VerHealth: Vetting Medical Voice Applications through Policy Enforcement

FAYSAL HOSSAIN SHEZAN, University of Virginia
HANG HU, Virginia Tech
GANG WANG, University of Illinois at Urbana-Champaign
YUAN TIAN, University of Virginia

Healthcare applications on Voice Personal Assistant System (e.g., Amazon Alexa), have shown a great promise to deliver personalized health services via a conversational interface. However, concerns are also raised about privacy, safety, and service quality. In this paper, we propose `VerHealth`, to systematically assess health-related applications on Alexa for how well they comply with existing privacy and safety policies. `VerHealth` contains a static module and a dynamic module based on machine learning that can trigger and detect violation behaviors hidden deep in the interaction threads. We use `VerHealth` to analyze 813 health-related applications on Alexa by sending over 855,000 probing questions and analyzing 863,000 responses. We also consult with three medical school students (domain experts) to confirm and assess the potential violations. We show that violations are quite common, e.g., 86.36% of them miss disclaimers when providing medical information; 30.23% of them store user physical or mental health data without approval. Domain experts believe that the applications' medical suggestions are often factually-correct but are of poor relevance, and applications should have asked more questions before providing suggestions for over half of the cases. Finally, we use our results to discuss possible directions for improvements.

CCS Concepts: • **Security and privacy → Social aspects of security and privacy**; • **Applied computing → Health care information systems**.

Additional Key Words and Phrases: Alexa, Google-Home, Skill, Medical-voice-applications, policy-enforcement, dynamic-analysis.

ACM Reference Format:
Faysal Hossain Shezan, Hang Hu, Gang Wang, and Yuan Tian. 2020. VerHealth: Vetting Medical Voice Applications through Policy Enforcement. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 153 (December 2020), 21 pages. https://doi.org/10.1145/3432233

1 INTRODUCTION

Voice Personal Assistant (VPA) systems such as Amazon Alexa have entered hundreds of millions of households around the world [63]. Such systems provide various services via the voice interface, ranging from checking the weather and playing music to controlling other smart-home devices. Just like smartphones, Amazon Alexa also supports third-party applications (called "skills") that can be installed/enabled from their app stores[1]. Among different applications, *healthcare applications* have drawn a major attention [47, 50, 53]. Recently, Amazon made a deal with NHS (National Health Service) of the UK, which allows Alexa skills to provide medical information to UK citizens [2]. It is intriguing to provide healthcare services via a personal voice assistant because users can

[1]In this paper, we refer to Amazon's voice applications as skills.

2474-9567/2020/12-ART153 $15.00
https://doi.org/10.1145/3432233

naturally interact with the system via question and answering. As a result, the number of healthcare applications has been increasing in the app stores [14, 74].

Despite the benefits, people have also expressed their concerns due to the sensitive nature of healthcare [44]. For example, various reports have discussed issues regarding data privacy [33, 57, 78], user safety [26], and generally the quality of healthcare services provided by the applications [68, 70]. To ensure privacy and safety, voice assistant platforms also established policies to regulate the healthcare skills published to the app stores. Take Amazon Alexa for example, on one hand, Amazon worked with a few healthcare providers to develop skills that are legally compliant to Health Insurance Portability and Accountability Act (HIPAA) [1]. These skills would allow patients and caregivers to manage healthcare data at home through Alexa [47]. On the other hand, the vast majority of skills are not HIPAA-compliant, and Amazon has designed platform-specific policies to restrict their actions. Some of the policies mimic the HIPAA requirements, for example, by regulating how user data is collected and stored, and other policies try to prevent skills from advertising drugs or providing an imprudent medical diagnosis.

In this paper, with a focus on Alexa voice assistant, we aim at systematically analyzing how well healthcare skills are compliant to privacy and safety related policies. While existing researchers have started to look into this problem [26, 32, 75], these initial efforts are limited to *manual analysis* on a handful of skills (*e.g.*, 20 skills). Meanwhile, based on our measurement, there are already nearly 900 healthcare skills in the Amazon app store as of 2019, and the number is still growing. As such, our goal is to develop a set of tools to facilitate automated analysis of the behaviors of applications on a large scale.

Challenges. There are key challenges to automatically detect and reason violations for voice assistant skills. *First*, skills have no executable files or source code to analyze their behaviors. Instead, their key function logic is hosted in the cloud as a black-box. This is fundamentally different from traditional mobile apps whose client-side code is available for analysis. As a result, traditional static or dynamic analysis methods [55, 82] cannot be applied to skills. *Second*, to detect policy violations, we need to generate the appropriate testing inputs (*e.g.*, leading questions) to trigger the unwanted behaviors. The challenge is that the violation does not always occur during the first interaction. Researchers often have to manually interact with the skills back-and-forth to trigger them. To automate this process, our tool needs to generate allowable inputs to the target skill, and dynamically change the inputs based on the skill's response. *Third*, different violations may have different implications. To reason and interpret the potential violations, we also need the inputs from domain experts in medicine and healthcare.

Our Approaches. We design `VerHealth` to examine whether a given skill has followed the predefined policies. `VerHealth` has a *static module* and a *dynamic module*. The static module analyzes the skill's web page to detect two types of violations: skills that provide life-saving assistance (prohibited by Amazon), and skills that fail to include disclaimers. The dynamic module aims to trigger potential violations by dynamically interact with the skills. More specifically, our tool asks carefully designed questions, analyzes the skill's responses, and then ask more follow-up questions. Through multiple rounds of interactions, we check whether a skill stores users' physical and mental health information without formal consent which violates Amazon's policy; whether the skill performs medical diagnosis and provides misleading information; and whether the skill advertises the black market sale of prescription drugs. Ground-truth evaluation shows both modules are highly accurate with accuracy over 97% (F-1 scores are 90.62% and 97.08% respectively). To assess the potential violations, especially regarding skills that provide medical diagnosis and advice, we consulted medical school students (domain experts).

We applied `VerHealth` to 813 healthcare Alexa skills. We in total tested 855,276 probing questions and analyzed 863,988 responses. We observed that policy violations were fairly prevalent among health-related skills. For example, out of 813 skills, we detected 244 skills (30.23%) storing user physical and mental health information, which violates Amazon's policy. Some of the skills even store the health-related information in their database for a long term. In addition, a large number of skills (86.36%) skills have missed the required disclaimers. Finally,

we show the medical advice provided by health-related skills is usually in poor quality (or of a low relevance), which are confirmed by domain experts.

Contributions. Overall, our results suggest there are still big gaps to fill before we can build usable voice assistant applications for healthcare services. We also point out possible future directions regarding enhancing privacy and improving application usability at the end of the paper. In summary, we make three main contributions.

- **New Tool.** We developed `VerHealth` to perform automated tests on healthcare voice applications. We shared our tools and the labeled datasets with the research community to facilitate future research [2].
- **New Measurements.** We applied `VerHealth` on 813 health-related applications in Amazon Alexa and detected various policy violations, which revealed the current state of the healthcare skills on Amazon.
- **Qualitative Assessment.** We consulted with domain experts (medical students) to confirm the violation and interpret the potential impact on users.

2 BACKGROUND AND RELATED WORKS

We start by introducing the background of the Alexa voice assistant system, and the policies on healthcare applications. Then we discuss key related works to ours.

2.1 Background of VPA and Policies

Amazon Alexa is the most popular VPA system [21] and it uses a cloud-based model to host skill applications. It works as the following: a user can give a voice command to the VPA device, and then the voice command is sent to the cloud. The cloud translates the natural language command into an API call and routes the command to the corresponding skill servers. The skill servers then generate a response or take actions according to the user command. Amazon Alexa maintains a "skill store" where each skill has its web page. The web page shows the skill description, developer information, user reviews, and the supported voice commands. Note that Amazon only allows up to three voice commands listed on the voice command section. Typically, developers would list commands that represent the skill's key functionality. If the skill has more commands, developers may also list them in the skill description.

Alexa Policies on Healthcare Skills. To ensure privacy and safety, Amazon Alexa has established policies to regulate the healthcare skills. There are two types of skills which are regulated differently.

First, Amazon has worked with a few healthcare providers to develop dedicated skills to handle sensitive medical data. These skills are required to be compliant with HIPAA [1]. HIPAA is a federal law and a national standard to protect sensitive patient health information. It introduces guidelines and imposes regulations on sharing and accessing patient's protected health information such as an individual's physical or mental health condition, the provision of health care to the individuals, and payment for the provision of health care to the individual. So far, there are 6 HIPAA-compliant skills (Express Scripts, Cigna Health Today, My Children's Enhanced Recovery After Surgery, Livongo Blood Sugar Lookup, Atrium Health, and Swedish Health Connect) on Amazon, which allows patients, caregivers, and health plan members to manage healthcare data through Alexa [47]. When installing a HIPAA-compliant skill, users need to give consent to share their health data with the healthcare providers. As of August 2020, Amazon officially establishes a HIPAA-Eligible Skill Program [43], allowing HIPAA covered entities and business associates to apply to develop HIPAA-compliant skills.

Second, the vast majority of healthcare skills are not HIPAA-compliant. Amazon has designed platform-specific policies to restrict their actions to avoid violations of HIPAA and protect user privacy. As of August 2020, Amazon's policies [7] on healthcare skills can be summarized into 5 items as shown in Table 1. More specifically, (P1) skills are not allowed to collect any physical or mental health information (*e.g.,* patient's disease, medication,

[2] https://github.com/faysalhossain2007/verhealth

Table 1. Amazon's policies on health-related skills. If any skill exhibits the listed characteristics, Amazon is supposed to remove the skill from the app store.

Policy	Description (originally as mentioned in website [7])	Category	ID
Storing Health Data	*"Collects information relating to any person's physical or mental health or condition, the provision of health care to a person, or payment for the same."*	Behavior	(P1)
Life-saving Assistance	*"Claims to provide life-saving assistance through the skill or in the skill name, invocation name or skill description"*	Behavior & Description	(P2)
Misleading Information	*"Contains false or misleading claims in the responses, description, invocation name, or home card regarding medicine, prescription drugs or other forms of treatment. This includes claims that a treatment can cure all diseases or specific incurable diseases. A claim can be misleading if relevant information is left out or if it suggests something that's not true."*	Behavior	(P3)
Prescription Drugs	*"Provides information about black market sale of prescription drugs."*	Behavior	(P4)
Disclaimer	*"Is a skill that provides health-related information, news, facts or tips and does not include a disclaimer in the skill description stating that the skill is not a substitute for professional medical advice."*	Description	(P5)

blood group, mental state). (P2) Skills are not permitted to provide (or claim to provide) life-saving assistance considering the potential risks. In particular, skills should not contact emergency responders (*e.g.,* calling "911"). Because regulatory rules require the device to call "911" must have the ability to receive incoming calls [39]. Whenever a skill makes a claim or provides the feature of calling emergency responders, then it violates this policy. (P3) Skills are not allowed to mislead users by providing incorrect information (regarding diagnosis and medical advice), considering the potential risks to users' safety [48, 71]. (P4) Skills should not inform users of the black market sale of prescription drugs. Given that prescription drugs are available in the black market [40], providing such information would be considered as a crime [36, 38]. (P5) Skills providing medical information need to include a disclaimer in its description. Disclaimers are necessary to make the users aware that the skills are not alternatives to real doctors. This policy is likely there to protect the device manufacture (*i.e.,* Amazon) from legal liability. All the non-HIPAA-compliant skills should follow the above-mentioned policies. If a skill fails to follow any one of the policies, then it is subjected to removal from the store [7].

Motivating Examples: Skills with Policy Violations During our manual investigation, we found skills that have violated one or more Alexa Policies, which should have been taken down from the store. Regarding P1, we found some skills storing user medication information, *e.g.*, "*Insulin Calculator*" stores blood glucose level, and "*Vaccine Buddy*" stores user vaccination date. Since none of these skills are HIPAA-compliant, storing physical or mental health data is prohibited by Amazon Alexa. Regarding P3, we found that Skills such as "*Dr. A.I. by HealthTap*" and "*Medical Assistant*" gave inaccurate or irrelevant medical information (more examples in Section 5.3). Regarding P5, "*EnT-Tips*", "*Health Facts*", and "*Fat Loss Tips*" provide health and disease information, but none of them have a disclaimer. In this paper, our goal is to build tools to detect such violations automatically.

2.2 Related Works

Next, we discuss how our work is related to and different from existing works on voice assistant analysis and healthcare applications.

Healthcare Applications Using Voice Assistant & IoT devices. Researchers have developed various *voice-based* healthcare systems. These include dialogue systems for patients with chronic pain [54], and systems for automatic voice pathology detection [45] and voice disorder detection [24]. Recently, researchers also explored to use data from IoT devices (Fitbit and smart-phones) to predict health risks [31], track flu spreading [30], diagnose and monitor Parkinson's disease [51], predict and track social anxiety and depression [72, 80], sense mental stress and compound emotion via smartphone sensing [85, 86], and facilitate the collection of nursing care records [46]. Similarly, voice personal assistants have been used for healthcare purposes too. For example, Ahmed *et al.* used VPA for early dementia detection [22]. Maor *et al.* analyzed the voice pattern to predict hidden heart disease [62].

While the above efforts are focused on leveraging the functionality of VPA and IoT systems to develop novel healthcare applications, other researchers focus on existing applications in VPA stores to study their usability and potential risks. For example, Bickmore *et al.* performed a user study where 54 participants used voice assistant to get information for corresponding medical problems [26]. Analyzing the responses, they concluded that reliance on such information exhibits a safety risk for patient health. Our work follows this direction to assess existing applications with respect to their compliance with privacy and safety policies. Compare to the existing work, we scale-up the analysis by developing automated tools that are customized to compliance checking.

Privacy in Voice Personal Assistant. Recent works have investigated the privacy concerns about voice assistants among users [33, 52, 57]. For example, researchers looked into the privacy implications of voice assistant devices regarding their "always-in-listening-mode" [78], and uttering the private information in public space [64]. Recently, Guo *et al.* investigated voice skills that ask user private information [41]. Comparing to this concurrent work, we focused on a broader set of policies (beyond data privacy) for healthcare skills (*e.g.*, life-saving assistance, misleading information, advertising illegal drugs, missing disclaimer).

NLP-based Security and Privacy Analysis. Our paper is also related to existing works that adopted Natural language processing (NLP) techniques for security and privacy analysis. For example, prior works have used NLP methods to analyze and summarize privacy policies [42], measured the trustworthiness of skill certifications [28], and reduce the ambiguity in privacy specifications [58]. For example, Zimmeck *et al.* focus on finding inconsistencies between an application's actual functionalities with the disclosed privacy policies [87]. However, none of the previous works can be applied to analyzing voice skills due to the unstructured nature of the skill's responses. Shezan *et al.* measures the attack surface of the skills by identifying the sensitive voice commands using active learning and keyword-based approach [73]. Unlike [73], our goal is not to locate sensitive commands, but to identify policy violations.

3 SYSTEM DESIGN AND IMPLEMENTATION

Next, we introduce our design of `VerHealth` to detect policy violations in health-related skills. As shown in Table 1, there are two types of policies: description-based and behavior-based. For description-based policies (such as P5), `VerHealth` looks for potential violations by statically analyzing the skill descriptions. For behavior-based policies (such as P3), `VerHealth` looks for potential violations by dynamically interacting with the skills.

To develop `VerHealth`, we have three major challenges. *First*, unlike smartphone apps whose binaries (or source code) are available for analysis, VPA skills are essentially web programs behind the (Amazon) cloud. We cannot characterize the skill's behavior using traditional API analysis. *Second*, the inputs to trigger sensitive behaviors are usually sparse. To reveal the undocumented behaviors hidden behind several rounds of interactions, we need to parse the skill's responses, and dynamically generate the next rounds of inputs accordingly. *Third*, the voice commands supported by VPA skills are typically short, which provides little information to run conventional NLP tools. In the following, we design a static module and a dynamic module to overcome these challenges.

3.1 Static Module

The static module is designed for handling description-based policies (P2 and P5). As is shown in Fig. 1, we first collect the static content of a skill (*i.e.,* title, skill description, invocation name) from skill's web page in the Alexa store. For P2, we aim to detect statements that the skill provides life-saving assistance. We use a rule-based method because statements related to "life-saving assistance" usually have clear patterns [84]. The rule-based method is also self-explanatory, which makes it easy to validate the detection result. For P5, we could

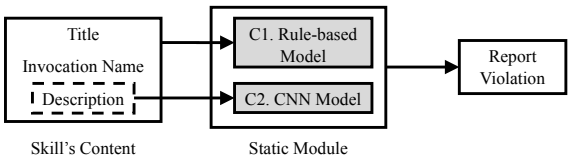

Fig. 1. Static Module contains (C1) a Rule-based model to detect P2-violation, and (C2) a CNN to detect P5-violation.

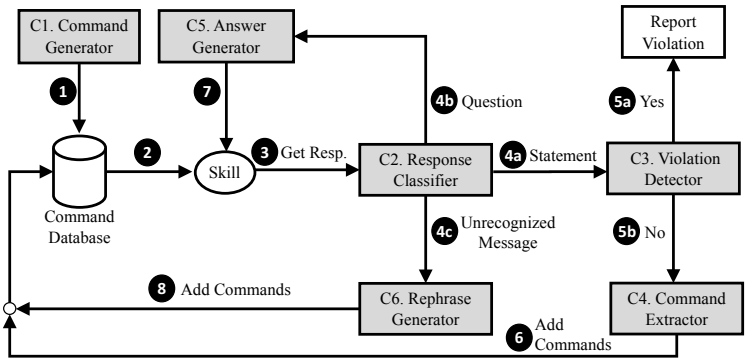

Fig. 2. System architecture of the dynamic module. **Step-1**: generate probing commands using (C1) Command Generator. **Step-2**: send a command to the skill using Alexa Testing Interface (Section 3.2). **Step-3**: use (C2) Response Classifier to classify the response according to Table 2 into three types: Statement, Question, Unrecognized Message. **Step-4a**: (C3) Violation Detector determines whether the skill violates the policy listed in Table 1. **Step-5a**: report a violation if found; **Step-5b**: otherwise, (C4) Command Extractor extracts the embedded voice command (if any) and add it to command database in **Step-6**. **Step-7**: if the response is a Question, then (C5) Answer Generator will respond with an appropriate answer. **Step-8**: (C6) Rephrase Generator will rephrase the command if the skill cannot understand the original question.

not use simple rules to detect disclaimers because the semantic context is more complicated, and it is not easy to find common keywords. For example, the three disclaimers below are very different in their sentence structures and keyword choices: "*1) If you need nutritional information due to a health issue, please do not depend on this skill just yet*"; "*2) This service is not meant as a substitute for medical care*"; "*3) This tool does not provide medical advice, and is for informational and educational purposes only*". To this end, we use a CNN (Convolutional Neural Network) model to detect if a sentence is part of a disclaimer. We choose CNN because previous works [49, 56] performed well on similar text mining tasks.

3.2 Dynamic Module

The dynamic module is designed for behavior-based policies (P1, P3, P4, and partially P2). By interacting with skills like a user, we aim to trigger the violation behaviors of skills. Figure 2 depicts the overall design, which includes 6 main components– (C1) Command Generator, (C2) Response Classifier, (C3) Violation Detector, (C4) Command Extractor, (C5) Answer Generator, and (C6) Rephrase Generator. We explain their designs as follows.

C1. Command Generator. To trigger violation behaviors of skill, we cannot just use the example commands on the skill's web page. This is because the number of example commands is limited to three, and the skill may have many undocumented commands. To increase the chance of revealing violations, we generate two types of testing commands: 1) *Passive probing commands*: voice commands the skills might understand, which helps to start the conversation; 2) *Active probing commands*: voice commands that might trigger sensitive behaviors.

For passive probing commands, we construct them from two information sources: (a) the recommended/example commands on the skill's web page, and (b) commands embedded in the skill description. In total, we have 2,178 passive voice commands from 813 skills (details in Section 4).

For active probing commands, we craft test inputs that are directly relevant to potential policy violations. The active probing commands would be different for different policies. For example, to detect violation of P1 (storing medical data), we first identify a list of personally identifiable information and common medical information [13, 59, 65]. Then we craft voice commands that give/send such medical information to the target

skill. One example command could be "my due date is August 2nd, 2020". Then we craft corresponding commands to ask "what is my due date?" to check if the medical information is permanently stored by the skill.

Using the above methodology, we can generate over 1 million active probing commands (see Section 5.3), which could take a long time to finish the testing. To reduce the number of active probing commands, we run a preliminary test (among a small number of skills) and select top m commands that can trigger the largest number of unique responses.

Table 2. Rules for classifying a response into three different classes.

Response Type	Rules
Unrecognized Message	I don't understand/ Repeated response/ Audio only response/ I am having trouble understanding/ I could not find/ I don't know/ I do not have/ I cannot understand/ Unable to find
Question	"wh"-questions/ "yes-no"- questions/ "choice"-questions [34]
Statement	Response not satisfying Unrecognized Message & Question rules

C2. Response Classifier. During our manual investigation, we find that certain violations are hidden behind several rounds of interactions. To reveal them automatically, we need to truly interact with the skill. For this reason, we develop a response classifier to analyze the response from the skill, and help `VerHealth` to determine the next move (as shown in Figure 2). We classify skill's responses into three classes– statement, unrecognized message, and question. If a skill successfully recognizes the input voice command, the skill will provide an answer or perform certain actions. The corresponding response is in the form of *statement*. Sometimes, the skill would ask further *questions* to proceed with the user's request. If the skill fails to recognize the voice commands, it will respond with *unrecognized messages* such as "sorry, I don't know that".

We use the rules in Table 2 to classify responses. We have tested more complicated NLP methods, and they are not necessarily outperforming our rules, possibly due to the short length of skill responses. Given a response, we first check if it is an unrecognized message, which usually has clear patterns (*e.g.*, "sorry", "I don't understand"). If not, then we check if the response starts with "wh"-questions, "yes/no"-questions, or "choice"-questions [34]. If a particular response does not fit in the above two-classes, we mark them as "statement".

Validation Experiment: To evaluate the performance of our response classifier, we manually labeled 250 responses from 189 random health-related skills. The result shows our response classifier is accurate: it only makes the wrong prediction for 1 out of 77 unrecognized messages, 5 out of 108 statement type responses, and 6 out of 65 questions type responses. The overall F1 score is 95.2%.

We have also compared the rule-based method with a more sophisticated NLP method. More specifically, we use a pre-trained CNN classifier on the *switchboard* corpus [76] which has 1,155 spontaneous human-to-human telephone conversations. We tried to transfer classifier to our domain, but the result is not as good as rule-based methods, as shown in Table 3. As such, we use the rule-based method for response classification.

Table 3. Performance comparison of our Response Classifier with the baseline.

Model	Accuracy	Precision	Recall	F1-Score
Baseline [76]	67%	53%	66.8%	56.9%
Our method	95.2%	95.4%	95.2%	95.2%

C3. Violation Detector. This component is responsible for detecting the violation of P1 – P4. As shown in Figure 2, the violation detector will analyze all the statement type responses to explore possible violations. For example, for P1, if we can successfully retrieve the medical/personal information that we previously shared with the skill, we can confirm the skill has stored such information (*i.e.*, violation). For P2, if a skill provides or claims to provide life-saving assistance through voice interaction, then we mark it as a violation. For P4, we check if a skill provides any sale information of the black market stores for the prescription drugs. The most difficult part is to determine P3 — inaccurate or misleading medical information, for which we consult with domain experts. The detailed approaches to detect these violations will be presented in Section 5.

C4. Command Extractor. Sometimes a skill's response would contain suggested commands. Our Command Extractor aims to extract these suggested commands so that we can use them to continue the conversation with the skill. For example, during the interaction with the skill, the skill may respond with "Please say yes", or "Please say set my weight". We extract suggested commands by following three steps: 1) split the response by "full stop" or ".", 2) search for the sentence which has "ask", "say" or "tell", and 3) list the part of a sentence which starts with "ask", "say" or "tell" as the command. For example, given a response "welcome back to my pregnancy from baby center. If you know your due date, you can say, set my due date.", we can use the described approach to find the suggesting command: "set my due date".

Validation Experiment: To assess the performance of the Command Extractor, we randomly select 200 responses from 168 skills, where 51 responses contain embedded voice commands. Our command extractor successfully extracted 48 embedded voice commands with good performance (95.5% Accuracy, 88.88% Precision, 94.12% Recall, 91.43% F1-Score).

C5. Answer Generator. Our Answer Generator is responsible for providing the appropriate answers for question type responses. Note that, by *appropriate*, we meant those answers can be recognized and processed by the skill. Generating answers from a question is still an open research problem in NLP [77]. We adopt a simple and yet effective knowledge-base approach: 1) we collect all the responses classified as questions; 2) we create a knowledge base (*i.e.*, a dictionary) for those questions with our pre-compiled answers.

Whenever we encounter questions that match the keywords of existing entries in our dictionary, we provide the corresponding answers. For example, if a skill asks about the user's height, we will respond "*five feet*" as the answer. The reason why we use the knowledge-base method is that the questions from skills are usually focused on a small set of questions about user attributes and common configurations (*e.g.*, height, gender, zip code). It is easier to prepare the answers ahead of time, compared to using NLP tools. We indeed tested tools trained on Squad [69] for generating answers, but the performance is worse than our knowledge-base method.

Validation Experiment: We test 200 question type responses from 111 skills. We find that Alexa can successfully recognize 174 of the responses generated by our method (success rate 87%). We also compare our performance with Squad (baseline) and find the success rate is near 0%. The reason is Squad is trained for general-purpose conversations, which cannot precisely answer questions from skills.

C6. Rephrase Generator. When a skill does not understand our voice command, it responds with the "unrecognized message". As shown in Figure 2, we would give "unrecognized messages" another chance by sending them to a Rephrase Generator. It is possible that the reason why the skill cannot recognize the voice commands is that the skill is expecting different wordings. Our Rephrase Generator works as follows: 1) we check whether the command is an original command in our command database; 2) if it is an original command, then we generate ten rephrase commands using [81]; 3) Otherwise, move to the next original command in our database.

To rephrase the command, we start with replacing synonyms of the words inside a sentence using Spacy. We find that this simple approach does not work: none of the rephrased sentences in our test can be recognized by the skills. Later, we use Wieting *et al.* model [81] to generate the rephrased commands. We select 0.8 as the sampling threshold, which produces good results.

Validation Experiment: To evaluate our Rephrase Generator, we select 200 commands from 178 skills for which we get the unrecognized message from the skills. For each of those commands, we generate ten rephrased commands using our Rephrase Generator. If the skill can recognize any of those ten commands, then the Rephrase Generator is helpful for that case. We find that out of the 200 cases, the skills can recognize our rephrased commands for 188 of the cases (success rate 94%). We have manually investigated the 12 failed cases. We find that (very likely) these skills do not have the built-in functionality to respond to the 12 voice commands, which is beyond the capability of rephrasing.

Implementation Dynamic Testing. We build a dynamic module using Amazon's testing interface [20]. Amazon's testing interface works like a real Alexa device except for a few aspects (*e.g.*, Alexa setting API, Alexa reminder API, and character display) which will not affect our evaluation. We can use this debugging console to interact with all the skills available in the store. We interact with skills through this console as following: 1) Start a skill by mentioning the skill's invocation name (*e.g.,* "open WebMD"); 2) Send a command; 3) If the skill understands the command and has the corresponding data, it will provide the answer; 4) Otherwise, it will notify the user that it cannot answer the question. 5) After completing the dynamic testing steps depicted in Figure 2, we can terminate the interaction by giving an exit command (*e.g.,* "stop", "exit").

End-to-End Validation of the Dynamic Module. We want to check whether the dynamic module can indeed trigger more responses from the skills. As a small end-to-end evaluation, we randomly select 25 skills and interact with them by sending ten commands to each skill. In total, there are 250 rounds of interaction triggered. Then we activate all the components in the dynamic module of `VerHealth` to interact with those 25 skills. This time, we can trigger 344 rounds of interactions, which means 94 (32%) more rounds of interactions triggered compared to the previous setting. This confirms the benefit of the dynamic module.

4 DATA COLLECTION

To evaluate the policy violations of the skills, we first built a dataset by automatically collecting skill information from Amazon Alexa's app store [5]. For each skill, we obtained the skill information (including- skill name, category, description, account linking details, developer information, user review, category information, up to three recommended voice commands) from the U.S. store in May 2019. To reach the introduction page of each skill, we built a crawler that ran a breadth-first search on skill-IDs. More specifically, the crawler started with the homepage of the "Health & Fitness" category in Alexa's skill store where the medical skills are listed. The crawler then visited the link of each of the listed skills. On each skill's page, there was a section called "Customers have also enabled", which showed a list of highly related skills. The crawler then looked for any *new* skill-IDs that also belong to "Health & Fitness". The crawler repeated the process unless there was no more new skill-ID showing up. In total, we get 813 health-related skills.

As contexts, healthcare skills only take a small portion of all skills in the store. In May 2019, we used the above methodology to crawl all 23 skill categories, and got 31,413 skills in total. As we mentioned, we focus on health-related skills for its high sensitivity nature.

Extracting Additional Voice Commands from Descriptions. Amazon Alexa only allows developers to list up to three voice commands on their skill page [5]. For skills with more than three commands, we found that developers often include additional voice commands in the skill description as a *list* or in a *double quote*. To extract voice commands from the skill description, we followed three steps: 1) We first pre-processed the text of the skill description following the standard NLP practice [61], such as converting all the words to the lowercase format, converting Unicode objects to ASCII strings, and removing special characters. 2) We divided the description (based on a line break, newline, and double-quote) into different chunks. 3) We marked each chunk as a voice command, if the text chunk starts with the *invocation word* (*i.e.*, "Alexa,"). In this way, we extracted 80,129 voice commands in total from 813 skills. This covers all the skills we can find on the public app store under the "Health & Fitness" category as of May 2019.

5 POLICY VIOLATION EXPERIMENTS, RESULTS, AND EVALUATIONS

In this section, we ran a series of experiments by using `VerHealth` to detect policy violations. We performed the end-to-end evaluation to examine `VerHealth`'s performance (*e.g.*, the accuracy of detecting violations). In addition, we performed an in-depth analysis of the detected violations. We performed all the experiments on a Desktop PC with 16 GB of RAM and 3.1 GHz Intel Core i5 processor, running windows 10.

5.1 P1: Skills Should Not Store Medical Information.

We used our dynamic module to investigate the violations. Note that HIPAA-compliant skills are allowed to store medical information, so we excluded the six HIPAA-compliant skills from our analysis.

Active Probing Commands. We generated probing commands from the common health conditions from medical literature [13, 59, 65]. Note that, we needed to generate two types of commands: (1) storing commands: store medical information (*e.g.,* I have diabetes), (2) querying command: retrieve that information (*e.g.,* do I have diabetes?). We created 12 pairs (24) of probing commands based on a small-scale test: we identified the commands with the highest numbers of unique responses. Initially, we selected 26 physical and mental health conditions according to healthcare regulations [13, 59]. All of these conditions are flagged as protected health information according to HIPAA [23]. We then crafted a command pair (storing command and querying command) for each condition to check whether a skill store that information or not. As a result, total computational overhead for testing all 813 skills with all that 26 conditions would be expensive and time consuming. To perform the experiment efficiently, we sent 26 pairs (52) of probing commands only to the top 20 (sorted by Amazon's featured rank) [5] skills and count the valid unique responses (any response except unrecognized message). Later, we filtered out the top 12 pairs (24) of probing commands for which we get the most number of unique responses. As from the 13th pair, we experienced a significant drop in the number of unique valid responses.

Violation Detector. `VerHealth` provides the physical and mental medical information (*e.g.,* my blood group is b positive), then tries to retrieve that information (*e.g.,* what is my blood group?). If `VerHealth` retrieves the information from the skill's response (*e.g., b positive*), then it means the skill stores the medical information. Note that, a malicious skill may store information but do not respond to the question when asked. Such hidden activity is not possible to detect without access to skills' servers (even Amazon and Google don't have access to them). Thus we reported the violations we identified given the practical constraints.

Detection Results. `VerHealth` detected 244 (30.23%) skills violating the medical data collection policy. Note that the six HIPAA complied skills are allowed to collect medical information, so we excluded them when reporting violations. To evaluate the effectiveness of `VerHealth` for detecting violations of P1, we manually labeled 250 responses from 200 skills to evaluate the performance of `VerHealth` in terms of detecting storing behavior of the skills. Among those interactions, 53 interactions indicate storing behavior. `VerHealth` achieved a good performance with 98.8% accuracy, 100% precision, 94.33% recall, and 97.08% F1-score.

Duration of Information Storage. Now, we want to reveal for how long does the skill remember information (Q1). We can divide the skill's memory into three categories – (1) Short term, (2) Long Term, and (3) Permanent. We selected 244 skills which violate (P1), and ran the following automatic experiments.

Short Term: Skills with short term memory will remember the information as long as it actively interacts with the user (after the user opens the skill and before the user quit the skill). If a skill only has short term memory, the skill will erase the information whenever users stop interacting with the skill. To run this experiment, we followed this approach– (1) Send storing commands to Alexa. (2) Terminate interaction by *exit* command. (3) Send querying commands to retrieve information. (4) If the violation detector detects as a violation of (P1), then mark the skill as at least has short term memory. *Long Term:* Skills with long term memory will remember the information even after disabling the skill. For testing this behavior, we follow this approach – (1) Send storing commands to Alexa. (2) Terminate interaction by *exit* command. (3) Automatically disable the skill using python web script. (4) Send querying commands to retrieve information. (5) If the violation detector detects as a violation of (P1), then mark the skill as at least has long term memory. *Permanent:* For permanent memory, our intuition was that a skill can remember the stored information as long as the device is linked to the user's Amazon account. We investigate it by the following steps– (1) Send storing command to Alexa. (2) Terminate interaction by *exit* command. (3) Restart the device with command and link another account. (4) Send querying

command to retrieve information. (5) If the violation detector detects as a violation of (P1), then mark the skill as a permanent memory. Note that VerHealth can launch all these activities automatically (except setting up the accounts in the beginning).

From Table 4, we can observe that all the 244 skills (100%) have short term memory. We found 86 skills (35.25%) remembering the information even after disabling it. We didn't find any skills with permanent memory.

Table 4. Duration of Storing Medical Information in Alexa Health Skills.

Duration	#Skill
Short Term Memory	244
Long Term Memory	86
Permanent Memory	0

5.2 P2: Claim to Provide Life-Saving Assistance

P2 is a hybrid policy, that includes regulation both for descriptions, and behaviors. As a result, we first used the static module to identify the violation of P2 in skills' title, invocation name, and description. Then, we used our dynamic module to interact with the skills. Throughout our analysis, we did not find any violations. We include our analysis here for completeness.

To identify the life-saving assistance claims, we first explored the pattern of such claims and the representative keywords by analyzing the current store [8, 11] and articles [25, 83]. To detect representative keywords in skills' information, we populated a keyword set, which includes - '911', 'emt' (Emergency Medical Technicians), 'first responder', 'responder', 'call', 'dial', 'emergency', 'contact', 'sos', 'life-saving', 'assistance', 'technician'. We built our rule-based approach using NLP POS (part-of-speech) tagger [60] to identify such a claim inside the text. We included three patterns (in total we have seven patterns) and the corresponding examples in Table 5. We used these patterns to identify such claims both in the skill's description (description category) and introduction message of the skill (behavior category). Also, we crafted four voice commands (*i.e.,* "call 911", "contact fire service") according to [25, 83] to check whether the skill actually provided such functionality without disclosing.

Table 5. Rules for identifying violations of P2. <> indicates keywords; (:parent) & (:siblings) indicates relations among tags.

Pattern	Example
{VP (:parent) -> { {VB <save, rescue>} (:siblings) {NP <life, health>}} } (:siblings) {NP <emergency>} }	"This emergency feature on your device could save your life"
{VP (:parent) -> { VB <text,sms> (:siblings) NP {NN <location, position, area>} }} (:siblings) {VP (:parent) { {VBP <trigger, initiate, stimulate>} (:siblings) { NP { {CD <911>} } }}}	"The App will also text multiple family members your location when you trigger a 911 call"
{ADVP - { RB <automatically, directly> } } (:siblings) {VP (:parent) -> {VBZ <sends> (:siblings) NP {NN <emergency, location }> (:siblings) PP {CD <911>} } }	"This app also has clever a feature that automatically sends your digital emergency card to 911 operators the minute you call for help"

We didn't find any violations for P2 from all the skills. We investigated the results to confirm if they are reliable. We first manually checked the description of the top 50 featured skills – we did not find any P2 violation. Also, we manually interacted with those skills to ask for life-saving assistance. In total, we examined 242 (200 active probing, 42 passive probing) voice commands. Consistent with VerHealth, we didn't find any relevant claims.

5.3 P3: Provide Wrong Medical Information.

Since this is a behavior-based policy, we ran our dynamic module to interact with the skills, and collected the skill's suggestion on various diseases/symptoms. To judge the correctness and the impact of the medical information, we verified those responses by consulting with medical school students (domain experts).

Active Probing Command. We collected symptoms from a healthcare dataset [19], which contains 388 unique symptoms. We selected 100 most common symptoms, based on diagnosis count [19]. For each symptom, we created probing commands using 6 templates according to what users normally ask Alexa for medical

advice [37]. For example, "what to do for X", "what is the medicine for X", where X represents the name of the symptom. The 6 templates are selected based on a small-scale test: we identified templates that triggered the highest numbers of unique responses. In total, we had 600 (6 x 100) active probing commands to test 813 skills.

Violation Detector. For detecting whether a skill provides wrong medical information, we tried several approaches. First, we checked whether, for the same symptom, two different skills provide two different conflicting suggestions. To do that, we used the POS (part-of-speech) tagger [60] to extract subject (NN-singular noun, NNS-plural noun, NP-singular proper noun, NPS-plural proper noun), action (VB, VBZ, VBG), value (QP-quantifier phrase, CD-cardinal number). By adapting Preclude's [67] solution, we investigated the conflict. Unfortunately, this approach didn't work because of two reasons - (1) we got the same responses for most of the symptoms, which might be due to the similar databases and natural language processing techniques the skills use (2) all the responses were complex, each containing more than one-word representing subject and action. So, it became very difficult to extract the suggestion correctly. Second, we compared skill's suggestion with the website's (*e.g.,* WebMD (https://www.webmd.com/), MayoClinic (https://www.mayoclinic.org/)) suggestion for the same disease. However, it is very difficult to extract useful suggestion from the website because the responses are long and complex [29]. It was very challenging for us to decide the accuracy of the information found on WebMD, especially when there could be a lot of potential medical reasons behind the symptoms.

Therefore, we asked medical students to verify the skill's suggestions. To do that, we recruited three medical students to validate those suggestions. It is difficult and expensive to consult domain experts. That's why we were only able to manage three medical students for assessing the quality of the suggestions.

Detection Results. In total, we collected 562,302 responses from 813 medical skills. We found that many skills give the same responses for the same symptoms, possibly because they rely on the same information sources or medical databases. The duplication rate was extremely high: after removing the exact same responses, we got 260 unique responses. This result has an important implication — if some of the 260 responses are incorrect or misleading, they can affect a large number of users as users can encounter these responses via different skills.

Removing Non-Medical Information. While checking these 260 responses, we found that many responses were not related to medical information. Instead, they guide how to use the skill (for example, when asked "what to do for low back ache", one skill replied "awesome, you can start by saying which day you want to start."). Even non-medical expert would understand the answer is not a medical suggestion. To save the domain expert's time, three computer science researchers (who are not medical experts) manually investigated those 260 responses. They annotated responses into two classes– *suggestion* and *not suggestion*. Whenever there was a conflict among the annotators, they resolved that by taking the majority vote. The labels were consistent across these responses (agreement rate = 96.15%, Fleiss' kappa = 0.944) [35]. In this way, we removed 170 out of 260 responses and kept the other 90 responses that were labeled as medical suggestions for the assessment by domain experts.

Assessment by Domain Experts. Then we uploaded those 90 medical suggestions (31,725 total responses before removing redundant ones) along with assessment questions from skills to *Qualtrics* (https://www.qualtrics.com/) to invite medical students for evaluating these suggestions. We recruited via a mailing list of medical students at our institution. We paid each of them with $100 Amazon Gift Card for completing all the 90 questions. Once they finished reviewing the quality of the suggestions, they needed to send us an email stating the survey codes which pop up at the end of the questionnaire. We verified their status as a medical student by their email addresses. Moreover, they needed to report their medical knowledge while answering the questions. The study is approved by the IRB.

In the following, we explain how we designed the questions. **Q1**, we asked the participants about their familiarity with the presented symptoms or diseases. They could select their familiarity level from the following options– extremely familiar, very familiar, moderately familiar, slightly familiar, not familiar at all, where

extremely familiar represents highly knowledgeable with the disease. **Q2**, we wanted to investigate whether the given responses matched the request. The participants could choose one of the following responses– extremely well, very well, moderately well, slightly well, not well at all. **Q3**, we wanted to reveal the skill's capability in providing suggestions with the available information. The participants could select one of the following options– yes, maybe, no, and I don't know that. **Q4**, we wanted to investigate whether the provided answer is correct or not. The participants could choose either one of the following options to determine the correctness of the responses– yes, no, I do not know that. Sometimes the participants might not be able to determine the correctness due to not knowing much about the disease or symptoms. We didn't force them to select between yes or no answer. That's why we included the third option here. If the answer for Q4 was incorrect, we navigated to a branch and asked them Q5, Q6 to understand the risk and other information needed. If the answer for Q4 was correct, then we asked Q7 to get more insights and balance the workload of the survey. **Q5**, for incorrect responses, we wanted to measure the riskiness among the following five classes– very risky, risky, neutral, less risky, and not risky. **Q6**, we asked the participants the other required information for the wrong responses to provide a correct response. Again, we only collected this information for the incorrect responses (as selected in Q4). **Q7**, when the participant marked the responses as correct in Q4, we asked for the additional required information to provide better suggestions.

To reduce the fatigue of a long questionnaire, we divided those 90 responses from medical skills into nine sets, where each set contained 10 suggestions. We have uploaded one set of questions under this anonymous link [16].

Analysis Results. Each of the 90 unique medical suggestions was evaluated by three medical students independently. When there were disagreements among participants, we used a majority vote to decide. All participants were in the Medical Doctor program. Their age range was between 22-28. All of them were male. They were familiar with the symptoms listed in 75 out of 90 (83.33%) responses. For the other 15 responses, they marked themselves as not familiar with the symptoms. On average, each set (10 responses) took 12 minutes to complete. In total, each participant needed 108 minutes to complete the nine sets (90 responses) of the survey.

Detailed results are listed in Figure 3. Among 90 responses from medical skills, only 29 suggestions match the asked questions according to the medical experts. Note that after the three computer science students filtered out the obviously non-relevant answers, many remaining responses still didn't answer the questions, usually they provided general medical information instead. For example, when "What to do for skin ulcers?" was asked, the skill responded with the information about 'Skin Ulcers', which was– "skin ulcers are caused by bad blood circulation from the leg." Even though the information was correct, the answer didn't match with the asked question. Considering that we already manually removed 170 responses that were not medical information (with the help of computer science researchers), the percentage of providing actual relevant medical suggestions was pretty low (11.2%, 29 out of 260). In addition, when asked whether they believed the VPA skills could answer the medical questions, the experts thought less than half of them (46.7%, 42 out of 90) had the capability to give useful suggestions about the symptom or disease given the medical question asked. Then we asked whether they think the provided suggestions were correct, 76.7% (69 out of 90) of provided suggestions were considered correct even though only 32.2% (29 out of 90) of them matched the asked questions. Domain experts believed for 53.33% (48 out of 90) of cases, the skills should have asked more questions before providing medical suggestions. Participants also suggested that without enough information, the skill only provided correct but not related responses. For example, when asked about what to do for a sore throat, a response describing sore throat would be considered correct but not relevant. For those incorrect responses, we continued by asking how risky they were. 3/12 (25%) of them are considered risky. However, the overall percentage (3/90) is pretty low. Participants raised a few concerns about the quality of the responses, even though those are correct. Here, we are listing some of their suggestion – *(1) Ask Follow-up Questions:* The experts suggested skills should ask more questions before giving suggestions. For example, when the patient asked about distention, ask the following question: "ask

about distention after eating or with different positions, alcohol use, blood product exposure, hepatitis history". *(2) Suggest to visit Doctor:* In several critical scenarios, the skill should suggest the patient visit a physician or a doctor. As quoted from the study– "stop all activity, alert someone near, call 911 if persistent after an hour.", "The patient should be urgently evaluated by a doctor". *(3) Suggest different treatment for different causes of the same disease:* The experts suggested that some treatment was not appropriate for all the etiologies of a certain disease. For example, causes of paresthesia are – diabetes, meralgia paresthetica, and multiple sclerosis. Treatment also varies for diabetes to meralgia paresthetica.

In conclusion, 88.8% (231(170+61) out of 260) of medical responses are not helpful mostly because they are non-relevant to the questions asked, not because they are providing medically inaccurate information. For those 12 responses that are incorrect, a quarter of them are considered risky. There are three major reasons why the medical suggestions are of low-quality. **First**, skills don't understand the voice command due to NLP challenges. The 170 out of 260 unique responses (260 out of 562,302 total responses) belong to this category. We also found anecdotal points of evidence when we manually analyzed five open-source medical skills [4, 6, 10, 12, 17]. These skills follow a non-flexible template based scheme to handle voice commands. Skill Developers define several intents (*e.g.,* "FindTherapist", "StartReport") and map the voice commands (*e.g.,* "can you find me a therapist", "can you send a report for me") to the corresponding intents according to a dictionary. Sometimes, it becomes very difficult to cover all the possible voice utterances which lead to incorrect behavior. A slight modification in the voice command can cause the skill to misinterpret the user request completely differently and initiate unintended activity. For example, DepressionAI [10] detects the intent as *positive feeling* when a user says their feelings as "not ok", due to the lack of wordings in their intent schema. **Second,** even when skills understand the questions, skills might still lack information to answer the user's medical question. From our domain expert's assessment (Figure 3(b)), we can observe that for 53.33% of the cases (48 out of 90 unique responses, 2,828 out of 31,725 total responses), skills do not have the information to provide useful suggestions about the symptom or disease. **Third,** even worse they might be providing wrong and risky answers. Out of 90 unique responses, we find 12 (13.33%) incorrect responses (51 out of 31,725 total responses). Among them, domain experts identify 3 unique responses (3 out of 31,725 total responses) as risky (Figure 3(d))).

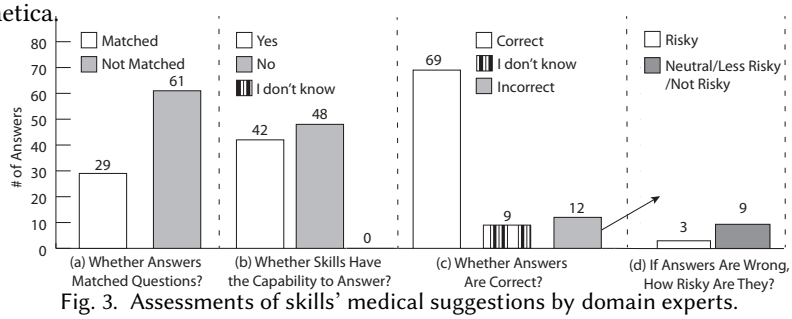

Fig. 3. Assessments of skills' medical suggestions by domain experts.

5.4 P4: Promote Black-Market Sale of Drugs.

We used our dynamic module to identify the violation of (P4) as it was a behavior-based policy. Through our analysis, we did not find any violations of P4 — we include the analysis for completeness.

We collected the list of black market items from a popular dataset [9]. This dataset [9] contains name, category, seller rating, price of 109,692 unique black-market items. Among them, 93,330 (85%) items belong to the drugs category. Out of those 93,330 drugs, we select the top 106 drugs (based on seller rating) to search their information in the skill market. We crafted four voice command templates. They were – 'i want to buy X', 'give me information about X', 'where can I get X', 'what is X', where X represents the name of drugs (*e.g.,* 'give me information about *Spice K2 Synthetic Marijuana*'). By connecting those four voice command templates with 106 different drugs, we created 424 voice commands. If any skills provided buying information about those 106 drugs, then we marked this behavior as a violation. Normally, people buy and sell black market items via crypto-currency [27]. Also, from the dataset [9], we can observe that all the prices are listed in bitcoin currency. We identified buying information by searching for the item's name, crypto-currency related words in the skill's response.

We didn't find any violation of the skills we tested. To ensure the results were reliable, we performed manual analysis on a sample of the results. We selected 15 drugs randomly from [9] and crafted 60 (15 X 4) active probing commands. We selected 20 skills (based on featured) and collected 56 passive voice commands. We tested 63 voice commands (approx.) per skill. In total, we investigated the response of 1,256 voice commands. However, we didn't find any skills providing the information of black-market sale of prescription drugs. The same thing happened when we used `VerHealth` to reveal skills providing such information.

5.5 P5: Absence of Disclaimer

We identified the violation of (P5) using our static module since it was a description-based policy.

CNN Model. For detecting the presence of a disclaimer, a simple approach would be to identify the existence of disclaimer representative keywords (such as,. "educational purposes", "informational purposes") in the description. However, from our initial investigation, we found that this naive approach would not work as different skills use different wordings to describe the disclaimer. Different from P2, to identify the disclaimer, we also need to understand the context information. As a consequence, we decided to train a deep learning model to identify a disclaimer inside a description. To do that, we selected 2,052 sentences from 400 skills to train our machine learning model. Out of those 400 skills, 59 skills included a disclaimer. For evaluation, we selected 1,241 sentences from 250 skills' descriptions. Our test set contained 33 skills that have a disclaimer in their description.

Whenever our model predicted any sentences as positive then we marked that document as positive, meaning it had the disclaimer in its description. Otherwise, we marked that skill as "missing disclaimer" if our model predicted none of the sentences as positive. We designed different network architectures (CNN, RNN-LSTM, Bi-Directional RNN) to compare the performance. Applying CNN to text classification has been proven effective compared to the other DNN (Deep Neural Network) models [49, 56]. We used 31,413 skill's description to build our embedding layer using word2vec [66]. We followed the network architecture from the paper [49]. Our model classifies the outcome of the inputted sentence using a sigmoid dense layer, predicting the presence of a disclaimer. We experimented with different hyperparameters (*e.g.,* epoch, learning rate, word embedding dimension, batch size) and evaluated the performance using the validation dataset. We selected the hyperparameters for which we get the best performance in the validation dataset.

Table 6. Performance comparison of different models in detecting the presence of disclaimer.

Model	Accuracy	Precision	Recall	F1-Score
CNN	97.01%	93.54%	87.87%	90.62%
RNN-LSTM	95.6%	82.35%	84.84%	83.58%
Bi Directional RNN	93.6%	81.48%	66%	73.33%

Detection Result. We found 697 (86.36%) skills missing a disclaimer in their description. From Table 6, we can observe that CNN outperforms the other models in terms of accuracy, precision, recall, and F1-Score. We performed 10-fold cross-validation to check the performance of our model. In the cross-validation, we achieved 99.2% accuracy, 97.7% precision, 89% recall, and 91.8% F1-score. We achieved a good performance with 97.01% accuracy, 93.54% precision, 87.87% recall, and 90.62% F1-score on the test document.

5.6 Summary of Experiments and Results

For each of the experiments, all the passive commands remain the same as they are collected and generated from static information such as descriptions. We crafted active probing commands differently based on the context of each of the experiments (Section 5). From Table 8, we show the total number of active probing commands that we used for each of the experiments. In total, we have 855,276 active probing commands that we tested using `VerHealth`. Also, we used 863,988 responses to detect policy violations.

To have a systematic understanding of policy violations, we combined our static module and dynamic module to run a full analysis of 813 health-related skills. From Table 7, we can observe that 244 skills (30.23%), 697 (86.36%) skills violate (P5). We didn't find any skill violating (P2) & (P4). We ran a manual analysis to verify P2 and P4 violations and get the same results as VerHealth. That means VerHealth has 100% accuracy in terms of detecting violation of (P2) & (P4).

Table 7. Performance computation of VerHealth and detection results on 813 health-related skills.

ID	Policy	#Skill	Accuracy	F1-Score
(P1)	Storing Information	244 (30.23%)	98.8%	97.08%
(P2)	Life-saving Assistance	0	100%	-
(P3)	Misleading Information	48 (5.9%)	-	-
(P4)	Prescription of Black Market	0	100%	-
(P5)	Missing Disclaimer	697 (86.36%)	97.01%	90.62%

For detecting violations of (P3), we got 260 unique responses from testing, and first filtered out 170 apparently irrelevant responses, and then asked three domain experts to evaluate the rest of 90 responses. The domain experts found that 13.33% (12 out of 90) responses were incorrect, and 3 incorrect responses were risky. The 12 incorrect responses were from 5.9% (48 out of 813) skills (note that some skills might have the same responses for the same questions). These skills had an average of 3.21/5.0 ratings. However, although most responses were correct, the domain experts reported that 67.78% (61 out of 90) was irrelevant to the asked question. Then for the total 260 unique responses (562,302 in total), we have 231/260 (88.84%) responses from 783 out of 813 skills (96.3%) are irrelevant. In addition, the domain experts suggested that 76.67% (69 out of 90) cases, the 25 out of 40 (62.5%) skills should have asked more questions before providing medical suggestions.

Table 8. Computation overhead of VerHealth.

ID	#Passive Prob.	#Active Prob.	#Skill	#Total Act. Prob.	#Total Cmd.	Dynamic Overhead (s)	Overall Overhead(s)
(P1)	2,178	24	813	19,512	21,690	39,042	39,042
(P2)	2,178	4	813	3,252	5,430	4,887	4,938
(P3)	2,178	600	813	487,800	489,978	538,975.8	538,975.8
(P4)	2,178	424	813	344,712	346,890	319,138.8	319,138.8
(P5)	-	-	813	-	-	-	2671

We also evaluated the computation overhead of VerHealth. For all the experiments, we kept the setting as described at the beginning of Section 5 (Desktop PC with 16 GB of RAM and 3.1 GHz Intel Core i5 processor). We show the overall time and overhead of dynamic testing of VerHealth in Table 8. Note that the dynamic testing and static testing all happened offline and won't impact user experiences. For dynamic testing of (P1)–(P4), we calculated the computation time and present it in dynamic overhead (in second) in Table 8. It took on an average 1.044 sec to process a command and store its response while doing dynamic testing. For the hybrid policy (P2), we also had static computation time which was 51 sec. In total, we had 4,938 sec computation overhead for detecting P2. For checking (P5) violation, our model took 2,329 sec for training and 342 sec for testing.

6 DISCUSSION & LIMITATION

In this section, we summarize the key results, and discuss their implications and future research directions.

Key Results and Implications. Our results have a number of important implications for improving healthcare applications on VPA platforms. *First*, our results suggest that there is a gap between the existing privacy/safety policies and their enforcement in practice. For example, out of 813 healthcare skills, we detected 244 skills (30.23%) storing user physical and mental health information; we also detected 697 (86.36%) skills missing the required disclaimers. The gap is likely due to the fact that VPA platforms such as Amazon Alexa lack the necessary vetting process on healthcare applications. To this end, VerHealth can be used by VPA platforms to check a

skill's compliance automatically before the skill can be published to the app store. VerHealth can also help skill developers to proactively check their own skills during the development phase to detect and address policy violations.

Second, our analysis shows that the medical suggestions given by Alexa skills are of low quality. More specifically, out of 562,302 total responses collected, the vast majority are template responses from a small set of medical databases. After removing duplicated responses, there are only 260 unique responses. More importantly, most of these responses (65.38%) are rated as irrelevant to the questions asked. The results suggest that most healthcare skills are simply matching the pre-scripted answers based on user questions but they are struggling to understand users' true intentions. To provide better medical services, on one hand, we need a substantially richer medical knowledge base in the backend. On the other hand, we need more advanced natural language processing techniques to infer user intention in order to generate relevant and helpful answers.

Third, working with domain experts (medical school students), we have identified a small number of wrong answers and even risky answers to users' questions. In particular, the domain experts pointed out that the skills are not qualified to provide emergent support for mental distresses. In addition, for more than 50% of the cases, the experts believe the skills should have asked more questions before giving the advice/diagnosis. These results suggest a critical problem for future research, that is, how to improve the *context awareness* of the healthcare skills. For example, the skill needs to learn to recognize problems that they are not qualified to answers (*e.g.*, emergent mental distress). Another open question is how to develop the interaction scheme between users and healthcare applications to collect the needed context information before generating helpful suggestions. Addressing this problem will require joint efforts from medical professionals and domain experts in HCI (human-computer interaction) and NLP (natural language processing).

Adaptability to Other Platforms and New Policies. VerHealth framework can be easily extended to vet other voice assistant platforms. First, policies to ensure safety and privacy are similar across different platforms. For example, Google Home also has its own app store, and has a similar set of policies for healthcare applications [15]. The main policies of Google Home include: 1) Skills should not involve the transmission of information that could be considered "Protected Health Information" under HIPAA (similar to Alexa's P1); 2) skills that provide health information must include a disclaimer at the beginning of the user's first conversation with the skill and in the skill description (similar to Alexa's P5); 3) Skills cannot facilitate the sale of recreational drugs (similar to Alexa's P4). Essentially, Google Home only has a subset of the Alexa policies. Second, considering the highly similar system architecture and interaction interface of the voice assistants, we expect VerHealth to be easily extended Google Home [3]. In particular, Google Home follows very similar patterns for building skills and voice commands (*e.g.,* wake word + action). We only need to change the endpoint of the debugging console to Google Home [3] to run VerHealth.

An interesting future work is to extend VerHealth to Google Homes to perform a cross-platform analysis. From our brief manual analysis on Google Home, we find several skills violating (P1). For example, *Hubble Baby* (5/5 stars) stores pregnancy due date, location, *Healthy Girl, She Bleeds* stores women's physical health information. We also find Google Home skills that might violate P3. For example, *OCD Finder* (2.6/5 stars) identifies whether a user has Obsessive-Compulsive Disorder (OCD) or not. They provide wrong analysis and we also find users are complaining about this skill by posting their reviews. *Fitness Tips* (4.4/5 stars), *Dr. Ayurveda* (4.8/5 stars) provides health-related information without disclaimers. Currently, VerHealth is only designed to check health-related policies for healthcare skills. There are other policies designed for general voice applications. Our future work will focus on extending our tool to check other policies and detect a broader set of violations.

Robustness. To ensure we identify the violations in *new* or *updated* skills, we can run VerHealth periodically to check for the violations. To improve the robustness of new skill behaviors, we can also retrain our models periodically as they are lightweight. There are several ways smart attackers can try to bypass the detection. First,

the skill might try to check if they are interacting with a testing framework or a user and behave differently. Second, the skill might run adaptive attacks to fool VerHealth's machine learning classifiers. For these two factors, we manually interacted with the skills, and had consistent results with VerHealth. In the future, there might be adaptive attacks, and we can improve the robustness by adversarial training [79] and other techniques. Third, for certain policies, skills might hide malicious behaviors (*e.g.,* for P1). Unless having access to skills' own servers, it is not possible to detect these behaviors.

Future Directions for VPA-based Healthcare Applications. In this paper, we advocate more rigorous compliance checking of healthcare skills against privacy and safety policies. Meanwhile, we admit that some of the policies indeed restrict the functionality of the skills, especially the non-HIPAA-compliant skills. For example, without the ability to store a user's physical or mental health information, it would be difficult/impossible for the application to offer any personalized services. To balance the need for privacy protection and rich functionality, a path forward is for VPA applications to facilitate a deep integration with hospital services and healthcare companies. By proactively complying with HIPAA and other privacy policies, the skills would be allowed to access the patient's data to offer richer functionality and personalized services.

Limitations. This work has a few limitations. **First**, a common issue for dynamic testing is the limited coverage, which leads to false negatives. In VerHealth, we introduced schemes (*e.g.*, rephrase generator) to improve the coverage of voice commands. From the end-to-end evaluation, we show VerHealth has a reasonably high recall (91.1%). Even so, we still expect a small number of false negatives (violations that are not detected). **Second**, Alexa Testing Interface (Section 3.2) does not always behave like the real Alexa device [18] due to the limited access to system APIs. For example, we cannot retrieve the time zone information (which needs *deviceID*) or render the video playback. Fortunately, these APIs are not needed for our tests. **Third**, the accuracy of certain components (*e.g.*, recommended command extractor) is not very high. This is indeed a limitation. However, together with other approaches to uncover the function logic of the skill, our end-to-end performance is satisfactory. As shown in Section 5, VerHealth can detect the policy violation accurately (F1-Score=93.85%). **Fourth**, to detect misleading medical suggestions, we consulted a small number of domain experts (people with a medical background). This effort is still very preliminary. One future direction is to recruit a large number of domain experts as well as normal users (people without medical background) to assess the quality of medical suggestions. This could help researchers to understand the experience of regular people, and across-compare their perceptions with domain experts. Another future direction is to develop automated tools to detect inaccurate/misleading medical suggestions and present warnings to users.

At the high-level, this paper focuses on designing probing questions and analysis tools to automatically detect policy violations. There are still parts of the system that require human involvements (*e.g.*, determining misleading information). A future direction is to engineer the automated analysis together with crowdsourcing efforts to jointly analyze policy violation behaviors or allow VPA users to report observed violations when they interact with voice assistant applications.

7 CONCLUSION

In this paper, we propose a system VerHealth, to systematically vet the health-related applications on Amazon Alexa for their compliance with privacy and safety policies. To detect violations, VerHealth combines a static module to analyze skills' descriptions, and a dynamic module to interact with the skills to check their behaviors. We use VerHealth to analyze 813 health-related applications on Amazon Alexa and identify pervasive violations among health-related skills. These violations include storing medical information, providing wrong information, and missing disclaimers. In particular, our study shows there are still many open challenges to build usable, privacy-friendly, and safe medical skills. More research is needed to explore the design space to further improve both the functionality and privacy protection.

ACKNOWLEDGMENTS

We thank anonymous reviewers for their helpful feedback and suggestions. This work was supported in part by National Science Foundation (NSF) grants CNS-1920462, CNS-1943100, and CNS-2030521.

REFERENCES

[1] 1996. Health Insurance Portability and Accountability Act. https://aspe.hhs.gov/report/health-insurance-portability-and-accountability-act-1996.
[2] 2019. NHS Signs Deal With Amazon For Medical Information. https://www.mediapost.com/publications/article/338031/nhs-signs-deal-with-amazon-for-medical-information.html.
[3] 2020. Actions Simulator. https://developers.google.com/assistant/console/simulator.
[4] 2020. Alexa Personal Health Assistant Skill. https://github.com/happyvig/alexa-personal-health-assistant-skill.
[5] 2020. Alexa Skill Website. https://www.amazon.com/alexa-skills/b/?ie=UTF8&node=13727921011.
[6] 2020. Amazon Alexa Nursing Skill Workshop. https://github.com/InternetOfHealthcare/nursing-alexa-skill-workshop.
[7] 2020. Amazon Policy for Skills. https://developer.amazon.com/docs/custom-skills/policy-testing-for-an-alexa-skill.html/#3-health.
[8] 2020. Apple app store. https://www.apple.com/ios/app-store/.
[9] 2020. Dark Net Marketplace Data. https://www.kaggle.com/philipjames11/dark-net-marketplace-drug-data-agora-20142015.
[10] 2020. DepressionAI. https://github.com/Jflick58/DepressionAI.
[11] 2020. Google play store. https://play.google.com/store/apps?hl=en_US.
[12] 2020. MedPal. https://github.com/sayak119/MedPal/.
[13] 2020. Mental Illness. https://www.mayoclinic.org/diseases-conditions/mental-illness/symptoms-causes/syc-20374968.
[14] 2020. New Research Utilizes Voice Assistant Systems for Early Detection of Cognitive Decline. https://www.eurekalert.org/pub_releases/2020-01/dmc-nru012720.php.
[15] 2020. Policies for Actions on Google. https://developers.google.com/assistant/console/policies/general-policies.
[16] 2020. Qualitative Assessment Question. https://drive.google.com/open?id=1oPB4Mat9whr3YNgHIALkFzyaB0tklVLE.
[17] 2020. Sample AWS Lambda function for Alexa. https://github.com/elementarydesigns/pillButler.
[18] 2020. Simulator Limitations. https://developer.amazon.com/en-US/docs/alexa/devconsole/test-your-skill.html#alexa-simulator-limitations.
[19] 2020. Symptom Disease Sorting. https://www.kaggle.com/plarmuseau/sdsort#symptoms2.csv.
[20] 2020. Test and Debug a Custom Skill. https://developer.amazon.com/docs/custom-skills/test-and-debug-a-custom-skill.html.
[21] 2020. The Best Voice Assistants. https://www.reviews.com/home/smart-home/best-voice-assistant/.
[22] Saleh Ahmed, Mahboob Qaosar, Rizka Wakhidatus Sholikah, and Yasuhiko Morimoto. 2018. Early Dementia Detection through Conversations to Virtual Personal Assistant. In *2018 AAAI Spring Symposium Series*.
[23] Steve Alder. 2018. What is Considered Protected Health Information Under HIPAA? https://www.hipaajournal.com/what-is-considered-protected-health-information-under-hipaa/.
[24] Zulfiqar Ali, Ghulam Muhammad, and Mohammed F Alhamid. 2017. An automatic health monitoring system for patients suffering from voice complications in smart cities. *IEEE Access* 5 (2017), 3900–3908.
[25] Kristin Arnold. 2019. Top 5 "In Case of Emergency" Apps. https://www.safety.com/emergency-apps/.
[26] Timothy W Bickmore, Ha Trinh, Stefan Olafsson, Teresa K O'Leary, Reza Asadi, Nathaniel M Rickles, and Ricardo Cruz. 2018. Patient and consumer safety risks when using conversational assistants for medical information: an observational study of Siri, Alexa, and Google Assistant. *Journal of medical Internet research* 20, 9 (2018), e11510.
[27] Adrian Chen. 2011. Underground Website Lets You Buy Any Drug Imaginable. https://www.wired.com/2011/06/silkroad-2/.
[28] Long Cheng, Christin Wilson, Song Liao, Jeffrey Young, Daniel Dong, and Hongxin Hu. [n.d.]. Dangerous Skills Got Certified: Measuring the Trustworthiness of Skill Certification in Voice Personal Assistant Platforms.
[29] Joseph A Diaz, Rebecca A Griffith, James J Ng, Steven E Reinert, Peter D Friedmann, and Anne W Moulton. 2002. Patients' use of the Internet for medical information. *Journal of general internal medicine* 17, 3 (2002), 180–185.
[30] Wen Dong, Tong Guan, Bruno Lepri, and Chunming Qiao. 2019. PocketCare: Tracking the Flu with Mobile Phones Using Partial Observations of Proximity and Symptoms. *Proc. of the UbiComp'19*.
[31] Afsaneh Doryab, Anind K Dey, Grace Kao, and Carissa Low. 2019. Modeling Biobehavioral Rhythms with Passive Sensing in the Wild: A Case Study to Predict Readmission Risk after Pancreatic Surgery. *Proc. of the UbiComp'19*.
[32] Nathan Eddy. 2019. Voice Assistants for Health Could Use Improvement, Study Finds. https://www.healthcareitnews.com/news/voice-assistants-health-could-use-improvement-study-finds.
[33] Justin Edwards and Elaheh Sanoubari. 2019. A need for trust in conversational interface research. In *Proceedings of the 1st International Conference on Conversational User Interfaces*. 1–3.

[34] Josef Essberger. 2020. Different Types Questions. https://www.englishclub.com/grammar/sentence/type-interrogative.htm.
[35] Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin* 76, 5 (1971), 378.
[36] Amy Fontinelle. 2019. How Black Markets Work. https://www.investopedia.com/articles/economics/12/mechanics-black-market.asp.
[37] Nicole Gallucci. 2017. Need Some Medical Advice? Try Asking Alexa. https://mashable.com/2017/03/07/amazon-echo-alexa-health-related-questions/.
[38] Jacqueline Garcia and Deepa Bharath. 2019. Black Market Drugs Offer Risky Relief for Uninsured, Low-Income Californians. https://www.centerforhealthjournalism.org/black-market-drugs-offer-risky-relief-uninsured-low-income-californians.
[39] Jefferson Graham. 2017. "Alexa, Call 911" Won't Work. Here's What Will. https://www.usatoday.com/story/tech/talkingtech/2017/07/19/alexa-cant-dial-911-but-google-alexa-and-siri-can-get-you-help/486075001/.
[40] Paul Grant. 2018. Tens of Millions of Prescription Drugs on the Black Market. https://www.bbc.com/news/health-42810148.
[41] Zhixiu Guo, Zijin Lin, Pan Li, and Kai Chen. 2020. SkillExplorer: Understanding the Behavior of Skills in Large Scale. In *Proc. of USENIX Security'20*.
[42] Hamza Harkous, Kassem Fawaz, Rémi Lebret, Florian Schaub, Kang G Shin, and Karl Aberer. 2018. Polisis: Automated analysis and presentation of privacy policies using deep learning. In *Proc. of USENIX Security'18*.
[43] Matt Hasten. 2020. Apply to the HIPAA-Eligible Skill Program. https://developer.amazon.com/en-US/blogs/alexa/alexa-skills-kit/2020/08/hipaa-eligible-skills.
[44] Healthline. 2019. Anonymous Nurse: Please Stop Using 'Dr. Google' to Diagnose Your Symptoms. https://www.healthline.com/health/please-stop-using-doctor-google-dangerous#1.
[45] M Shamim Hossain, Ghulam Muhammad, and Atif Alamri. 2019. Smart healthcare monitoring: A voice pathology detection paradigm for smart cities. *Multimedia Systems* 25, 5 (2019), 565–575.
[46] Sozo Inoue, Paula Lago, Tahera Hossain, Tittaya Mairittha, and Nattaya Mairittha. 2019. Integrating Activity Recognition and Nursing Care Records: The System, Deployment, and a Verification Study. *Proc. of the UbiComp'19*.
[47] Rachel Jiang. 2019. Introducing New Alexa Healthcare Skills. https://developer.amazon.com/blogs/alexa/post/ff33dbc7-6cf5-4db8-b203-99144a251a21/introducing-new-alexa-healthcare-skills.
[48] Mi Ok Kim, Enrico Coiera, and Farah Magrabi. 2017. Problems with health information technology and their effects on care delivery and patient outcomes: a systematic review. *Journal of the American Medical Informatics Association* 24, 2 (2017), 246–250.
[49] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *Proc. of EMNLP'14*.
[50] Hannah Kuchler. 2019. Amazon Expands Healthcare Services With Alexa Deals. https://www.ft.com/content/9c3fb428-56e7-11e9-a3db-1fe89bedc16e.
[51] Raina Langevin, Mohammad Rafayet Ali, Taylan Sen, Christopher Snyder, Taylor Myers, E Ray Dorsey, and Mohammed Ehsan Hoque. 2019. The PARK Framework for Automated Analysis of Parkinson's Disease Characteristics. *Proc. of the UbiComp'19*.
[52] Josephine Lau, Benjamin Zimmerman, and Florian Schaub. 2018. Alexa, are you listening? privacy perceptions, concerns and privacy-seeking behaviors with smart speakers. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–31.
[53] David Lazarus. 2019. Alexa May Be Key to Amazon's Looming Domination of the Healthcare Market. https://www.latimes.com/business/lazarus/la-fi-lazarus-amazon-healthcare-privacy-20190716-story.html.
[54] Esther Levin and Alex Levin. 2006. Evaluation of spoken dialogue technology for real-time health data collection. *Journal of Medical Internet Research* 8, 4 (2006), e30.
[55] Li Li, Tegawendé F Bissyandé, Mike Papadakis, Siegfried Rasthofer, Alexandre Bartel, Damien Octeau, Jacques Klein, and Le Traon. 2017. Static analysis of android apps: A systematic literature review. *Information and Software Technology* 88 (2017), 67–95.
[56] Shen Li, Zhe Zhao, Tao Liu, Renfen Hu, and Xiaoyong Du. 2017. Initializing convolutional filters with semantic features for text classification. In *Proc. of EMNLP'17*.
[57] Yuting Liao, Jessica Vitak, Priya Kumar, Michael Zimmer, and Katherine Kritikos. 2019. Understanding the role of privacy and trust in intelligent personal assistant adoption. In *International Conference on Information*. Springer, 102–113.
[58] Fei Liu, Nicole Lee Fella, and Kexin Liao. 2016. Modeling language vagueness in privacy policies using deep neural networks. In *Proc. of AAAI'16*.
[59] Sarah M. 2020. What Is Physical Health? - Definition, Components & Examples. https://study.com/academy/lesson/what-is-physical-health-definition-components-examples.html.
[60] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*. 55–60.
[61] Christopher D. Manning, Hinrich Schütze, and Prabhakar Raghavan. 2009. Dropping Common Terms: Stop Words. https://nlp.stanford.edu/IR-book/html/htmledition/dropping-common-terms-stop-words-1.html.
[62] Elad Maor, Daniella Perry, Dana Mevorach, Nimrod Taiblum, Yotam Luz, Israel Mazin, Amir Lerman, Gideon Koren, and Varda Shalev. 2020. Vocal Biomarker Is Associated With Hospitalization and Mortality Among Heart Failure Patients. *Journal of the American Heart Association* 9, 7 (2020), e013359.

[63] Lucas Matney. 2019. More than 100 million Alexa devices have been sold. https://techcrunch.com/2019/01/04/more-than-100-million-alexa-devices-have-been-sold. *TechCrunch* (2019).
[64] Aarthi Easwara Moorthy. 2013. *Voice activated personal assistant: Privacy concerns in the public space.* California State University, Long Beach.
[65] Tim Newman. 2009. What is Mental Health? https://www.medicalnewstoday.com/articles/154543.php.
[66] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.
[67] Sarah Masud Preum, Abu Sayeed Mondol, Meiyi Ma, Hongning Wang, and John A Stankovic. 2017. Preclude2: Personalized conflict detection in heterogeneous health applications. *Pervasive and Mobile Computing* 42 (2017), 226–247.
[68] Bhagyashree R. 2019. Amazon's Partnership With NHS to Make Alexa Offer Medical Advice Raises Privacy Concerns and Public Backlash. https://hub.packtpub.com/amazons-partnership-with-nhs-to-make-alexa-offer-medical-advice-raises-privacy-concerns-and-public-backlash/.
[69] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250* (2016).
[70] Elisabeth Rosenthal. 2019. Analysis: Why Alexa's Bedside Manner Is Bad for Health Care. https://khn.org/news/analysis-why-alexas-bedside-manner-is-bad-for-health-care/.
[71] Lizawati Salahuddin and Zuraini Ismail. 2015. Classification of antecedents towards safety use of health information technology: A systematic review. *International journal of medical informatics* 84, 11 (2015), 877–891.
[72] Asif Salekin, Jeremy W Eberle, Jeffrey J Glenn, Bethany A Teachman, and John A Stankovic. 2019. A weakly supervised learning framework for detecting social anxiety and depression. *Proc. of the UbiComp'19*.
[73] Faysal Hossain Shezan, Hang Hu, Jiamin Wang, Gang Wang, and Yuan Tian. 2020. Read Between the Lines: An Empirical Measurement of Sensitive Applications of Voice Personal Assistant Systems. In *Proceedings of The Web Conference 2020*. 1006–1017.
[74] Bill Siwicki. 2018. Special Report: AI Voice Assistants Making an Impact in Healthcare. https://www.healthcareitnews.com/news/special-report-ai-voice-assistants-making-impact-healthcare.
[75] Julie Spitzer. 2018. Many Amazon Alexa Health Skills Violate Company Policies. https://qz.com/1323940/alexa-is-a-terrible-doctor/.
[76] Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics* 26, 3 (2000), 339–373.
[77] Eylon Stroh and Priyank Mathur. 2016. Question answering using deep learning.
[78] Madiha Tabassum, Tomasz Kosiński, Alisa Frik, Nathan Malkin, Primal Wijesekera, Serge Egelman, and Heather Richter Lipford. 2019. Investigating Users' Preferences and Expectations for Always-Listening Voice Assistants. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 4 (2019), 1–23.
[79] Florian Tramèr and Dan Boneh. 2019. Adversarial training and robustness for multiple perturbations. In *Advances in Neural Information Processing Systems*. 5858–5868.
[80] Rui Wang, Weichen Wang, Alex daSilva, Jeremy F Huckins, William M Kelley, Todd F Heatherton, and Andrew T Campbell. 2019. Tracking depression dynamics in college students using mobile phone and wearable sensing. *Proc. of the UbiComp'19*.
[81] John Wieting, Jonathan Mallinson, and Kevin Gimpel. 2017. Learning paraphrastic sentence embeddings from back-translated bitext. *arXiv preprint arXiv:1706.01847* (2017).
[82] Michelle Y Wong and David Lie. 2016. IntelliDroid: A Targeted Input Generator for the Dynamic Analysis of Android Malware.. In *NDSS*, Vol. 16. 21–24.
[83] Zack Zarrilli. 2016. Need to Call 911? There's an App For That! https://www.techsafety.org/blog/2016/8/25/need-to-call-911-theres-an-app-for-that.
[84] Zack Zarrilli. 2016. The Future of Life-Saving: 5 Life-Saving Technologies You Need to Know About. https://www.surefirecpr.com/future-life-saving-5-life-saving-technologies-need-know/.
[85] Xiao Zhang, Wenzhong Li, Xu Chen, and Sanglu Lu. 2019. Moodexplorer: Towards compound emotion detection via smartphone sensing. *Proc. of the UbiComp'19*.
[86] Xiao Zhang, Yongqiang Lyu, Xiaomin Luo, Jingyu Zhang, Chun Yu, Hao Yin, and Yuanchun Shi. 2019. Touch Sense: Touch Screen Based Mental Stress Sense. *Proc. of the UbiComp'19*.
[87] Sebastian Zimmeck, Ziqi Wang, Lieyong Zou, Roger Iyengar, Bin Liu, Florian Schaub, Shomir Wilson, Norman Sadeh, Steven Bellovin, and Joel Reidenberg. 2016. Automated analysis of privacy requirements for mobile apps. In *Proc. of AAAI'16*.