilitates users with a visualization tool for mapping text segments onto meaningful categories of data collection and use practices [18]. Zimmeck et al. introduced a scalable system to help predict the compliance of Android apps' with privacy requirements by combining machine learning-based privacy policy analysis with static code analysis of apps [19]. Wilson et al. created a dataset of privacy policies from websites, and had domain experts annotate the different sections in a policy with data practice category labels. [20].

Amos et al. curated and longitudinally analysed policies from 130,000 distinct websites spanning over two decades [21]. But despite tremendous effort and research on identifying policy content categories, works in detecting changes between revisions of a privacy policy are rare.

Based on this literature review, we identify that the potential for natural language processing to perform change detection in privacy policies is still untapped.

III. Data Sets

A. OPP-115 Corpus

The (Online Privacy Policies) OPP-115 corpus is a collection of 115 website privacy policies with annotations that specify data practices in the text. Each privacy policy was read and annotated by three graduate students in law [20]. At a high-level, these annotations fall into one of the ten data practice categories developed by a team of legal experts – First Party Collection/Use; Third Party Sharing/Collection; User Choice/Control; User Access, Edit, & Deletion; Data Retention; Data Security; Policy Change; Do Not Track; International & Specific Audiences; and Other. We extract sentence text and data practice pairs from the corpus files to build our sentence classifier.

B. Privacy policies

Using the Wayback Machine archive[1], we collected the available versions of the privacy policy for six global tech-

[1]https://archive.org/web/

nology organizations. These policies are collected as HTML pages, then processed into a text file by discarding HTML tags. We also manually followed the privacy-relevant links and added the text from those secondary pages into the policy text file. We collected multiple versions of policies for the following organizations: Facebook (18 policies from 2005 to 2018), WhatsApp (6 policies from 2009 to 2019), Twitter (14 policies from 2007 to 2020), Google (27 policies from 1999 to 2017), LinkedIn (5 policies from 2013 to 2020) and Snapchat (9 policies from 2017 to 2020). We chose these select few websites since a large part of their operations depend on information collected from users. The snapshots were chosen such that we have a single copy of each version of the policy, identified using the revision date in the policy.

C. Other data

We use the glove.-840B.300d corpora[2] to train our word embedding tools (GloVe and Word2Vec). This corpora contains 840 billion tokens, 2.2 million vocabulary and 300 dimensional cased vectors created using a common crawl of the web.

IV. Methodology

Our workflow towards change detection consists of three stages: sentence classification, sentence matching and change detection. Sentence classification first annotates each sentence in the two policies under comparison with a data practice category as identified for the OPP-115 corpus. Subsequently, during sentence matching, we consider sentence pairs (for matching) between the two policies only if they have the same category label. Limiting the matching process in this manner not only enhances the computational efficiency of the process, but also reduces the chances of a bad match owing to similar usage of words but with respect to different privacy categories that are identifiable by only a few terms in the sentences. Each sentence in a matched pair is then processed using a dependency tree to extract a *token list* including identified nouns and proper nouns, along with their compound or adjective modifiers. The token lists of the two sentences are then compared to identify additions/deletions of words, which can then be visualized side-by-side in an appropriate interface. We elaborate on the three stages in the next few sections.

A. Sentence classification

We begin this stage by first converting all sentences in all policies (the OPP-115 corpus, as well as the policies manually downloaded) to lowercase, and then tokenizing each sentence. The tokens of a sentence are further processed by removing English stop words and non-alphabetic words. Finally, the tokens are lemmatized and added to the final list of tokens for the sentence.

Next, a TF-IDF based vectorizer is prepared using the OPP-115 corpus. The weights of these words, i.e. the value put into the vector, are determined based on their number of appearances in a document relative to their appearance in the entire corpus. Use of the OPP-115 corpus to determine the weights is critical so that privacy policy related tokens are assigned appropriate weights. The prepared vectorizer is then used to encode the token list of each sentence to a TF-IDF vector.

Next, we split the encoded vectors and their corresponding categories from the OPP-115 corpus into a 70% training set and a 30% testing set. The training set is used to train two supervised learning models, namely logistic regression and support vector machine, and evaluated using the testing set. This process is repeated 10 times for both models, each time randomly splitting the training and testing set according to the mentioned ratio. Logistic regression performed with a precision of 0.59, recall of 0.77 and F1-score of 0.65, whereas the SVM classifier had a precision of 0.74, recall of 0.83 and F1-score of 0.78. Our observed results of the evaluation are on par with results from already established tools using these machine learning models [10], and validated the proficiency of a SVM classifier and it's superiority over logistic regression in automated annotation of privacy policies. We also evaluated the classifiers using the policies of Facebook, Twitter and WhatsApp, which produced similar results. Hence, the SVM classifier is subsequently used to assign category labels to the sentences in our downloaded privacy policies.

B. Sentence matching

The primary goal in sentence matching is to generate pairs of sentences across two versions of a privacy policy such that both sentences have the same semantics and help detect user-relevant changes between them. Word2Vec and GloVe embedding are potential techniques that we evaluate to finalize one for this task. Word2Vec employs either continuous bag-of-words or skip-gram architectures to learn syntactic and semantic information among words, and preserves such regularities in the generated high-dimensional vector representations of the words [22]. GloVe (global vectors for word representation) is an unsupervised learning algorithm for obtaining vector representations using word-word co-occurrence [3]. While Word2Vec is directed by statistics of co-occurring words in the neighborhood of a given word, GloVe relies on global aggregated word-word co-occurrence counts. In addition to the word-embedding method, the similarity measure also plays a pivotal role. Hence, cosine similarity, Word Mover's Distance(WMD) and a third variant (Smooth Inverse Frequency) are also evaluated in permutation with Word2Vec and GloVe.

We use the Facebook privacy policies from September 2016 and October 2018 to select one of these permutations as the final one. This pair of policies has relevant changes between semantically same sentences from both the documents, but the overall changes are not so extreme as to make the whole matching process irrelevant. During matching, a sentence is taken from the September 2016 policy and then paired with the highest similarity score sentence in the October 2018 policy. We create a ground truth by manually pairing up

[2]http://nlp.stanford.edu/data/wordvecs/glove.840B.300d.zip

[3]https://nlp.stanford.edu/projects/glove/

TABLE I
TOP-THREE MATCHING SENTENCES TO AN EXAMPLE SENTENCE S_1 FROM THE FACEBOOK SEPTEMBER 2016 POLICY USING WORD2VEC WITH COSINE SIMILARITY VERSUS GLOVE WITH WMD.

Word2Vec with cosine similarity	GloVe with WMD
Facebook may share information internally within our family of companies or with third parties for purposes described in this policy.	Facebook may share information internally within our family of companies or with third parties for purposes described in this policy.
In some cases, people you share and communicate with may download or re-share this content with others on and off our Services.	We share information we have about you within the family of companies that are part of Facebook.
We work with third party companies who help us provide and improve our Services or who use advertising or related products, which makes it possible to operate our companies and provide free services to people around the world.	We receive information about you from companies that are owned or operated by Facebook, in accordance with their terms and policies.

the semantically same sentences from the two policies. It is expected that a good combination of word-embedding and similarity measure choices will result in paired sentences to have a score with a high margin of difference from similarity measures for dissimilar sentence pairs.

Following an analysis of the different permutations of methods, we identified that GloVe embedding with WMD as the similarity measure results in the best precision in matching, with 67 out of the 92 sentence pairs in the ground truth (Facebook September 2016 versus October 2018) being correctly paired. Comparatively, other permutations were able to match fewer pairs correctly, or the score corresponding to a matched pair was too close to the neighboring matches (next two candidate matches), or the matches identified for a sentence, beyond the first one, had little or no relation to the sentence. For example, considering the following sample sentence from the September 2016 policy, Table I lists the top three matches found by Word2Vec with cosine similarity versus the matches found by GloVe with WMD.

> S_1:"We share information globally, both internally within the Facebook Companies, and externally with our partners and with those you connect and share with around the world in accordance with this policy."

While both methods identified the correct matching sentence in the October 2018 version, the other candidate sentences identified by Word2Vec with cosine similarity deviate more aggressively from the semantics of S_1 than those identified by GloVe with WMD. Observe that Word2Vec with cosine similarity missed the candidate "*We share information we have about you within the family of companies that are part of Facebook.*" in its top-three list, even though it is semantically closer to S_1.

1) Using a threshold: Matching sentences without having a threshold on the similarity score leads to formation of incorrect pairs. For a sentence that should not have a match with any sentence in the newer version of the policy (no semantically equivalent sentence due to deletion, or a major policy revision), the absence of a threshold in similarity measures can lead to matching of such sentences with a sentence from the next policy. To prevent this, we use a threshold value such that if the similarity score for a matched pair of sentences is worse than this value, then the pair is not considered as a valid match. Matching of sentences without a threshold leads to high recall at the cost of precision (due to increase in false positives). We perform a precision versus threshold, and a recall versus threshold, exploration on the September 2016 and October 2018 Facebook privacy policy versions, and use the threshold value corresponding to the intersection of the two curves as the threshold choice (Figure 2). Any unmatched sentence in either policy is indicated to have a NULL match, implying that the entire sentence is either deleted (from the old policy) or newly added (to the new policy). We do note that the threshold choice is only based on the two Facebook policies in this work; albeit, an extended precision/recall analysis can be performed on a collection of policy pairs provided ground truth matching is available for them. Such ground truth datasets are currently

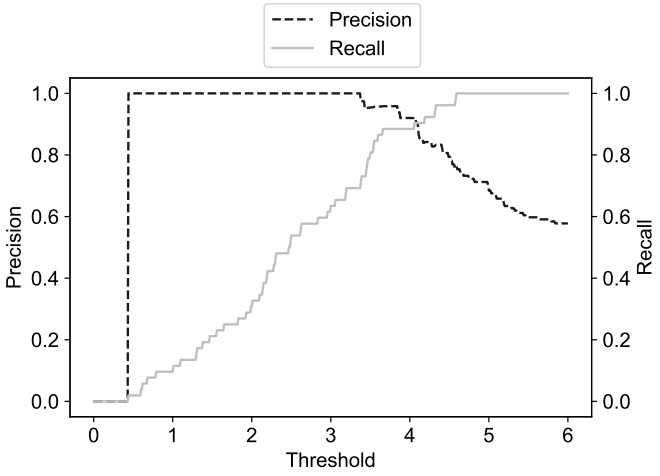

Fig. 2. Precision–Recall–Threshold graph of sentence matching between Facebook privacy policies of September 2016 and October 2018 using GloVe with WMD.

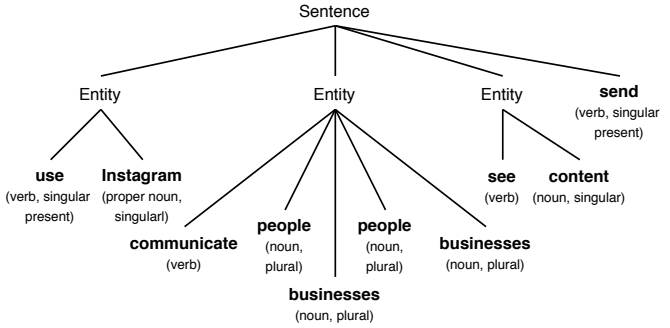

Fig. 3. Parse tree for sentence *''Similarly, when you use Messenger or Instagram to communicate with people or businesses, those people and businesses can see the content you send."*

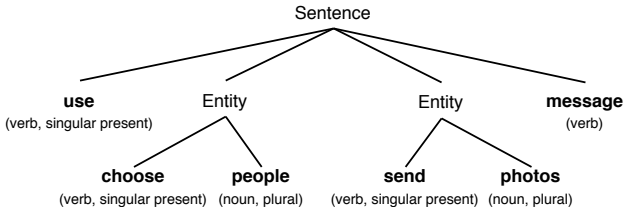

Fig. 4. Parse tree for sentence *"Likewise, when you use Messenger, you also choose the people you send photos to or message."*

non-existent, are resource-intensive to create, and also does not necessarily guarantee superior results when used. Nonetheless, even if a single policy pair was used for threshold choice, the obtained results were found to be sufficient and not lacking, as validated by us (Section V-C).

2) Matching within categories: To improve the matching speed and make the pairing process more efficient, we leverage our sentence classifier. If two sentences form a correct pair, then both of them should belong to the same category as well. We first categorize each sentence of two consecutive policies, say P_1 and P_2, using the classifier, and then for each sentence of P_1, we compute the similarity measures with only those sentences in P_2 that have the same data practice category as the sentence from P_1. For the Facebook privacy policy pair of September 2016 and October 2018, the precision improved from 0.73 to 0.83 (76 out of 92 pairs correctly matched) by performing such intra-category matching.

C. Change detection

The change detection process extracts user-relevant changes between two matched sentences by identifying the additions and deletions of nouns between them. Identifying noun changes serve as a basic step towards learning about changes in policy critical artifacts such as data collection methods, third party vendors, shared data types, control mechanisms, retention periods, and communication addresses, among others. We also extract the verb describing the action for the added or deleted nouns to provide context of the change. This identification will help inform users of changes in a more readable and comprehensible manner, enabling them to understand the changes introduced in the newer revision of the policy.

1) Parse tree: The first approach we explore is to generate a parse tree of the two sentences and compare their structures to detect changes between them. We identify changes in nouns and verbs describing the action, state, or occurrence of the said noun. Hence, during tokenization of the sentences, only noun or verb tokens are kept, and tokens with other parts of speech tags are removed from the list. Sentences comprising of only nouns and verbs are then chunked into parse trees using the chunk regular expression parser in nltk. The chunking returns a tree where the verb token and its corresponding subject noun tokens are grouped together in the same subtree as leaf nodes. Each of these subtrees are labeled as "Entity" and they represent the context of the noun tokens in the sentence. For example sentences:

> S_2: "Similarly, when you use Messenger or Instagram to communicate with people or businesses, those people and businesses can see the content you send."

from the Facebook privacy policy of October 2018 and

> S_3: "Likewise, when you use Messenger, you also choose the people you send photos to or message."

from the Facebook privacy policy of September 2016, their respective parse trees are shown in Figure 3 and Figure 4.

We hypothesized that additions and deletions of subtrees of the matched sentences, and changes in the structure of such subtrees, should be sufficient to extract changes between two sentences. However, we observe a few issues with this approach that can make detecting changes difficult. The first issue is due to incorrect tagging of parts of speech for some tokens. For example, in both sentences S_2 and S_3, the word "Messenger" is not identified as a noun; instead it is tagged as an adjective, leading to absence of "Messenger" from the parse trees. Ideally the parse tree for S_2 should have an entity subtree with nodes "use", "Instagram" and "Messenger", and the parse tree for S_3 should have had an entity subtree with nodes "use" and "Messenger". The change could have then been detected by comparing the two subtrees from both sentences. Another issue with using a chunk parser is that some of the dependency links get lost. Dependency links represent the relationship between two tokens in a sentence. For example, in S_2, the dependency link between the verb "send" and it's subject "content" is lost due to the unilateral nature of the chunking process. Hence, "send" and "content" are not in the same entity subtree.

Even though this approach failed, the analysis helped in formulating the final method for detecting changes. We observed that user-relevant changes can be detected by using the dependency links of noun tokens to identify the context verb. We move to use a dependency tree to implement this approach.

2) Dependency tree: A dependency tree is a parse tree based on the dependency grammar. Dependency grammar cap-

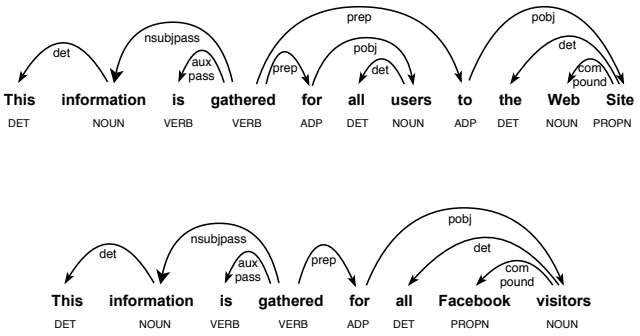

Fig. 5. Dependency trees for sentences S_4 (top) and S_5 (bottom). Labels *DET, NOUN, VERB, ADP* and *PROPN* are pos-tags for determiner, noun, verb, adpositon and proper noun respectively. Edge labels *det* (determiner), *nsubjpass* (passive nominal subject), *auxpass* (auxiliary passive), *prep* (prepositional modifier), *pobj* (object of preposition) and *compound* (compound) represent the dependency relationships.

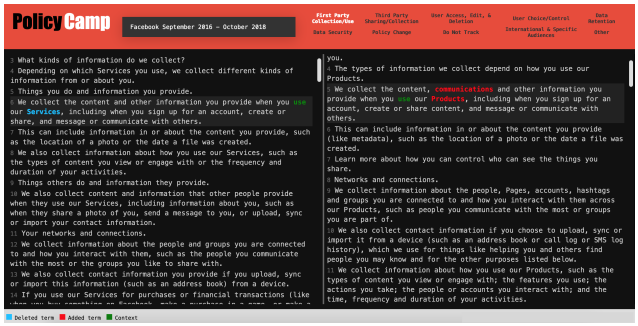

Fig. 6. Example interface for comparing two versions of a privacy policy highlighting sentence categories and changes.

tures the relationship between words or tokens in a sentence. The idea behind this is that every word is connected to each other by a direct or an indirect link. A verb token of the sentence is taken to be the structural center of the dependency tree and other words are either directly or indirectly connected to this verb token in terms of the directed links, which are called dependencies. The overall structure of the tree is determined by the relation between the head token (center verb or root word) and its dependent tokens (other words in the sentence) connected by these links.

Consider the matched sentences

> S_4: "This information is gathered for all users to the Web Site."

from the Facebook privacy policy of June 2005 and

> S_5: "This information is gathered for all Facebook visitors."

from the Facebook privacy policy of February 2006. Their dependency trees are shown in Figure 5. As seen in the figure, dependency based parse trees provide a better parts-of-speech tagging for the tokens using dependencies between words rather than using a dictionary for pos-tagging. It also provides relationship labels between tokens.

We first traverse the trees to identify nouns and proper nouns, and other tokens having compound or adjective modifier relationship with the identified noun/proper noun tokens. In the process, we create two dictionaries for the two sentences using the identified tokens, along with their adjective modifier and compound tokens, as the key, and the tokens themselves as the value. Subtraction between these two dictionaries using the keys gives the deleted and added noun tokens between two sentences. The purpose of forming keys in this manner is to eliminate false detection due to different arrangement of words but implying the same meaning. Using only noun tokens without their adjective modifiers can lead to missing of some of the important changes introduced by it. After the new or deleted noun tokens are identified, we identify the context verbs by traversing from a token to the *root* of the tree using the dependency links. The dependency links ensure that there exists a path to the action verb of the token, if it is present, thus identifying the action or state of the token. Using this method, changes between two matched sentences are extracted in terms of an added token and a deleted token list, which can then be highlighted in a user interface. Figure 6 shows an example web application interface showing such changes.

In summary, the change detection workflow consists of a SVM classifier to assign data practice categories to sentences in a policy, GloVe with WMD (applied intra-category) to match sentences between a policy and its subsequent revision, and a dependency tree based token extraction to identify additions and deletions on matched sentences. We next apply this workflow (with no further changes to the tuned parameters) to all the policy pairs in the manually downloaded set.

V. Policy Comparison Observations

The 79 manually downloaded policies gives us a total of 73 policy pairs to compare. Each policy is compared with the subsequent policy within an organization.

A. Execution time

We measure the execution time to perform sentence classification, sentence matching, and change detection in each policy pair comparison. Sentence classification and change detection takes around 0.3 seconds on average (across 10 runs). Sentence matching is observed as the bottle neck, with computation time reaching as high as 10 minutes for a large policy pair. However, policy comparisons are one-time tasks that only need to be run when a policy revision occurs.

B. Sentence classification

We categorized all 12,188 sentences in the 79 privacy policies using the SVM classifier. The category wise percentage composition of the data set is shown in Table II. Most sentences belong to the "First Party Collection/Use" category. This is also the case with the OPP-115 corpus, which leads us to have higher confidence in the correctness of these classifications [20]. Consequently, we can say that privacy policies are mostly focused on how user data is collected and used by an organization. The other two major categories,

TABLE II
DISTRIBUTION OF SENTENCE CATEGORY LABELS ACROSS COLLECTED POLICIES.

Category	Percentage (%)
First Party Collection/Use	41.87
Third Party Sharing/Collection	19.16
Other	18.44
User Choice/Control	6.56
User Access, Edit and Deletion	5.07
Data Security	3.34
Policy Change	2.07
Data Retention	1.83
International and Specific Audiences	1.63
Do Not Track	0.03

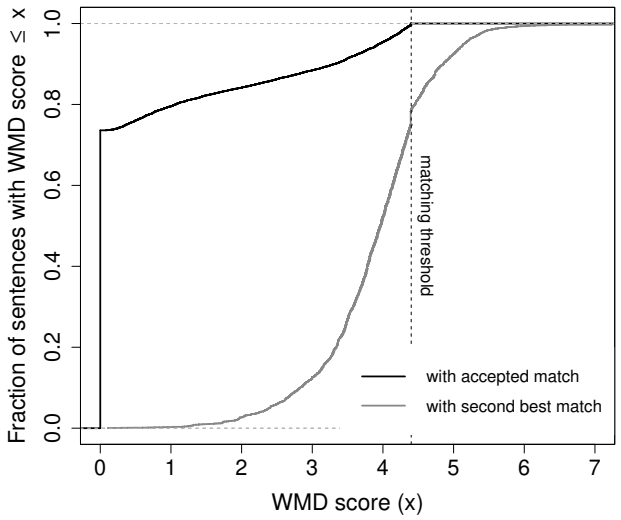

Fig. 7. Empirical cumulative distribution of WMD scores (lower is better) during sentence matching.

"Third Party Sharing/Collection" and "Other," interchangeably have the second and third highest ratio in the documents. How a company shares user data with third party services or companies have been a major concern for many users, and the high ratio of sentences for "Third Party Sharing/Collection" shows that privacy policies have tried to address these user concerns. "Other" category represents sentences which cannot be confidently categorized into other categories due to lack of clear interpretation. The high frequency of "Other" category sentences can be taken as a hint towards the ambiguous nature of many privacy policies. Low number of "User Choice/Control", "Do Not Track", "User Access, Edit and Deletion" and "Data Retention," cumulatively comprising of 13.49 percent of sentences, shows how limited control users have over their own data. Even if the users are provided with options of controlling the use and collection of data, policies rarely provide detailed explanation or instructions.

Through manual analysis of select few policy pairs, we observed that certain tokens which have higher association with other categories can sometimes create a bias in classification. For example, the sentence *"We also offer easy-to-use security tools that add an extra layer of security to your account"* should fall under the "Data Security" category but is misclassified as"First Party Collection/Use" due to the presence of words "we", "use" and "account," which are commonly observed in sentences belonging to"First Party Collection/Use." Some sentences such as *''We collect information from or about the computers, phones, or other devices where you install or access our Services, depending on the permissions you have granted"* can technically belong to more than one category (either "User Choice/Control" or "First Party Collection/Use" in this case but predicted as "Other"), and often get wrongly classified. This indicates that the ambiguity in interpretation of sentences while annotating texts during the creation of OPP-115 corpus has been reflected in the trained classifier, leading to a fuzzy-logic-like classification in some cases. However, the observed bias is not significant enough to invalidate the bulk of the classification results.

C. Sentence matching

Figure 7 shows the empirical cumulative distribution of the WMD similarity scores for sentences that had a match, i.e. the best matching sentence in the revised policy had a score better than the threshold. WMD is a distance based measure, hence a lower score is better, with a score of zero being an exact match. Recall that we obtained a threshold value (4.4) based off of only a pair of Facebook privacy policies. About 80% of the matches are accepted with a score less than one. Further, we see a high margin between the scores for the best match and the second best match, indicating that most sentences are matched with high confidence.

Reading and checking the correctness of all the sentence pairs is a laborious task, but Figure 7 gives an estimate of the confidence in the sentence matching method. Nonetheless, we manually validated each matched sentence pair in the Facebook, Twitter and WhatsApp policies (5,044 matched sentence pairs across 35 policy pairs). This allowed us to judge the precision of the matching process for the three organizations, but we are unable to determine the recall (what percentage of actual matches has been detected). In terms of precision, 30 of the 35 policy pair comparisons had a precision higher than 80% in the matched sentences (95% in 23 of the 35 pairs). We also observed that lower matching precision often appeared with a high number of NULL matches (number of sentences in an original policy that were not matched to any sentence in the revised policy). Scrutiny of such policy pairs revealed that the policy went significant structural changes during the revision. For example, there were 118 NULL matches during the Facebook policy revision from November 2013 to January 2015, and 42 NULL matches during the WhatsApp policy revision from July 2012 to August 2016. These policy revisions encompass a time

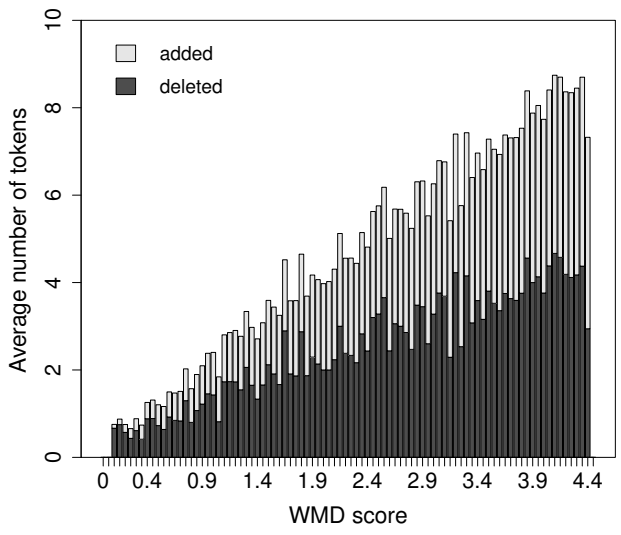

Fig. 8. Average number of added and deleted tokens (stacked) for matched sentence pairs with a given WMD score (lower score implies better match). Scores are binned into groups of size 0.05 for plotting.

period when Facebook purchased WhatsApp in 2014, and introduced changes owing to the change of ownership. As such, NULL matches can be a valuable indicator of potentially large changes in an organization's structure and practices.

The manual validation also showed that GloVe with WMD had no difficult matching sentences that did not change across revisions. The method also matched semantically similar sentences with slight restructuring–*"Depending on which Services you use, we collect different kinds of information from or about you."* and *"The types of information we collect depend on how you use our Products."*–as well as sentences that underwent significant deletion of words–*"In addition, when you download or use such third-party services, they can access your Public Profile, which includes your username or user ID, your age range and country/language, your list of friends, as well as any information that you share with them."* and *"Also, when you download or use such third-party services, they can access your public profile on Facebook, and any information that you share with them ."*

D. Change detection

Figure 8 shows the number of added/deleted tokens between a pair of matched sentences with respect to the similarity score of the pair. As the similarity measure becomes worse, the number of detected changes increases. This is expected as changes between sentence pairs reduce the similarity measure. This also creates an avenue for an improvement–high number of detected addition and deletion of tokens in sentence pairs with poor similarity measures show that a threshold may be computable for the number of detected changes to eliminate false sentence matches. This can be further extended to identify extreme changes between a policy pair to estimate the applicability of the proposed method for the pair. Extreme changes make comparing policies using the proposed methods redundant, as the newer version of the policy can be considered a new document.

To exemplify the kind of detected changes, we consider few sample sentences from Facebook's revisions. In the following, changes are marked with an underline and context verbs, if present, are emphasized. In the pair dependency tree analysis

"Depending on which Services you *use* , we collect different kinds of information from or about you."	'The types of information we collect depend on how you *use* our Products."

helped infer the change from "Services" to "Products," and the replacement of "different kinds" with "types." The token "use", which defines the action on the changed tokens is also detected. Note that using relationship links between tokens ensured that "different" and "kinds" are treated as a single entity. We refer to this form of change detection as "complete." Next, consider the pair: Even though the method highlighted

"You can find additional tools and information at Privacy Basics."	"You can find additional tools and information in the Facebook Settings and Instagram Settings."

the relevant changes, it should have also highlighted "Settings" after "Facebook" in the second sentence. The relationship between "Facebook" and "Settings" is not captured in the dependency tree, and hence the two words are not treated as a single entity. We observed similar failure to detect some of the compound nouns in other sentence pairs as well. We refer to such detection where minute changes are missed as "almost complete." In some sentence pairs, such as the following, the

"When you *comment* on another person's post or like their content on Facebook, that person decides the audience who can see your comment or like."	"Also, when you *comment* on someone else's post or react to their content, your comment or reaction is visible to anyone who can see the other person's content, and that person can change the audience later."

method does not completely detect the change from "person's" to "someone else's". Also the subtle change in the language used for audience selection of the post is missed. This is due to the fact that we are using only changes in nouns to derive other changes between the sentences. We refer to such a detection as "partial."

Occasionally, changes involve revised URLs or numbers; the method often fails to detect such changes as it is not

tuned for such tasks. We refer to these cases as "empty" detections. In addition, some changes may be detected that are redundant, mostly occurring due to grammatical numbering of a word, presence of compound nouns, special characters like apostrophes, or splitting of a sentence. We refer to such detections as "redundant."

After analyzing multiple examples and identifying the different types of detection, we manually categorized the change detections in each correctly matched sentence pair in the Facebook, Twitter and WhatsApp policy pairs (a total of 4,138 sentence pairs). Table III shows the percentage composition of the different categories. We observe a high percentage of "complete" detections across different policy pairs. This gives us confidence that, for a correct pair of sentences, dependency tree analysis can extract most of the relevant changes between them.

TABLE III
DISTRIBUTION OF CHANGE DETECTION WITH DIFFERENT CORRECTNESS ACROSS MATCHED SENTENCE PAIRS.

Detection category	Percentage (%)
complete	95.4
almost complete	0.8
partial	0.5
empty	0.5
redundant	2.8

E. GitHub track-changes

Amos et al. collected different versions of an organization's privacy policy in a GitHub repository [21], which then allows one to use GitHub's in-built change-tracking abilities to observe the evolution of the policies. We make a few observations with regards to GitHub's change-tracking tool's competency in helping see policy changes.

The GitHub tool performs well if the two policies are organized similarly in terms of sentence locations. Then, for small sentences, the tool manages to highlight changes between correctly paired sentences well. For the following sentence pair from the Facebook 2006 and Facebook 2007 policies,

> "Facebook *will* send you only service-related announcements from time to time through the general operation of the service."

and

> "Facebook *may* send you service-related announcements from time to time through the general operation of the service."

the GitHub tool is able to detect the change from "will" to "may", which is missed in our method.

However, the GitHub tool fails when the line number of a sentence is changed in the new version due to addition or deletion of sentences. Even when the same sentence is present in both versions of a policy, the sentence is treated as deleted in the previous version and a new addition in the new policy version. A cascading effect is observed when a sentence is added or deleted, where all the following sentences get matched with bogus sentences, despite the existence of semantically similar or even the same sentence in the revised version of a policy[4]. This can result in an overwhelming set of changes to parse. We also observe that for larger sentence pairs, the GitHub tool does not generate any change information, and treats the whole sentence as deleted in the previous version and as an addition in the new version.

Sometimes keyword changes that can provide valuable information to users are at risk of being missed by the GitHub tool due to sentence misalignment. For example, the following change is detected in our approach when applied to the linkedin.com 2008A[5] and 2008B[6] policies : *"We also receive the Internet protocol (IP) address of your computer (or the proxy server you use to access the World Wide Web), your computer operating system and type of web browser you are using, as well as the name of your ISP. "* and *"We also receive the Internet protocol (IP) address of your computer (or the proxy server you use to access the World Wide Web), your computer operating system and type of web browser you are using, email patterns, as well as the name of your ISP."*
Such a change, where addition of "email patterns" in the list of collected data is a privacy-relevant piece of information, is completely missed by the GitHub tool as the original sentence is mismatched with a different sentence due to its position in the document. Even if the GitHub tool is used to compare two correct sentence pairs, it finds changes between lines in terms of simple presence or absence of words without using any speech related information. For example, if "Your name will be shared" is replaced with "We share your name" in a policy, GitHub will highlight the sentences as deleted and added, which is redundant when the goal is to present users with minimal yet relevant change information. Pairing rephrased sentences together is hence important, but always treating them as a new addition or deletion defeats the purpose. Natural language processing can overcome such issues, and direct the focus to extracting more meaningful, as well as presentable, changes in a policy.

VI. CONCLUSION AND FUTURE WORK

In this work, we explored a workflow for comparing two versions of a privacy policy by using machine learning and natural language processing, with the objective of enabling users to compare privacy policies at a sentence level and observe the relevant changes. We realized the workflow by first using a sentence classification method to organize policy texts into expert defined policy practice categories. Sentence classification gave us some interesting insights into privacy

[4]https://github.com/citp/privacy-policy-historical/commit/5b9a7fa360 : example mismatches due to alignment

[5]https://github.com/citp/privacy-policy-historical/blob/648f7822e1/l/li/lin/linkedin.com.md

[6]https://github.com/citp/privacy-policy-historical/blob/5b9a7fa360/l/li/lin/linkedin.com.md

policy structures, confirming our concerns raised by the nature of these documents. This is followed by sentence matching, where we used GloVe encoding combined with word mover's distance to match the sentence of a policy to the semantically most similar sentence in the next version of the policy. The method could successfully match sentences that went through restructuring or addition/deletion of terms. In addition, the number of unmatched sentences has been found to be an indicator of potentially large changes in a policy. Lastly, matched sentences were used as input for our change detection method, which detects addition and deletion changes between the sentence pairs. We only explored the change detection in terms of nouns and the context; exploring other parts of speech tags and dependency relationships may also present interesting results in terms of changes.

This paper presents a prototype interface for users to get informed about the changes in a privacy policy through simple highlighting. The change highlight gives a snapshot of the changes to the user that can be easily eye balled, greatly reducing the reading and understanding effort. If a highlighted change seems relevant to a user, the user can read the entire sentence or paragraph containing it. Aside from this, the most important contribution of this work is to demonstrate that, in the world of privacy policies where periodic revisions are fairly common, a change based method can prove useful to the users to be aware of the policy practices without getting overwhelmed by the revisions of the policies. This work presents a new approach towards usable privacy polices and lays the foundation for future works.

Besides enhancing the performance of the change detection, there are many directions to take from here. Methods can be designed to summarize the full body of detected changes and alert users of concerning changes in a policy, preferably based on some preference. The changes can also be classified into severity levels that reflect the impact it will have on a user's privacy. Through proper back-end implementation, a tool can be given the ability to monitor the privacy policies of user selected websites. Future work on identifying contextual changes can provide significant utility as well. Extending such change detection to be performed in-situ for applications in a mobile ecosystem, or a browser, can help provide real-time notifications to a user. Finally, policy comparison across different products of the same organization is another direction.

REFERENCES

[1] I. O. for Standardization, "ISO/IEC 29100 information technology–security techniques–privacy framework," in *Technical report, ISO/IEC 29100:2011(E)*, 2011.

[2] S. Zimmeck, "The information privacy law of web applications and cloud computing," *Santa Clara Computer & High Technology Law Journal*, vol. 29, p. 451, 2012.

[3] A. M. McDonald and L. F. Cranor, "The cost of reading privacy policies," *Journal of Law and Policy for the Information Society*, vol. 4, p. 543, 2008.

[4] L. F. Cranor, "Necessary but not sufficient: Standardized mechanisms for privacy notice and choice," *Journal on Telecommunications and High Technology Law*, vol. 10, p. 273, 2012.

[5] F. Schaub, R. Balebako, A. L. Durity, and L. F. Cranor, "A design space for effective privacy notices," in *Symposium On Usable Privacy and Security*, 2015, pp. 1–17.

[6] J. R. Reidenberg, T. Breaux, L. F. Cranor, B. French, A. Grannis, J. T. Graves, F. Liu, A. McDonald, T. B. Norton, and R. Ramanath, "Disagreeable privacy policies: Mismatches between meaning and users' understanding," *Berkeley Technology Law Journal*, vol. 30, p. 39, 2015.

[7] M. W. Vail, J. B. Earp, and A. I. Antón, "An empirical study of consumer perceptions and comprehension of web site privacy policies," *IEEE Transactions on Engineering Management*, vol. 55, no. 3, pp. 442–454, 2008.

[8] A. Micheti, J. Burkell, and V. Steeves, "Fixing broken doors: Strategies for drafting privacy policies young people can understand," *Bulletin of Science, Technology & Society*, vol. 30, no. 2, pp. 130–143, 2010.

[9] N. Sadeh, A. Acquisti, T. D. Breaux, L. F. Cranor, A. M. McDonald, J. R. Reidenberg, N. A. Smith, F. Liu, N. C. Russell, F. Schaub, and S. Wilson, "The usable privacy policy project," in *Technical report, CMU-ISR-13-119*. Carnegie Mellon University, 2013.

[10] F. Liu, S. Wilson, P. Story, S. Zimmeck, and N. Sadeh, "Towards automatic classification of privacy policy text," in *Techical Report CMU-ISR-17-118R and CMULTI-17-010*. Carnegie Mellon University, 2018.

[11] R. Ramanath, F. Liu, N. Sadeh, and N. A. Smith, "Unsupervised alignment of privacy policies using hidden markov models," in *Annual Meeting of the Association for Computational Linguistics*, 2014, pp. 605–610.

[12] W. Ammar, S. Wilson, N. Sadeh, and N. A. Smith, "Automatic categorization of privacy policies: A pilot study," in *Technical Report, CMU-LTI-12-019*. Carnegie Mellon University, 2012.

[13] S. Wilson, F. Schaub, F. Liu, K. M. Sathyendra, D. Smullen, S. Zimmeck, R. Ramanath, P. Story, F. Liu, N. Sadeh, and N. A. Smith, "Analyzing privacy policies at scale: From crowdsourcing to automated annotations," *ACM Transactions on the Web*, vol. 13, no. 1, pp. 1–29, 2018.

[14] K. M. Sathyendra, S. Wilson, F. Schaub, S. Zimmeck, and N. Sadeh, "Identifying the provision of choices in privacy policy text," in *Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 2774–2779.

[15] H. Habib, S. Pearman, J. Wang, Y. Zou, A. Acquisti, L. F. Cranor, N. Sadeh, and F. Schaub, ""It's a scavenger hunt": Usability of websites' opt-out and data deletion choices," in *Conference on Human Factors in Computing Systems*, 2020, pp. 1–12.

[16] K. M. Sathyendra, A. Ravichander, P. G. Story, A. W. Black, and N. Sadeh, "Helping users understand privacy notices with automated query answering functionality: An exploratory study," in *Technical report, CMU-ISR-17-114R*. Carnegie Mellon University, 2017.

[17] H. Harkous, K. Fawaz, R. Lebret, F. Schaub, K. G. Shin, and K. Aberer, "Polisis: Automated analysis and presentation of privacy policies using deep learning," in *27th USENIX Security Symposium*, 2018, pp. 531–548.

[18] S. K. Cherivirala, F. Schaub, M. S. Andersen, S. Wilson, N. Sadeh, and J. R. Reidenberg, "Visualization and interactive exploration of data practices in privacy policies," in *Symposium on Usable Privacy and Security*, 2016, pp. 3–10.

[19] S. Zimmeck, Z. Wang, L. Zou, R. Iyengar, B. Liu, F. Schaub, S. Wilson, N. M. Sadeh, S. M. Bellovin, and J. R. Reidenberg, "Automated analysis of privacy requirements for mobile apps," in *Network and Distributed System Security Symposium*, 2017.

[20] S. Wilson, F. Schaub, A. A. Dara, F. Liu, S. Cherivirala, P. G. Leon, M. S. Andersen, S. Zimmeck, K. M. Sathyendra, N. C. Russell, T. B. Norton, E. Hovy, J. Reidenberg, and N. Sadeh, "The creation and analysis of a website privacy policy corpus," in *Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 1330–1340.

[21] R. Amos, G. Acar, E. Lucherini, M. Kshirsagar, A. Narayanan, and J. Mayer, "Privacy policies over time: Curation and analysis of a million-document dataset," pp. 2165–2176, 2021.

[22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.