Federated Learning for Preventing Eavesdropping

1st Abdullah Al Masud
dept of CSE
United International University
Dhaka, Bangladesh
amasud183040@bscse.uiu.ac.bd

2nd Safinaz Khan
dept. of CSE
United International University
Dhaka, Bangladesh
skhan183006@bscse.uiu.ac.bd

3rd Mejbaul Alam
dept. of CSE
United International University
Dhaka, Bangladesh
malam183026@bscse.uiu.ac.bd

***Abstract*—Everything we do is online today. From talking, texting, sharing thoughts, shopping, learning and whatnot. The world is at our fingertips. While having thousands of benefits, this also has some grave consequences. We are never alone online. The apps, websites, and smart devices we use are constantly eavesdropping on us. Whatever we say, do is always being heard, recorded and most of the time sent to third parties without our approval or even knowledge. We can't even completely terminate this even if we want to because terminating this can also affect our user experience as our data allows our machines to learn our preferences and work better for us. To prevent these eavesdroppers from always tracking and recording us, we propose a distributed system using Federated Learning where our speech will be detected by our system before going to the main device. Our system will decide if the speech is relevant to the main command and pass it to the main device only if it is and filters it out if it is not, ensuring that our private information or speech irrelevant to the command does not reach the eavesdropping apps, websites or devices. Our system works 91% accurately and successfully meets the goal of this paper.**

***Index Terms*—Federated Learning, Voice Privacy, Network Security, Eavesdropping, Privacy Threats, Smart Devices, Voice Assistants**

I. Introduction

In this digital age, almost anything and everything is at the tip of our fingers. Anything we need is just a few touches away. It surely is a boon to be alive in today's world where everything is so easy and accessible. But just like every coin, this also has two sides. At some point, we have all experienced something similar to incidents where we come across something we have been talking about or messaging about to someone although we have never looked up for it online on our own. Or something we have been discussing with our friends and suddenly everything on our feed is filled with similar or relevant posts, articles, products, etc. Although it might be helpful and fun in some cases, but if we think deeper, this is a breach of our privacy and that too without our consent.

Browsers, various websites, voice assistants, e.g. Alexa, Siri, Google Assistant, etc., and social media apps have become extremely popular in recent years. They have become a part of our life and sometimes, the most important one. However, our involvement with these also raises concerns about our privacy, as they constantly listen to and record users' conversations. They collect a large amount of information about users that includes their locations, voice recordings, search history and other sensitive information. This puts the users at risk of their data being misused due to being exposed to unwanted third parties. It is also rare for these websites, virtual assistants, apps, etc. to let us know exactly what data they collect and who they share with. This lack of transparency makes it difficult for users to make informed decisions about their privacy. Although the companies claim to collect our data for providing us with better experiences and training our devices according to our needs, even without all the issues, it is still naturally very uncomfortable for users to be heard, recorded or tracked all the time. Despite the fact that these companies do take measures to ensure our privacy, the risk still remains.

Apart from privacy issues, there also lies unwanted network traffic issues. Since our devices are always listening, recording and tracking us, they are not keeping these data to themselves. They are constantly sending these data to the eavesdropping apps' or websites' respective servers resulting in unwanted network traffic and causing our internet to slow down or use up the data plan quicker.

While the devices collecting our data put our privacy at risk, terminating this can really downgrade our user experience. Machine learning is a subfield of artificial intelligence that allows computer systems to learn and improve from experience without being explicitly programmed. Training smart devices with machine learning involves using algorithms and statistical models to enable the device to learn from data and improve its performance over time. It involves collecting and processing data from us. Smart devices that use machine learning can improve their performance over time by continuously learning from new data. For example, a smart speaker recommending songs on its own from learning our preferences. While this makes the user experience a lot better, this again raises privacy, security and data ownership concerns. This is where federated learning comes in. Federated learning is a machine learning approach that allows multiple parties to collaboratively train a model without sharing their data with each other. Instead, each party trains a local model on its own data, and then the models' updates are aggregated into a global model. This way, data privacy is preserved, and the parties can benefit from the knowledge learned from each other's data without compromising confidentiality. Federated learning is particularly useful when the data is sensitive, distributed, or too large to be centralized. It has applications in various fields, such as

Identify applicable funding agency here. If none, delete this.

healthcare, finance, and the Internet of Things. Researchers are also actively investigating further methods to improve the efficiency, scalability, and robustness of federated learning.

Hence, keeping everything in mind, this paper proposes a distributed system that listens to our speech and forwards it to the mother device in an encrypted format only if it's relevant, preventing eavesdropping and preserving client privacy. Since our lives are intertwined with our devices and the internet today, is it important to maintain a balance and protect our data while also ensuring the best experience to make our tasks and lives easier. Our system listens, decides if the speech is something helpful or needed for the actual device and then passes or filters out the speech protecting our sensitive information and ensuring our privacy while also helping our devices to learn our preferences and provide us a better experience.

This paper contributes to:

- Unwanted eavesdropping protection
- Data security
- Reducing network traffic
- Better control over the data and devices
- Better user experience

The whole paper is organized into 7 different sections. Section 1 is the introduction where we get brief ideas about why and how our sensitive information is at risk, machine learning, federated learning, the issues that arise if we try to compromise privacy and user experience and a short description of our proposed work. Next in Section 2, we discuss some of the related works on this topic and how they are different from or similar to ours briefly. Section 3 has our proposed work's methodology where we explain and describe how we achieve our desired outcome. In Section 4 we present our experimental results and discuss them further in detail. Finally, the last section, Section 5, is our conclusion where we summarize the paper and discuss future works. Our codes are publicly available at

https://github.com/AAMasud040/Voice-Privacy-using-Federated-Learning.git .

II. Related Works

The paper "Federated Learning-based Anomaly Detection for IoT Security Attacks" (Viraaji Mothukuri, et. al 2021) presents a new approach to detecting anomalies in IoT networks using a decentralized federated learning method with an ensemble [3]. This approach allows for training the anomaly detection machine learning model on the IoT networks without transferring network data to a centralized server, and it enables on-device training. The use of federated learning benefits user data privacy and adds an extra layer of security to IoT networks, making IoT devices more trustworthy.

Sharing speech data can potentially have negative consequences on people's lives. Speech emotion recognition (SER) poses a challenge due to privacy concerns. The paper "Federated Learning for Speech Emotion Recognition Applications" (Siddique Latif, et al. 2020) proposes using federated learning for speech emotion recognition (SER) to preserve user privacy [4]. The authors claim devices trained in a federated environment obtained promising results compared to the state-of-the-art approaches that rely on server-based systems.

Speaker characteristics can be helpful in improving speaker recognition accuracy. However, this type of information is often private. The paper "Improving on-device speaker verification using federated learning with privacy" (Filip Granqvist, et al. 2020) explores how privacy-preserving learning can improve a speaker verification system by using privacy-sensitive speaker data to train an auxiliary classification model [5]. They combine federated learning and differential privacy mechanisms to protect user privacy while enabling learning on a large population of speakers. The auxiliary model predicts speaker characteristic labels considered useful as side information. This knowledge is then distilled into a speaker verification system using multi-task learning, resulting in a relative improvement in equal error rate over a baseline system.

Tanweer Alam, et al. 2022 in "Federated Learning and Its Role in the Privacy Preservation of IoT Devices" states that Federated Learning is crucial in developing AI and is one of the best ways to ensure privacy and provide security for a variety of applications [7]. The review paper also states the challenges of FL, meaning it is still not as perfect as we would want it to be. There are still more problems to be solved, research to be conducted and solutions to be found. A few challenges of FL include being unsure about exactly how much communication is necessary for FL, issues in federated networks such as scalability, heterogeneity, and privacy or practical issues where devices behave differently at different times.

The survey paper "Federated Learning Meets Natural Language Processing: A Survey" [8] discusses existing federated learning algorithms for various Natural Language Processing tasks algorithms starting from language modeling where most language models lack privacy. Stating that the Language model (LM) is widely used in various NLP tasks, the paper discusses text classification, sequence tagging, recommendation, health text mining as well as speech recognition which is the core of our paper. Ming Liu et al. 2021 describe speech recognition methods like dynamic time wrapping, modern end-to-end deep neural models, and Hidden Markov Models. There are mentions of different papers proposing different solutions and frameworks to solve different problems and make speech recognition more accurate, better trained, quality impact of non-IID distributions, and cost-efficient. They discuss the "Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition" (Yang et al. 2020) [9] paper's proposed scheme that is built on a quantum convolutional neural network (QCNN) composed of a quantum circuit encoder for feature extraction, and a recurrent neural network (RNN) based end-to-end acoustic model (AM) that takes advantage of the quantum learning progress to secure models and avoid privacy leakage attacks.

Speech Emotion Recognition (SER) is the recognition of human emotions through natural speech and can be beneficial for building intelligent systems [10]. Vasileios Tsouvalas, et

al. 2022 in "Privacy-preserving Speech Emotion Recognition through Semi-Supervised Federated Learning" discusses about besides being beneficial, the SER can also be a threat to our privacy at the same time. The proposed model in this paper is supposedly the first SER based on FL and is claimed to be much more effective and privacy-preserving than centralized SER.

Filtering the speech to stop our sensitive information from getting leaked is the prime focus of our work. One way to do this could be by training the Automatic Speech Recognition (ASR) model using a self-supervised learning feature so that the ASR can not recognize selective sensitive information such as digits, i.e. card or social security numbers [11]. The proposed approach by Yuchen Liu, et al. 2022 in "Preserve User Privacy for Automatic Speech Recognition" conducted experiments in a simulated environment with 70-80% promising outcomes.

III. Methodology

To ensure the privacy of voice data, we used two separate machines to listen to voice commands, process them, and only pass the crucial part of voice command on to the second device to ensure the execution of an order. For the first device, we used a Laptop as a prototype for our sound detector and an ESP32 for the second device. Furthermore, we used Google's speech-to-text library in Python SpeechRecognition [1] in addition to BERT as our Natural Language Processing (NLP) model for finding the key commands only using COSINE similarity cite.

Google's speech-to-text library, SpeechRecognition(SR), can extract sentences from acoustic voice signals. This API supports other APIs like CMUSphinx (an offline library), Google Cloud Speech API, and many more, ensuring an accurate speech-to-text most of the time.

BERT (Bidirectional Encoder Representations from Transformers) [2] is a pre-trained natural language processing model developed by Google. It uses transformer architecture to learn contextual relations between words in a text. BERT uses a bidirectional approach to process text. It keeps track of the preceding and following words when generating embeddings, which are high-dimensional vectors. BERT cosine similarity finds similarity between two sentences based on the cosine similarity of their BERT embeddings 2. On comparing, the model gives a similarity score ranging from -1 to 1, meaning opposite or similar to each other.

We employed both google's SR followed by BERT cosine similarity to classify if a voice command is relevant to the voice commands available. Afterwards, we pass the representation of the voice command to the mobile device of action, an ESP32. The ESP32 receives and generates the action , turning on a LED 1. In the diagram 1, the SR converts speech to text which is similarity checked by BERT for relevance. It then forwards to the base64 encrypter generating an encrypted text and sending it over the network to the target mobile device. The mobile device decrypts the message and acts.

IV. Evaluation and Results

Based on our model we evaluated results on the accuracy of action and it is seen that we had around 91% accuracy, the data table is given below. Among all the commands, our system could successfully work 91% of the time and failed to work only 9% of the time. It could successfully detect the necessary command and filter out the unnecessary speech which was the main goal of our work. As it is working as per our plan, the system can successfully provide more privacy, keeping our sensitive/private data safe from eavesdropping apps and websites.

V. Conclusions

The proposed system consists of 2 different machines, one that decides the relevancy of our speech and the main device our speech is dedicated to. This adds an extra layer of security and ensures the privacy of our sensitive information. The experiments conducted for this robust system gave promising results with 91% accuracy rate where our speech is able to be successfully filtered out or passed on to the main device.

A. Future Works

Although the results are promising, all our experiments were conducted in a specific environment and had certain limitations, due to which the system might lag in different environments. We plan to further polish it more to make it work perfectly in every situation.

References

[1] https://pypi.org/project/SpeechRecognition/ [10-5-23]

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" 2019 arxiv

[3] Mothukuri, V., Khare, P., Parizi, R. M., Pouriyeh, S., Dehghantanha, A., Srivastava, G. (2021). Federated-learning-based anomaly detection for iot security attacks. IEEE Internet of Things Journal, 9(4), 2545-2554.

[4] Latif, S., Khalifa, S., Rana, R., Jurdak, R. (2020, April). Federated learning for speech emotion recognition applications. In 2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN) (pp. 341-342). IEEE.

[5] Granqvist, F., Seigel, M., Van Dalen, R., Cahill, A., Shum, S., Paulik, M. (2020). Improving on-device speaker verification using federated learning with privacy. arXiv preprint arXiv:2008.02651.

[6] Guliani, D., Beaufays, F., Motta, G. (2021, June). Training speech recognition models with federated learning: A quality/cost framework. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 3080-3084). IEEE.

[7] Alam, T. and Gupta, R., 2022. Federated learning and its role in the privacy preservation of IoT devices. Future Internet, 14(9), p.246.

[8] Liu, M., Ho, S., Wang, M., Gao, L., Jin, Y. and Zhang, H., 2021. Federated learning meets natural language processing: a survey. arXiv preprint arXiv:2107.12603.

[9] Yang, C.H.H., Qi, J., Chen, S.Y.C., Chen, P.Y., Siniscalchi, S.M., Ma, X. and Lee, C.H., 2021, June. Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6523-6527). IEEE.

[10] Tsouvalas, V., Ozcelebi, T. and Meratnia, N., 2022, March. Privacy-preserving speech emotion recognition through semi-supervised federated learning. In 2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops) (pp. 359-364). IEEE.

[11] Liu, Y., Kapadia, A. and Williamson, D., 2022. Preventing sensitive-word recognition using self-supervised learning to preserve user-privacy for automatic speech recognition. Proc. Interspeech 2022, pp.4207-4211.

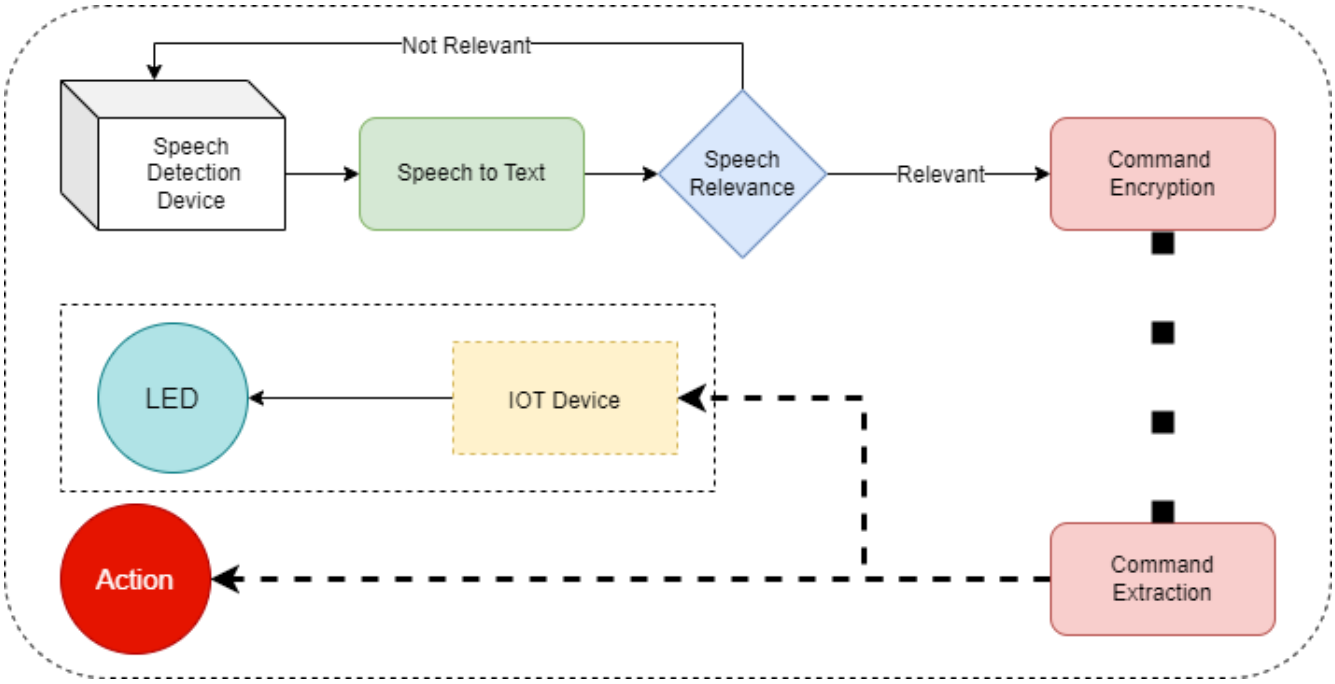

Fig. 1. Model Architecture

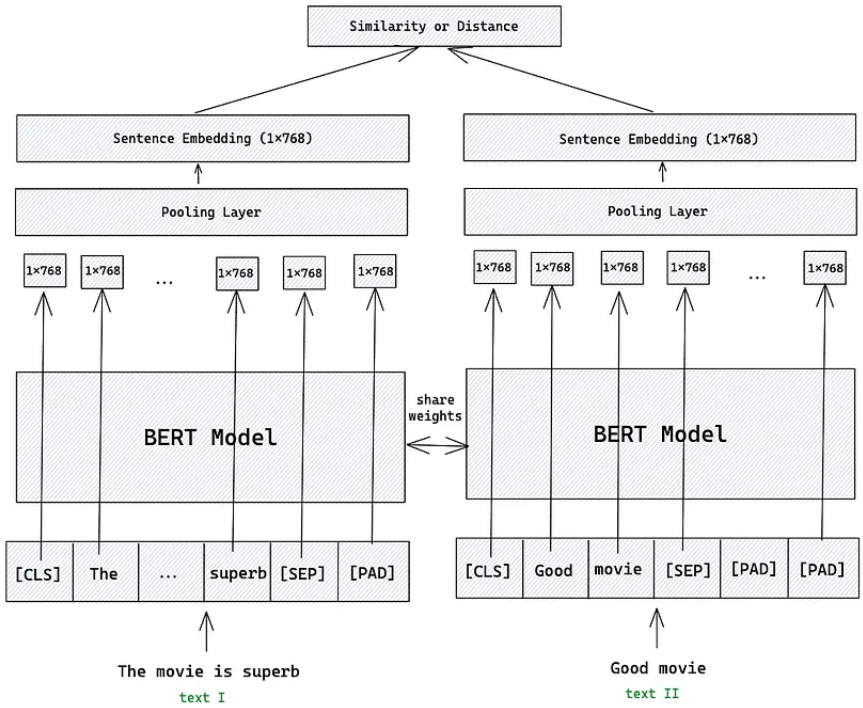

Fig. 2. Model Architecture

TABLE I
200 COMMAND DATASET LISTING

SL.	Command ”Turn the Lights On”	Results
1	Turn on the lights	Worked
2	Switch on the lights	Worked
3	Power on the light bulbs	Failed
4	Illuminate the room	Worked
5	Activate the light fixtures	Failed
6	Brighten up the space	Worked
7	Light up the room	Worked
8	Turn off the lights	Worked
9	Switch off the lights	Worked
10	Power off the light bulbs	Failed
11	Dim the lights	Worked
12	Deactivate the light fixtures	Failed
13	Darken the room	Worked
14	Turn up the brightness	Worked
15	Turn down the brightness	Worked
16	Increase the light level	Worked
17	Decrease the light level	Worked
18	Make the room brighter	Worked
19	Make the room darker	Worked
20	Adjust the lighting	Worked
21	Set the lights to full brightness	Worked
22	Set the lights to half brightness	Worked
23	Set the lights to low brightness	Worked
24	Set the lights to maximum brightness	Worked
25	Set the lights to minimum brightness	Worked
26	Turn the lamp on	Worked
27	Turn the lamp off	Worked
28	Turn the desk lamp on	Worked
29	Turn the desk lamp off	Worked
30	Turn the floor lamp on	Worked
31	Turn the floor lamp off	Worked
32	Turn the table lamp on	Worked
33	Turn the table lamp off	Worked
34	Turn the chandelier on	Worked
35	Turn the chandelier off	Worked
36	Turn the ceiling light on	Worked
37	Turn the ceiling light off	Worked
38	Turn the wall sconce on	Worked
39	Turn the wall sconce off	Worked
40	Turn the pendant light on	Worked
41	Turn the pendant light off	Worked
42	Turn the track light on	Worked
43	Turn the track light off	Worked
44	Turn the recessed light on	Worked
45	Turn the recessed light off	Worked
46	Turn the spotlight on	Worked
47	Turn the spotlight off	Worked
48	Turn the floodlight on	Worked
49	Turn the floodlight off	Worked
50	Turn the porch light on	Worked

TABLE II
200 COMMAND DATASET LISTING (CONTINUED)

SL.	Command ”Turn the Lights On”	Results
51	Turn the porch light off	Worked
52	Turn the garden light on	Worked
53	Turn the garden light off	Worked
54	Turn the patio light on	Worked
55	Turn the patio light off	Worked
56	Turn the driveway light on	Worked
57	Turn the driveway light off	Worked
58	Turn the garage light on	Worked
59	Turn the garage light off	Worked
60	Turn the hallway light on	Worked
61	Turn the hallway light off	Worked
62	Turn the bathroom light on	Worked
63	Turn the bathroom light off	Worked
64	Turn the bedroom light on	Worked
65	Turn the bedroom light off	Worked
66	Turn the closet light on	Worked
67	Turn the closet light off	Worked
68	Turn the living room light on	Worked
69	Turn the living room light off	Worked
70	Turn the dining room light on	Worked
71	Turn the dining room light off	Worked
72	Turn the kitchen light on	Worked
73	Turn the kitchen light off	Worked
74	Turn the family room light on	Worked
75	Turn the family room light off	Worked
76	Turn the study light on	Worked
77	Turn the study light off	Worked
78	Turn the basement light on	Worked
79	Turn the basement light off	Worked
80	Turn the attic light on	Worked
81	Turn the attic light off	Worked
82	Turn the garage door light on	Worked
83	Turn the garage door light off	Worked
84	Turn the back door light on	Worked
85	Turn the back door light off	Worked
86	Turn the front door light on	Worked
87	Turn the front door light off	Worked
88	Turn the porch ceiling fan light on	Worked
89	Turn the porch ceiling fan light off	Worked
90	Turn the living room ceiling fan light on	Worked
91	Turn the living room ceiling fan light off	Worked
92	Turn the bedroom ceiling fan light on	Worked
93	Turn the bedroom ceiling fan light off	Worked
94	Turn the bathroom exhaust fan on	Worked
95	Turn the bathroom exhaust fan off	Worked
96	Turn the kitchen exhaust fan on	Worked
97	Turn the kitchen exhaust fan off	Worked
98	Turn the range hood fan on	Worked
99	Turn the range hood fan off	Worked
100	Turn the air conditioning unit on	Worked

TABLE III
200 COMMAND DATASET LISTING (CONTINUED)

SL.	Command ”Turn the Lights Off”	Results
1	Turn off the lights	Worked
2	Switch off the lights	Worked
3	Power down the lighting	Worked
4	Deactivate the light fixtures	Worked
5	Shut off the bulbs	Worked
6	Cut off the illumination	Worked
7	Put out the lights	Worked
8	Turn down the brightness	Worked
9	Dim the lights	Worked
10	Set the brightness to minimum	Worked
11	Lower the lighting level	Worked
12	Turn the light switch off	Worked
13	Kill the lights	Worked
14	Stop the illumination	Worked
15	Turn the lamps off	Worked
16	Switch off the light bulbs	Worked
17	Put the lights out	Worked
18	Power off the light sources	Worked
19	Turn the illumination off	Worked
20	Cease the lighting	Worked
21	Shut down the light fixtures	Worked
22	Turn the brightness down	Worked
23	Cut the lighting off	Worked
24	Turn off the lighting fixtures	Worked
25	Switch off the lamps	Worked
26	Power off the lighting system	Worked
27	Deactivate the lamps	Worked
28	Turn the room lights off	Worked
29	Stop the light flow	Worked
30	Turn off the bulbs	Worked
31	Switch off the light fixtures	Worked
32	Put the bulbs out	Worked
33	Power off the room lights	Worked
34	Turn the lighting down	Worked
35	Stop the luminance	Worked
36	Turn the light down	Worked
37	Put the lamps out	Worked
38	Cut the light sources off	Worked
39	Shut off the lighting fixtures	Worked
40	Turn the light sources off	Worked
41	Switch off the light sources	Worked
42	Power off the bulbs	Worked
43	Deactivate the room lights	Worked
44	Turn off the light switches	Worked
45	Shut the light bulbs off	Worked
46	Cut the light fixtures off	Worked
47	Stop the lights	Worked
48	Turn off the light	Worked
49	Switch the lights off	Worked
50	Power off the lamps	Worked

TABLE IV
200 COMMAND DATASET LISTING (CONTINUED)

SL.	Command ”Turn the Lights Off”	Results
51	Put off the light	Worked
52	Turn off the light bulbs	Worked
53	Deactivate the lighting	Worked
54	Cease the luminance	Worked
55	Turn off the light source	Worked
56	Stop the light fixtures	Worked
57	Shut the lamps off	Worked
58	Cut off the lamps	Worked
59	Turn the light fixture off	Worked
60	Switch off the room lights	Worked
61	Shut down the lighting system	Failed
62	Put out the lights	Worked
63	Cut off the lights	Worked
64	Switch the room light off	Worked
65	Turn the room lights off	Worked
66	Switch off the bedroom light	Worked
67	Turn the kitchen light off	Worked
68	Turn off the bathroom light	Worked
69	Turn off the living room light	Worked
70	Turn off the dining room light	Worked
71	Turn off the garage light	Worked
72	Turn off the porch light	Worked
73	Turn off the hallway light	Worked
74	Turn off the study light	Worked
75	Turn off the closet light	Worked
76	Power down the room light	Failed
77	Power off the kitchen light	Failed
78	Power off the bathroom light	Failed
79	Turn the room lamps off	Worked
80	Switch off the living room light	Worked
81	Turn off the bedroom lamps	Worked
82	Turn off the kitchen lamps	Worked
83	Turn off the bathroom lamps	Worked
84	Turn off the dining room lamps	Worked
85	Turn off the garage lamps	Worked
86	Turn off the porch lamps	Worked
87	Turn off the hallway lamps	Worked
88	Turn off the study lamps	Worked
89	Turn off the closet lamps	Worked
90	Power down the living room light	Failed
91	Power down the bedroom light	Failed
92	Power down the kitchen light	Failed
93	Power down the bathroom light	Failed
94	Power down the dining room light	Failed
95	Power down the garage light	Failed
96	Power down the porch light	Failed
97	Power down the hallway light	Failed
98	Power down the study light	Failed
99	Power down the closet light	Failed
100	Switch off the office light	Worked