Privacy Preserving Chatbot Conversations

Debmalya Biswas
AI Center of Excellence
Philip Morris International
Lausanne, Switzerland
Email: debmalya.biswas@pmi.com

Abstract—With chatbots gaining traction and their adoption growing in different verticals, e.g. Health, Banking, Dating; and users sharing more and more private information with chatbots - studies have started to highlight the privacy risks of chatbots. In this paper, we propose two privacy-preserving approaches for chatbot conversations. The first approach applies 'entity' based privacy filtering and transformation, and can be applied directly on the app (client) side. It however requires knowledge of the chatbot design to be enabled. We present a second scheme based on Searchable Encryption that is able to preserve user chat privacy, without requiring any knowledge of the chatbot design. Finally, we present some experimental results based on a real-life employee Help Desk chatbot that validates both the need and feasibility of the proposed approaches.

I. Introduction

Chatbots have been touted as the 'Next Interaction Layer', which implies that the way we currently consume information by interacting with websites/apps will in many cases be replaced by chatbots (conversations). A recent Forrester report [1] summarizes the state-of-the-art of chatbots as follows:

> Despite consumers' mixed feelings about and backlash against chatbots, businesses understand chatbots' value and continue to adopt them as a primary engagement channel for customer support. The impact of the COVID-19 pandemic has strengthened organizational resolve to use chatbots to improve servicing and engagement and to mitigate crisis situations. The past two years have also seen chatbot technology vendors rapidly apply conversational computing to customer service across domains and verticals.

Chatbot research has mostly focused on improving the underlying Natural Language Processing (NLP) precision [2], [3], such that chatbots are more proficient in understanding and responding to user queries. With chatbots gaining traction and their adoption growing in different verticals, e.g. Health, Banking, Dating, etc., and users sharing more and more private information with chatbots; studies have started to highlight the privacy risks of chatbots [4]–[6].

However, the proposed approaches are restricted to explicitly shared Personally Identifiable Information (PII), e.g. credit card numbers, bank account details, health conditions, dating preferences; and the solutions restricted to traditional software security techniques, e.g. storage encryption and multi-factor authentication. While security basics are definitely needed, the more advanced and implicit privacy risks of open-ended queries posed by users have not been addressed in literature.

For example, let us consider the two use-cases below to understand the significance of such privacy risks:

Use-case 1. Privacy risks of sentiment analysis: Sentiment analysis is basically an NLP Classification task that allows the chatbot to determine the user's (current) sentiment based on his chat responses (e.g., used in Help Desk bots), such that the bot is able to adapt it's response based on the user's sentiment.

While this is done for a "good" cause, let us now consider their usage in an e-commerce scenario. With dynamic pricing, the bot can quote a higher price based on a very "enthusiastic" query by a user. For example, "Wow, this dress looks amazing! What is its price?" might lead to a higher price quote than a more neutral query: "This dress fits my requirements. What is its price?"

Use-case 2. Open ended queries (location based): Let us consider the privacy risks posed by open ended queries with respect to the user's location. Most chatbots are usually designed/deployed for specific regions. For example, an HR Info bot might be designed for the locations where the company has offices, similarly an e-commerce bot would also be deployed in only those countries where the vendor currently ships their products. Given this, a query such as "Hi, I am currently in Geneva. What are the shipping charges for Geneva?" unnecessarily reveals the user's location given that the vendor does not deliver in Switzerland.

In an organizational context, with an HR chatbot maintained by an outsourced vendor (on a Cloud platform), deployed in Geneva and Krakow; frequent employee queries of the form: "Where is the restaurant in our Milan office?" might reveal to the vendor that company employees have recently been traveling a lot to the Milan office. Traditional security mechanisms, e.g. restricting access to the chatbot logs (via encryption, access control policies, etc.) are not sufficient; as the logs need to be analyzed for continuous improvement of the bots [7].

To address the the above privacy risks posed by chatbot conversations, we propose two privacy preserving approaches in this paper: 'Entity' based (i) privacy filtering and transformation that can be applied on the client/app side (Section II-B), and (ii) Searchable Encryption that can be applied independent of the chatbot design (Section II-D). Implementation/validation results are presented in Section II-C, with Section III concluding the paper and providing some directions for future work.

II. Privacy Preserving Chats

A. Chatbot Basics

We first provide some background in terms of how current chatbots work. In an ideal world, given a user query in natural language, a bot would respond as follows:

1) Understand the user's intent;
2) Retrieve the relevant content from its Knowledge base (KB);
3) Synthesize the answer and respond to the user (again, in natural language);
4) Retain the conversation context to answer any follow-up questions by the user.

Unfortunately, numerous technical limitations prevent us from enabling the above workflow. Enterprise chatbots today (e.g. the ones based on IBM Watson Assistant, AWS Lex, Microsoft LUIS, Google Dialogflow) first need to be trained by providing a set of questions, question variations, and their corresponding answers. The questions can be grouped into 'intents'. Question variations, referred to as 'utterances' in bot terminology, refer to sample variations in which the same question can be posed by end-users. The idea is to provide 5 to 10 such utterances (for each question) as input, based on which the bot will hopefully be able understand 50 different variations of the question. Most bot engines perform intent matching and sentiment analysis using a mix of statistical (e.g. tf-idf, Bag-of-Words) and Deep Learning (e.g. BERT) techniques. When no intent is matched with a confidence level above 30% (configurable), the chatbot returns a fallback answer. For all others, the engine returns the corresponding confidence level along with the response.

B. Entity based Privacy Preservation

In this section, we outline the 'entity' based privacy preservation approach. Together with 'intents' and 'utterances', an essential part in customizing chatbots is to provide 'entities' [8]. The entities refer to the domain specific vocabulary, e.g. they can refer to the office locations, in the context of the HR Bot outlined in Use-case 2; and can be used to customize the chatbot responses according to user (query) location. The entities based approach is applied on the client/app side by a module referred to as the Privacy Preserving Chat Module (PPCM). The PPCM design is dependent on knowledge of the original chatbot content and the underlying NLP techniques used by the chatbot platform. For instance, with reference to Use-case 2, the PPCM needs to be aware of the entities list used in the original chatbot design (allowed office locations), such that it can apply the necessary privacy preserving measure(s) accordingly. The PPCM solution architecture is illustrated in Fig. 1. It applies a mix of privacy preserving techniques based on Filtering and Transformation to address the user chat privacy concerns.

- Filtering: For Use-case 2, with reference to the user query: "Where is the restaurant in our Milan office?", PPCM uses the same text extraction technique as employed by the chatbot NLP engine to infer that 'Milan' is a value of entity type 'Location'. This is followed by filtering/deleting the query such that it does not get sent to the backend NLP engine, with an appropriate message relayed to the user.
- Transformation: To counteract the pricing disadvantage highlighted in Use-case 1, the PPCM needs to adapt the original user query to a more "neutral" response with the same semantics; such that the user sentiment of the transformed query also gets classified as 'neutral' by the chatbot NLP engine - illustrated in Fig. 1. Transformation in the form of abstraction can also be applied for the 'Location' entity type in Use-case 2, where 'Milan' is abstracted to 'somewhere in Europe - to address the user chat privacy concerns.

C. Validation

We validated the proposed entity based privacy preservation approaches on a Help Desk chatbot available for employees in our Geneva and Krakow offices. The chatbot was developed on IBM Watson Assistant and has around 400 intents covering a range of topics related to office equipment, transportation, restaurants, leisure, etc. facilities. The intents configuration file is available at (link). The chatbot has been live for more than 6 months now and we noticed that the chatbot is equally popular among new employees, employees on short term assignment, and regular employees based in Geneva and Krakow - giving us a test audience of around 5000 unique users. We report some observations based on analyzing the first 10000 posed queries. The results validated our hypothesis that many employees still talk to a chatbot as they would to a human being. Rather than asking direct questions, they start their queries by providing some context first. Below are a few sample queries (edited to remove company specific info):

> "I am so stressed after my X^{th} meeting. What is the menu in restaurant Y?"
> "Help, my mobile outlook is stuck again. Where the **** is the Help Desk here?"
> "Hi, I am from Jakarta office. Where is the post office located in this building?"
> "Are the fitness courses here the same as I take in London?"

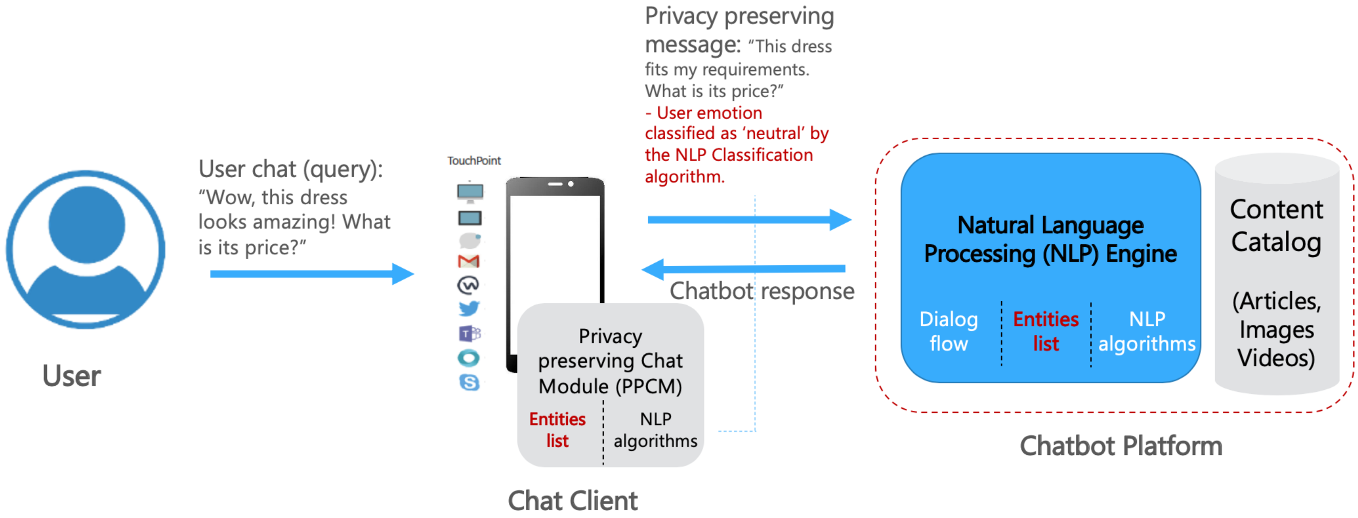
Privacy preserving message: "This dress fits my requirements. What is its price?"
- User emotion classified as 'neutral' by the NLP Classification algorithm.
User chat (query): "Wow, this dress looks amazing! What is its price?"
TouchPoint
User
Privacy preserving Chat Module (PPCM)
Entities list
NLP algorithms
Chat Client
Chatbot response
Natural Language Processing (NLP) Engine
Dialog flow
Entities list
NLP algorithms
Content Catalog
(Articles, Images Videos)
Chatbot Platform

Fig. 1. PPCM architecture highlighting Entity based Privacy Preservation (Use-case 1)

Needless to say, the first two queries are sensitive from an employee mental state (HR) perspective - revealing employee sentiment in terms of stress and distress. The last two queries unnecessarily reveal the employee's base location. We noticed such privacy sensitive information embedded in almost 20% of the queries.

To address the above privacy concerns, we implemented a PPCM client in Python using AWS Lex for sentiment analysis. While Lex is different from IBM Tone Analyzer, the API used by the chatbot for sentiment analysis; they both return the sentiment as a value between 0 to 1. So we were able to cross validate the sentiment values, and also show that different NLP engines can be used for the chatbot and PPCM as long as they have "similar" functionality. As location was the only privacy sensitive entity here, we could implement it as a parser based on a location values dictionary.

In both cases, the privacy sensitive queries having sentiment values greater than a certain threshold and unsupported entities (locations) were satisfactorily handled. In terms of performance, we were able to process both API calls within the allowed 2 second response time. The only disadvantage is the doubling of cost, as a result of the additional PPCM API call. However, chatbot API calls are getting cheaper, and Open Source NLP/Chatbot engines, such as RASA can be leveraged if cost becomes a bottleneck.

D. Distributed Privacy Preservation based on Searchable Encryption

We now address the scenario where the chatbot implementation is closed (black-box) - applies mostly to external chatbots. We propose a Searchable Encryption based scheme that is able to preserve user chat privacy, without requiring any knowledge of the chatbot design and NLP engine algorithms.

Searchable Encryption (SE) [9] is a technique to protect sensitive data, while preserving the ability to search on the server side (cloud [10]). SE allows the server to search encrypted data without leaking information in plaintext data. The two main branches of SE are Searchable Symmetric Encryption (SSE) and Public key Encryption with Keyword Search (PEKS). In this work, we focus on PEKS which enables a number of users who know the public key to produce ciphertexts, but allows only the private key holder to create trapdoors.

We present a Searchable Encryption based scheme for (Chat) Entities (SEE), based on the PERK scheme in [11]. It consists of the following polynomial time randomized algorithms:

- $KGEN(1^k)$ outputs a public-private key pair: (A_{pub}, A_{priv}).
- $SENC(A_{pub}, w, m)$ outputs a searchable encryption s_w of chat message m under entity w and public key A_{pub}.
- $DOOR(A_{priv}, w)$ outputs a trapdoor t_w that allows to search by entity w.
- $TEST(A_{pub}, s_w, t_{w'})$ outputs the message m if $w = w'$.

The encryption scheme $SEE = (KGEN, SENC, DOOR, TEST)$ is semantically secure against a chosen plaintext attack in the random oracle model assuming the Decisional Bilinear Diffie-Hellman (DBDH) problem is intractable [12].

Note that the PPCM is distributed in this case, as such we refer to the Client and Server (Cloud) side components of the PPCM as PPCM$_C$ and PPCM$_S$, respectively. The solution architecture is illustrated in Fig. 2.

Given the SEE scheme, the PPCM steps to enable privacy preserving chats are given below:

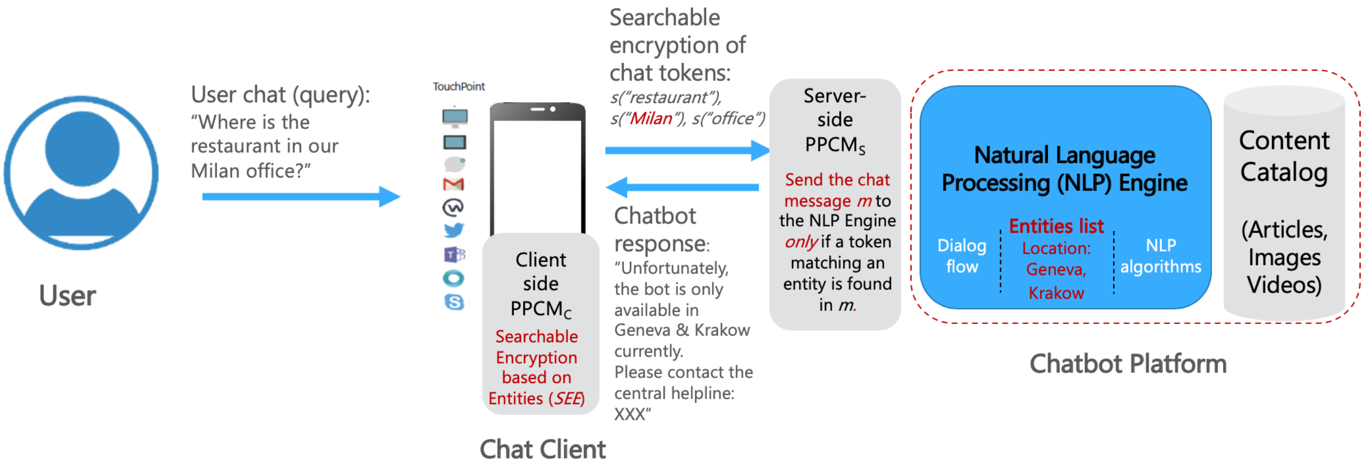
User chat (query):
“Where is the restaurant in our Milan office?”
User
TouchPoint
Client side PPCM$_C$
Searchable Encryption based on Entities (SEE)
Chat Client
Searchable encryption of chat tokens:
s(“restaurant”), s(“Milan”), s(“office”)
Chatbot response:
“Unfortunately, the bot is only available in Geneva & Krakow currently. Please contact the central helpline: XXX”
Server-side PPCM$_S$
Send the chat message m to the NLP Engine only if a token matching an entity is found in m.
Natural Language Processing (NLP) Engine
Dialog flow
Entities list
Location: Geneva, Krakow
NLP algorithms
Content Catalog
(Articles, Images Videos)
Chatbot Platform

Fig. 2. Distributed PPCM highlighting Entity based Search Encryption (Use-case 2)

Initialization

1) The Mobile Chat App ($PPCM_C$) and chatbot Platform ($PPCM_S$), referred to as C and P respectively, run the algorithm $KGEN(1^k)$ to generate their public-private key pairs: (C_{pub}, C_{priv}) and (P_{pub}, P_{priv}).
2) C generates the trapdoor $t_w = DOOR(C_{priv}, w)$, for all entities w in the Entities List L and sends to P.

For every user chat message m:

1) C parses m to extract all tokens (words) W_m in m. It then generates a searchable encryption $s_w = SENC(C_{pub}, w, m)$ for each token w in W_m.
2) C sends to S, the list S_w corresponding to searchable encryption s_w of all tokens w in W_m.
3) For each entity w in L, P compares its trapdoor values t_w with the searchable encryption values s_w shared by C. (Only) On successful match, P obtains the chat message
$m = TEST(C_{pub}, s_f, t_f)$
and processes it with a chatbot response based on its designed dialog flow (as it would do in general without privacy constraints).

Without a successful match, P responds with an “error handling” message and it never gets to see the user's original chat message - preserving user privacy.

III. Conclusion

We outlined two approaches in this paper to perform privacy preserving conversations based on (chat) entities - which approach to apply depends on the transparency of the chatbot design and implementation architecture (client/app side only vs. distributed deployment). We hope that the proposed approaches will lead to increased enterprise adoption of chatbots, by addressing the growing issue of privacy risks in chatbots.

As future work, we are working towards an implementation of the SEE scheme integrated with RASA that would allow us to validate and benchmark both proposed solutions.

References

[1] V. Srinivasan, A. Sharma, D. Hong, S. Dangi, and R. Birrell, “The Conversational Chatbot Buyer's Guide,” Forrester, 2020. [Online]. Available: https://www.forrester.com/report/The+Conversational+Chatbot+Buyers+Guide/-/E-RES158704

[2] I. V. Serban, C. Sankar, M. Germain, S. Zhang, Z. Lin, S. Subramanian, T. Kim, M. Pieper, S. Chandar, N. R. Ke, S. Rajeshwar, A. de Brebisson, J. M. R. Sotelo, D. Suhubdy, V. Michalski, A. Nguyen, J. Pineau, and Y. Bengio, “A Deep Reinforcement Learning Chatbot,” ArXiv, vol. abs/1709.02349, 2017.

[3] B. Hancock, A. Bordes, P.-E. Mazaré, and J. Weston, “Learning from Dialogue after Deployment: Feed Yourself, Chatbot!” 2019.

[4] discover.bot, “Chatbot Security: Putting Customer Privacy First,” 2019. [Online]. Available: https://discover.bot/bot-talk/chatbot-security-putting-customer-privacy-first/

[5] BotsCrew, “How To Make Your Chatbot GDPR Compliant,” 2020. [Online]. Available: https://botscrew.com/blog/how-to-make-your-chatbot-gdpr-compliant/

[6] B. Ondrisek, “Privacy and Data Security of Chatbots,” Medium, 2020. [Online]. Available: https://medium.com/@electrobabe/privacy-and-data-security-of-chatbots-6ab87773aadc

[7] E. Ricciardelli and D. Biswas, “Self-improving Chatbots based on Reinforcement Learning,” in 4th Multidisciplinary Conference on Reinforcement Learning and Decision Making (RLDM), 2019.

[8] W. Shalaby, A. Arantes, T. G. Diaz, and C. Gupta, “Building chatbots from large scale domain-specific knowledge bases: Challenges and opportunities,” ArXiv, vol. abs/2001.00100, 2020.

[9] Y. Wang, J. Wang, and X. Chen, “Secure Searchable Encryption: A Survey,” J. Commun. Inf. Netw., vol. 1, pp. 52–65, 2016.

[10] D. Biswas and K. Vidyasankar, “Privacy Preserving and Transactional Advertising for Mobile Services,” Computing, vol. 96, pp. 613–630, 2014.

[11] D. Biswas, S. Haller, and F. Kerschbaum, “Privacy-preserving Outsourced Profiling,” in 12th IEEE Conference on Commerce and Enterprise Computing, 2010, pp. 136–143.

[12] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, “Public Key Encryption with Keyword Search,” in International Conference on the Theory and Applications of Cryptographic Techniques, 2004, pp. 506–522.