Augmenting Conversational Agents with Ambient Acoustic Contexts

Chunjong Park*
University of Washington
cjparkuw@cs.washington.edu

Chulhong Min
Nokia Bell Labs
chulhong.min@nokia-bell-labs.com

Sourav Bhattacharya*
Samsung AI Center, Cambridge
sourav.b1@samsung.com

Fahim Kawsar
Nokia Bell Labs
fahim.kawsar@nokia-bell-labs.com

ABSTRACT

Conversational agents are rich in content today. However, they are entirely oblivious to users' situational context, limiting their ability to adapt their response and interaction style. To this end, we explore the design space for a context augmented conversational agent, including analysis of input segment dynamics and computational alternatives. Building on these, we propose a solution that redesigns the input segment intelligently for ambient context recognition, achieved in a two-step inference pipeline. We first separate the non-speech segment from acoustic signals and then use a neural network to infer diverse ambient contexts. To build the network, we curated a public audio dataset through crowdsourcing. Our experimental results demonstrate that the proposed network can distinguish between 9 ambient contexts with an average F_1 score of 0.80 with a computational latency of 3 milliseconds. We also build a compressed neural network for on-device processing, optimised for both accuracy and latency. Finally, we present a concrete manifestation of our solution in designing a context-aware conversational agent and demonstrate use cases.

CCS CONCEPTS

• Human-centered computing → Ubiquitous and mobile computing; **• Computing methodologies** → *Neural networks.*

KEYWORDS

Conversational agents; Acoustic ambient context

ACM Reference Format:
Chunjong Park, Chulhong Min, Sourav Bhattacharya, and Fahim Kawsar. 2020. Augmenting Conversational Agents with Ambient Acoustic Contexts. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '20), October 5–8, 2020, Oldenburg, Germany.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3379503.3403535

1 INTRODUCTION

Conversational agents [1] are now pervasive, integrated into mobile phones, smart speakers, and even in cars. Remarkable advancement of machine learning is causing a seismic shift, in that conversational agents are now able to understand human speech and transform text into speech in a similar way to humans [36] in everyday living spaces (even on-the-go). Naturally, this created interminable possibilities, uncovering novel, productive and useful experiences with conversational agents for accessing and interacting with digital services in many and diverse applications including HCI [27], customer experience [30], conversational commerce [31], medicine [38, 39], entertainment [21], education [29], and social work [6].

For long, context-aware computing research has attempted to understand situational awareness from acoustic signals, e.g., surrounding events [23, 25] and human speech emotion [13]. Unfortunately, the implications of this research in our everyday experience are still limited. None of the commercial-grade conversational agents (Alexa, Siri, Google, Cortana, etc.) today can react and adapt to users' situational context. We argue that simple adjustments of the interaction style of the agents' responses can increase users' conversational experience with these agents. For instance, ambient context information retrieved from audio signals can be used to respond to users in a more calming or alarming manner, in a quieter or louder fashion. It is also possible to adapt the agents' responses based on a user's acoustic context. When the agents are asked to provide a recipe, they can provide a full recipe when a user is cooking at a kitchen or an ingredient list when a user is in a grocery store. Such non-speech contextual information would provide important cues to enable conversational agents to provide more customised and fulfilling experiences across different environments.

The development of conversational agent systems to understand users' acoustic environments in real-time has proven challenging. Identifying a proper moment for inferring acoustic context in-the-wild in uncontrolled environments is an incredibly difficult problem. The moment the should be long enough for the accurate recognition of ambient contexts, but also short enough to provide responsive responses. With the study, we discover an opportunity of a 1-second-long pause naturally made between a wakeup word and a user query. However, although numerous audio datasets are available, most of them are not suitable for ambient context analysis. Moreover, the latency requirement of these systems for faster responses makes it extremely hard to deploy such systems in the real-world.

*This work was done when these authors were affiliated with Nokia Bell Labs.

[1] Here, conversational agents refer to voice assistants such as Alexa, Siri, Cortana, etc.

MobileHCI '20, October 5–8, 2020, Oldenburg, Germany

ACM ISBN 978-1-4503-7516-0/20/10…$15.00
https://doi.org/10.1145/3379503.3403535

To this end, we first systematically explore the design space for a *context augmented conversational agent*, including analysis of input segment dynamics and computational alternatives. We identify an opportunity for seamlessly recognising ambient contexts without affecting user experience. Then, we present a light-weight purpose-built deep neural network solution for ambient context analysis. The proposed solution consists of an intelligently designed input segment for capturing audio data, and a neural network model that uses audio embedding generated by VGGish model [15]. For training this network, we leverage Google's AudioSet that contains 5.8 thousands of hours of audio with 527 different event classes from white noise to animal sounds. Despite its richness and diversity of content, this dataset cannot be directly used for the purpose of ambient audio analysis on a short-duration signal due to its poor annotation granularity, i.e., 10 seconds. To address this caveat, we adopt a crowdsourcing-based and quality-aware annotation strategy and transform a subset (60K) of this dataset containing ambient audio events with a duration of 1-second and accurate annotation. On top of the trained model, we prototype an ambient context-augmented conversational agent as a concrete manifestation of our solution and showcase several use cases.

Building on this data, our model achieves an F_1 score of 0.80 and an inference latency of 3 ms on NVidia DGX Station and 360 ms on Raspberry Pi 3+. We further train a distilled network optimised for both recognition accuracy and latency and achieve an F_1-score of 0.73 with inference latency of 1 ms on NVidia DGX Station and 10 ms on Raspberry Pi 3+. Combining these and the rest of our results, we show that it is possible to infer ambient audio context (e.g., crowd sound, background chatter sound, and footstep sound) at a fraction of the time required for traditional conversational agents, thus allowing their incorporation without compromising agents' responsiveness.

The main contributions of this work are as follows:

- We demonstrate that it is possible to infer ambient acoustic contexts under extreme latency requirements using a light-weight deep neural network. This ability uncovers a unique opportunity for conversational agents to augment their contextual awareness and adapt behaviour.
- We offer a well-curated dataset suitable for ambient acoustic context analysis. The refined temporal granularity of this dataset enables most popular audio models to be more fine-tuned, whose input signatures are often above one second. As such, it opens up brand-new avenues for the audio-based interactive system design.
- We present a concrete manifestation of our solution in designing a context-aware conversational agent and demonstrate use cases on top of the prototype system.

In what follows, we present the overall design space and describe the dataset and its construction method. Next, we provide an in-depth technical view of our context models. We present the evaluation of the system and a prototype application. Then, we revisit the related past research before concluding the paper.

2 DESIGN CHALLENGES

Augmenting conversational agents with ambient contexts for contextual adaptation puts forward a set of challenges. Some are with data, some are with underlying systems, and some are with user experience. In this section, we reflect on these challenges and present our design decisions that underpin this work.

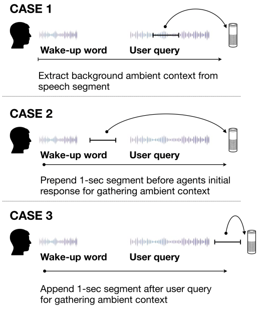

CASE 1
Wake-up word
User query
Extract background ambient context from speech segment
CASE 2
Wake-up word
User query
Prepend 1-sec segment before agents initial response for gathering ambient context
CASE 3
Wake-up word
User query
Append 1-sec segment after user query for gathering ambient context

Figure 1: Different alternatives for designing input segments for ambient context recognition.

2.1 Data Challenges

Modelling acoustic ambience from unconstrained audio signal inherently depends on quantity, quality and diversity of data. These characteristics are particularly critical for representation learning techniques, such as a deep neural network. Besides, the quality of models depends on specific properties, including temporal granularity of labels for time-varying audio signals, and the balance of classes in the data. For conversational agents to be aware of users' ambient contexts, these facets essentially translate into the challenge of constructing a dataset containing rich and balanced ambient context events with labels of fine temporal granularity. In Section 3.1, we discuss a dataset that we have designed taking these considerations into account.

2.2 System and User Experience Challenges

Augmenting ambient contexts demands systematic and careful refinements of data processing and presentation. Conventional pipeline for a conversational agent is purposefully designed for automatic speech recognition, language processing, and information delivery over voice [4, 7, 12, 14, 44]. Moreover, user experience depends heavily on the responsiveness of the agent, and as such accurate and low-latency recognition and response have always been the critical design priority in the development of conversational agents. Naturally, ambient context recognition adds complexity to this pipeline, and in particular, this augmentation must not come at the expense of the degradation of primary functional attributes. To this end, we observe three key challenges regarding the system and user experience:

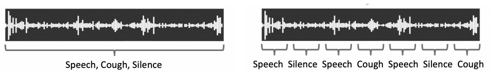
Input segment placement: Understanding contexts from an audio segment requires analysis of the background acoustics. This analysis can be performed on the original speech segment spoken by a user (i.e., voice command) or can be intelligently placed before or after the user's speech. These alternatives are illustrated in Figure 1. Analysis of ambient contexts in the presence of speech (Case 1) is naturally computationally heavy and challenging to the inference model due to the mixed signal. On the other hand, such context analysis in the absence of speech offers a comparatively cleaner signal (Case 2 or 3). These observations suggest that we can extend the input recording sequence to intelligently include a short recording segment that might be enough to capture audio signals for context analysis accurately. This inclusion could happen before a user query (Case 2) or at the end of a user query (Case 3). The latter, however, inherently increases the response latency, thereby degrading user experience significantly. Given that modern conversational agents are triggered with a wake-up keyword and provide visual and vocal feedback, it is natural to extract the input segment during the pause that a user makes to wait for the notification and speak the query (Case 2); for example, Alexa responds to wake-up keywords with a visual notification. We omit the case when the agent continuously performs ambient context recognition in the background because it introduces severe privacy concerns and incurs nontrivial system cost for battery-powered devices.

Input segment duration: Several past studies reported the dynamics of delay and user experience concerning user interaction with conversational agents [40]. A variety of factors contribute to these dynamics, including perceived humanness of the agents, user satisfaction, system operation delay, etc. Building on this research, our premise is that the input segment designed for ambient context analysis needs to be minimal without compromising user experience. At the same time, this segment needs to contain enough information to understand the ambient context reliably. We empirically observed that a one-second delay is enough to capture audio for ambient context with negligible impact on user experience (See Section 2.3.)

Input segment processing: This aspect begs two questions - 1) the placement alternative of the processing, and 2) its implications both on computation and user experience. On the former, while the natural choice is to push the computation for the ambient context recognition to the cloud where agents execute the speech recognition and following operations, increasingly we are observing the emergence of on-device modelling of audio data [11, 22]. Naturally, the on-device approach, i.e., recognising ambient contexts on the conversational agent device, offers faster execution and lower latency; however, the adaptation of the actual response is limited to its interaction style. On the other hand, cloud processing incurs additional latency due to transport and remote processing, but it offers a better opportunity for adapting not only interaction style but also the content of the interaction. It is important to note that our design choice of taking the input segment between the wake-up word and user query as in Case 2 naturally resolves the latency issue. On user experience, it is evident that the on-device approach preserves user privacy significantly better than cloud processing approach. As most conversational agents execute speech recognition and consequent actions in the cloud, in this work, we have taken the remote processing route for ambient context recognition as if it offers maximum flexibility for content adaptation. However, we also show that we can port our technique easily to an on-device setting, especially with the distillation method (explained in Section 5.3.1) we have adopted in our solution.

2.3 Opportunity for Sensing Ambient Context

We quantify the opportunity for ambient context recognition between a wake-up keyword and a query (Case 2), i.e., how long will the pause between a wake-up word and a user query be in real-life situations. For the study, we recruited 10 participants (4 females, 6 males, 24 – 32 years old), who are already using conversational agents at their homes on a daily basis. We asked them to use Amazon Alexa to naturally input 5 different commands (weather, time, funny joke, playing music, and set a timer). We recorded their interactions and measured the interval between the wake-up word ("Alexa") and the query. From 50 commands in total, the interval was 1.29 seconds on average (SD: 0.36, min: 0.81 max: 2.19). Assuming a Gaussian distribution with the above mean and standard deviation, we expect that there will be longer than 1-second intervals for 80% of the time.



(a) Weak labels **(b) Weak labels**

Figure 2: Example of weak labels and strong labels.

Events	Sub-events	Clips
Crowd	Crowd (1,218); Chatter (158); Hubbub (227)	1,603
Applause	Applause (360); Clapping (240)	600
Laughter	Laughter	530
Typing/Clicking	Typing (315); Clicking (88)	403
Door	Door (113); Knock (180)	293
Silence	Silence	251
Television	Television	239
Walk	Walk	220
Speech	Speech (28,671); Female speech (136); Male speech (124); Conversation (120)	29,051
Others	Non-target; unidentifiable events	24,081

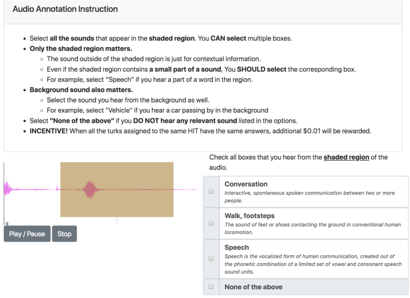
Table 1: Distribution of audio segments after cleaning.

3 AMBIENT ACOUSTIC DATA

3.1 Dataset

To train a model for the recognition of acoustic contexts, we leverage the AudioSet [10] released by Google in 2017. It is a large-scale set of 10-second-long audio segments and associated labels extracted from 2.1 million YouTube videos. In total, it contains 5.8 thousand hours of audio and 527 classes annotated by human experts.

However, it is not straightforward to build a responsive audio model to recognise an ambient event with 1-second audio clip using the AudioSet data due to its poor annotation granularity, i.e., events are *weakly* labelled to the 10-second clip as shown in Figure 2a. While acoustic events occur intermittently and shortly within a 10-second-long audio clip, which is natural considering the characteristics of acoustic events, they are associated with the whole segment without temporal information. Thus, segmenting a 10-second clip into ten 1-second ones and labelling all 1-second clips as the original label would not work.



Audio Annotation Instruction

- Select **all the sounds** that appear in the **shaded region**. You **CAN select** multiple boxes.
- **Only the shaded region matters.**
  - The sound outside of the shaded region is just for contextual information.
  - Even if the shaded region contains **a small part of a sound**, You **SHOULD select** the corresponding box.
  - For example, select "Speech" if you hear a part of a word in the region.
- **Background sound also matters.**
  - Select the sound you hear from the background as well.
  - For example, select "Vehicle" if you hear a car passing by in the background
- Select **"None of the above"** if you **DO NOT hear any relevant sound** listed in the options.
- **INCENTIVE!** When all the turks assigned to the same HIT have the same answers, additional $0.01 will be rewarded.

Check all boxes that you hear from the **shaded region** of the audio.

□ **Conversation**
Interactive, spontaneous spoken communication between two or more people.

□ **Walk, footsteps**
The sound of feet or shoes contacting the ground in conventional human locomotion.

□ **Speech**
Speech is the vocalized form of human communication, created out of the phonetic combination of a limited set of vowel and consonant speech sound units.

□ **None of the above**

Figure 3: Amazon Mechanical Turk Interface. Annotation instructions, audio waveform, and descriptions for each response option are designed to help annotators perform the task correctly.

To this end, we present a new dataset which is refined based on the AudioSet with *strong* labelling, i.e., annotating the event with a 1-second resolution, as shown in Figure 2b. To the best of our knowledge, our dataset is a first-kind-of-dataset of acoustic events with the 1-second temporal resolution. We initially select 17 target events (13 ambient contexts and 4 speech events) out of 527 classes, which are relevant to indoor environments where conversational agents are usually deployed (See the sub-events column in Table 1 for the details); we added the speech events for the comparative study. Then, we group 13 ambient sub-events into 9 events based on the ontology provided by the AudioSet, a hierarchy of audio collections, and map the events to semantic contexts, e.g., television to living room, crowd to social situation, typing/clicking to work. We leverage such semantic for the adaptation of the response and interaction style of conversational agents.

3.2 Data Collection Methodology

We segment 10-second-long audio clips into 1-second-long ones and employ a crowdsource approach to correctly annotate each 1-second segment. We use the Amazon Mechanical Turk as a crowdsourcing platform. Figure 3 shows the user interface of the task. In the audio player at the bottom-left section, Turkers can easily play, pause, and stop the audio clip. We also provide 0.5 seconds of audio before and after the target 1-second to provide contextual information of the clip. Response options are obtained from the classes belong to the original 10-second-long clip and 'None of the above' is added. To avoid ambiguity from Turkers, we also provide a definition of each event label.

To assure the quality of responses, we recruited three Turkers for each 1-second clip and selected the label that reaches a majority agreement. For rewards, we provided $0.01 per task and additional $0.01 when all Turkers reach unanimous agreement to encourage them to accurately perform the task. 59,287 tasks were done by Turkers in 5 days; for each task, the average time taken was 14.2 seconds (min= 3, max= 59). As a result, 53.6% of clips are agreed by all three Turkers and 43.0% of them are agreed by two. Only 3.4% of clips have a disagreement with all Turkers and are excluded. In total, we have around 57,000 valid segments.

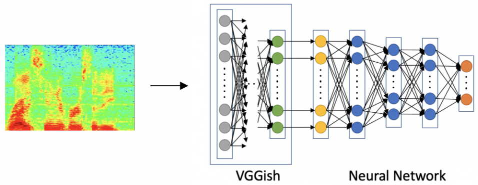


Figure 4: Network architecture for understanding acoustic ambient contexts. Mel-log spectrogram is converted into 128-dimensional embedding by VGGish audio encoder. The embeddging is then fed into 5 fully-connected layers.

We release the refined dataset [2]. Target audio files from AudioSet are split into 1-second clip and each 1-second clip has its corresponding labels obtained from crowd workers. The structure and filename convention follows AudioSet's in which each audio file has its own directory named by ID. In each directory, audio file and label are stored. Since we segment the original 10-second clips into 1-second ones, we have around 10 audio files with labels in each directory. We also include a binary npy file that has log-mel spectrogram, audio embedding generaged by VGGish encoder [15], and labels for all audio files. In addition, we release the inteface code (Figure 3) and example data for Machanical Turk to help researchers easily build a new dataset using crowdsourcing.

Note that our main purpose is to demonstrate the end-to-end process of building augmented conversational agent. A different set of target events can be selected and used in this pipeline for different purposes. For example, to target home environment, domestic sound events such as flushing, water running, and door bell can be selected as target events. One can select the sound events of their interests from AudioSet. Then, they can upload audio annotation task using the released Mechanical Turk interface and get their own curated dataset for the specific purpose.

4 RECOGNISING AMBIENT CONTEXTS

4.1 Recognition Model

We present the design and implementation of the ambient context recognition model. It consists of two main components, VGGish audio encoder and a neural network classifier as shown in Figure 4. Since the state-of-art models [9, 35] for speech activity detection show high accuracy in distinguishing between speech and non-speech sound, in our system, we employ the speech activity detection model to first filter non-speech sound to the context recognition model.

Audio encoder: For audio encoding, we use the VGGish model [15] that generates a 128-dimensional embedding as a feature extractor. Audio clips are resampled to 16 kHz mono and converted into log-mel spectrogram by computing Short-Time Fourier Transform with a window size of 25 ms, a window hop of 10 ms, and periodic Hann window and mapping it to 64 mel bins to cover 125-7500 Hz. The VGG model is modified to fit the dimension of the spectrogram (96x64) and has four conv/maxpool layers, and 128-wide fully connected layer. The modified VGG model, so called VGGish model, outputs 128-dimensional embeddings for 1-second audio clip.

[2] https://www.esense.io/datasets/ambientacousticcontext/index.html

Classifiers	Precision	Recall	F_1 score
Baseline	0.692	0.589	0.636
Our model	0.796	0.804	0.800

Table 2: Classification performance.

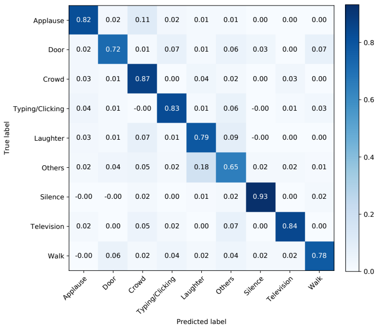


Figure 5: Confusion matrix of the target events.

Classifier: We then build a classifier that takes a 128-dimensional embedding as input to detect an ambient event happening in the audio clip. Because the input dimension is relative low, i.e., 1x128, we design a neural network that has five fully connected layers, input layer in the beginning, and output layer at the end, with batch normalisation and LeakyReLu ($\alpha = 0.01$) between each layer. To prevent from over-fitting, we add two dropout layer ($p = 0.2$) between 2nd and 3rd layer, and 3rd and 4th layer, and apply early stopping when validation loss reaches minimum point to prevent the model from overfitting. During the training, learning rate of 0.001 and batch size of 256 are used.

4.2 Recognition Performance

Experimental setup: We present the recognition performance of ambient contexts. We first filter out speech events from the dataset in Table 1 and split the complementary set (i.e., non-speech events) into the train, validation, and test set with a ratio of 70%/15%/15%. We used validation set for early stopping and hyper-parameter searching. The test set is only used for the final performance evaluation. Since the number of events labelled *other* is extremely larger than the number of *other* events, we randomly choose 500 clips for *other* event. As a baseline, we consider a single-integrated classifier that distinguishes all acoustic events (including ‘speech’) at once (i.e., a model tries to classify sound events using the input audio segments for both speech and other ambient contexts). Similarly, we split the whole dataset into the train and test set with the same ratio. We further evaluate the performance of the compressed model on the non-speech events to compare latency and accuracy to original model’s performance.

Classifier performance: The experimental results validate our design choice of extracting the input segment between the wake-up word and user query (i.e., Case 2 in Figure 1). Table 2 shows its overall performance. With a given non-speech audio segment, our model distinguishes between 9 ambient contexts with an average F_1 score of 0.800. However, when the *speech* event is mixed to the ambient context and needed to be additionally distinguished, F_1 score decreases to 0.636.

Figure 5 shows the normalised confusion matrix of our model. Overall, the model shows reasonable performance across all events. However, as expected, the precision and recall for the *other* events is lower than those for the rest of events. This is mainly because the *other* events consist of a number of miscellaneous types of acoustic events, thereby not having its own unique signal characteristic. We can also observe that the events of applause, crowd, and laughter are relatively more confused than across other events because they are all made from human voice.

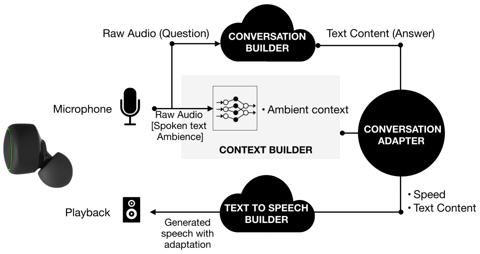


Figure 6: Prototype conversational agent with earables.

5 PROTOTYPE

We present a concrete manifestation of the model in a prototype conversational agent that dynamically adjusts its conversation style and content in response to users context gathered through ambient sound signatures. On top of the system, we showcase several use cases.

5.1 Conversational Agent

The prototype system is composed of the following components as illustrated in Figure 6:

- **Conversation Builder:** This component enables a user to interact with the agent using a predefined dialogue base. For this prototype, we have used Dialogflow [1] populated with a set of situation-specific dialogues.
- **Conversation Adapter:** This component is responsible for guiding the adaptation strategy for the agent’s response corresponding to a user’s ambient context, taking into account the output of the context builder and a data-driven rule engine. We have devised a set of simple adaptation rules as a proof-of-concept to adapt agent responses by changing the tone, volume, response delay and content (See Section 5.2). These rules are not exhaustive rather a simple demonstration of the application of our solution.
- **Text-to-Speech Builder**: This component is responsible for synthesising the agent’s response in a voice that accurately reflects a user’s situation using IBM Bluemix Voice service [2]. This synthesis process interplays various voice attributes, e.g., pitch, rate, breathiness, glottal tension etc. to transform agents voice according to the rule of the conversation adapter.

These components and the end-to-end system are realised with an eSense earable [19] (the source of sensory signals and playback) and an Android smartphone (service platform). For the conversational adapter, we considered two platforms, NVidia DGX with Tesla V100 and Raspberry Pi 3+, each of which represents cloud and on-device processing, respectively, as mentioned in Section 2.2.

5.2 Use Cases

We illustrate use case scenarios to show how ambient context understanding enhances the interaction with conversational agents. Then, we present the adaptation rules we applied for.

Privacy-preserving response: People often check notifications or read messages using conversational agent, especially when users' hands are occupied. However, notifications and messages often contain private sensitive information and are not suitable to broadcast when other people are around. Conversational agents with ambient context understanding can read aloud the notifications when there is no one around; if it detects there are other people, it can ask back to the user to confirm to reading aloud the notification.

Adaptive content: Imagine a user wants to find a recipe for lasagna. The information that she needs will be different depending on her current situation. For example, when she is outside her home, she needs an ingredient list for grocery shopping. When she is at kitchen, she needs directions for cooking lasagna. A conversational agent, recognising her current context, can provide ingredients-focused information if she is outside or directions-focused information if she is at home. In another example, a user wants to play music using conversational agent. If the agent detects that crowd is around, it can play upbeat music and make the volume high. On the other hand, if the agent detects no one's around, it can play his usual playlist with usual volume setting.

Interaction with environment: Users' command to conversational agents can conflict with the current environment. For example, a user can ask to play music while TV is turned on. If the agent does as user's command, the outcome might not be the most desirable since the sound from TV and music clashes. The agent with ambient context understanding can detect whether TV is on. It can ask the user to turn off the TV or it can turn off the TV if the TV is connected to the agent.

5.3 Implementation

5.3.1 Model Compression. The above encoder and classifier configuration is optimized to yield high accuracy, but requires larger memory footprint and higher latency. While this configuration is purposely built for the cloud, it is not ideal for preserving users' privacy, especially when sensitive audio data are transferred to the cloud. On-device processing can preserve users' privacy by keeping the sensitive data within the device. However, limited computation power on an embedded device often result in increased inference time and poor user experience.

As a possible solution for the optimisation of on-device processing, we present the knowledge distillation-based [16] model compression in which two neural networks are involved, called *teacher* and *student* networks. Its intuition is to train a smaller-size network that mimics the outputs generated by the original network. We use the autoencoder and classifier configuration as our *teacher* network. We adopt cnn-low-latency [34] as our student model, since it takes audio as an input and shows reasonable performance in keyword spotting. cnn-low-latency consists of one convolution layer, two fully connected layers, and softmax as an activation function.

```
Algorithm 1: Rule adaptation based on ambient context
Function main():
    context = classify($\text{audio}_{\text{ambient}}$)
    query = speech_to_text($\text{audio}_{\text{speech}}$)
    if is_privacy_sensitive(query) && context == CROWD
     then
        Ask user whether to perform
        wait_for_confirmation()
        if not confirmed then
            return
    response, volume = query_to_response(query, context)
    play_audio(response, volume)
Function query_to_response(query, context):
    if is_asking_information(query) then
        response = information based on context
    else if is_activating_devices(query) then
        if other_devices_running then
            Ask user whether to turn off other devices
            wait_for_confirmation()
            if confirmed then
                response = perform_task(query)
    else
        response = perform_task(query)
    volume = current_noise_level(context)
    return response, volume
```

Classifiers	Latency (ms) (Nvidia DGX)	Latency (ms) (Raspberry Pi 3+)	F1-score
Original model	3.4 (SD: 0.30)	364.3 (SD: 8.30)	0.80
Compressed model	1.2 (SD: 0.44)	10.3 (SD: 3.97)	0.73

Table 3: Performance of teacher and student models.

5.3.2 Rule Adaptation. We present our implementation of the rule adaptation. As described in the example in Algorithm 1, the agent checks whether the query is privacy sensitive; if so, the agent confirms with users whether to proceed. Based on the use cases described in Section 5.2, the agent then provides information based on user's context, interact with other devices to provide better quality of service, or perform tasks normally if there is no context specific requirement.

5.4 Micro benchmark

Compressed classifier performance: Table 3 shows the inference latency on two platforms, NVidia DGX with Tesla V100 and Raspberry Pi 3+, and accuracy of the teacher (i.e., original) and student (i.e., compressed) network. Surprisingly, the latency decreases 3.4 ms (original) to 1.2 ms (compressed), i.e., almost 3x reduction. In Raspberry Pi 3+, the difference becomes even larger. The inference time of the original network is 364.3 ms, whereas that of the compressed network is 10.3 ms. As the models run on CPU in Raspberry Pi 3+, the performance gain from the compression is more outstanding than in GPU. The accuracy decreases by 0.066, from 0.800 (original) to 0.734 (compressed), but considering the latency benefit, we believe the model compression can be a reasonable solution to enable on-device processing of the ambient context recognition.

Adaptation latency: We measure the adaptation latency, i.e., the time to be taken to apply for the adaptation rule. Since the number of rules is relatively lower, in the range between 3 and 5, the latency was negligible on both platforms, i.e., under 1 ms.

6 DISCUSSION

As an initial attempt to realise ambient context-augmented conversational agents, in this work, we primarily aimed at systematically exploring the design space of such agents and proposing a lightweight purpose-built deep neural network solution for ambient context analysis. Below, we discuss the implications and limitations of our work.

Applicability in other scenarios: This work leverages a short pause for detecting ambient acoustic context, naturally made between a user's wakeup word and query. It opens up a new opportunity for ambient context sensing in various conversation scenarios. For example, beyond a simple question and answer, Google's Duplex [3] and Meena [3] can perform naturalistic open-domain conversation with users. Hwang et al. envisioned interpersonal assistants that monitor ongoing human-to-human conversations and offer useful services just-in-time [17]. We can easily expect that those agents can utilise ambient acoustic contexts to improve user experiences in a similar way to the use cases we presented in Section 5.2. Considering that there would exist multiple, natural pauses between speech turns, they can achieve more accurate and robust recognition results.

Optimising model performance: In this paper, we did not tackle the model performance, i.e., recognition accuracy of ambient acoustic contexts, as a main problem. However, achieving higher accuracy would be essential to guarantee high degree of user experiences and make our system to be accepted by users. We discuss two possible solutions. First, we can adopt other public audio datasets, e.g., FreeSound[4] and DCASE challenge[5], for enriching the training data. These datasets also have similar limitations to AudioSet, poor annotation granularity and weak labelling, but we expect that they can be easily converted with a crowdsourcing approach using the tool we release. Second, we can further augment the train set by combining various sound effects [24] and noise signals [25]. We leave it as future work.

Real-life deployment: To deeply investigate our prototype in real-life situations, we plan to deploy the agents and application we implemented and conduct a user study. We expect to evaluate the overall performance of our proposed model in real-world settings. Also, by comparing user perception between conventional and context-augmented agents, we expect to unveil interesting aspects of conversational agents, such as usability of context augmented agent, user tolerance to possible failure (wrong suggestions), and presence of uncanny valley.

7 RELATED WORK

7.1 Acoustic Event Classification

There have been many research efforts on understanding activities and ambient contexts from audio signals. The AmbientSense application [32] processes audio signal from smartphones and showed reasonable performance on classifying 23 context of daily life. SoundSense [26] detects multiple speech, music and ambient sound categories based on mobile platforms. Rossi et al. [33] showed potential of using MFCC features on Gaussian Mixture Model (GMM) to recognize contexts using Freesound dataset, which is a popular public dataset that contains 120,000 annotated audio clips. In recent years, deep neural network has been adopted in audio sensing. Lane et al. presented DeepEar [22] that classifies high-level activities using a lightweight deep neural network. Since Google released AudioSet [10] that contains 5.8 thousand hours of audio extracted from YouTube videos with 527 sound events, numerous efforts have been introduced to build deep learning model for audio understanding. Hershey et al. [15] presented a CNN architecture for large scale audio classification on AudioSet, achieving mean precision of 0.381 on over 400 sound events using ResNet-50. Kong et al. [20] and Yu et al. [42] employed various attention model based techniques to improve the classification accuracy. Lupta et al. [23] and Liang et al. [25] fine-tuned the existing pre-trained model [15] for various domestic events using augmented data. In fine-tuning stage, they augmented the existing AudioSet with various sound effects or random noise for more robust training set. These efforts has focused on building an accurate model to recognise various acoustic events from feature-based machine learning to deep neural network. In this work, we target the augmentation of conversational agents as a main problem and identify its unique design requirements for the recognition of ambient contexts, i.e., recognising ambient contexts with 1-second audio signal. To this end, we offer a partial AudioSet dataset well-curated through crowdsourcing and present a purposefully-built recognition model of ambient acoustic events by adopting the existing audio sensing techniques.

7.2 Conversational Agents with Context

Research community has studied on various aspects of conversational agents as increasing number of text-based chatbots and smart devices with voice interface has been adopted in everyday life. Cohen et al. [8] pointed out that recognizing users attention is an important factor to improve current conversational agents. Some [28, 43] focuses on understanding user needs and evaluating user satisfaction. Yang et al. [41] has studied user's affective experiences with the conversational agents. Also, as argued in Section 2, most of the attempts on building conversational agents have been focused on building the pipeline for automatic speech recognition, language processing, and information delivery over voice [4, 7, 12, 14, 44]. Ongoing efforts to build contextual chatbots [18, 37, 45] focus on retrieving and understanding various contexts from text-based conversations. However, augmenting acoustic ambient contexts to adapt the response and interaction style has not been explored much in the research community, even though

[3] https://ai.googleblog.com/2018/05/duplex-ai-system-for-natural-conversation.html
[4] https://annotator.freesound.org/
[5] http://dcase.community/

conversational agents' input can be both user's voice and contextual information [5]. In this work, we argue that the recognition of ambient acoustic contexts will play an important role to augment ambient contexts to the conversational agents and propose design challenges and the core component, the ambient context recognition.

8 CONCLUSION

In this work, we systematically explored the design space for a context augmented conversational agent, including analysis of input segment dynamics and computational alternatives. To build the system, we offered a well-curated dataset suitable for ambient acoustic context analysis, by refining the AudioSet, a large-scale audio dataset through crowdsourcing. Then, we devised a lightweight purpose-built deep neural network solution that consists of an intelligently designed input segment for capturing audio data for ambient contexts and a neural network model that uses audio embedding generated by VGGish model [15]. Our experimental results shows that the proposed network can distinguish between 9 different ambient contexts with an average F_1 score of 0.80 and a computational latency of 3 milliseconds. We also presented a concrete manifestation of our solution in designing a context-augmented conversational agent with kinetic earables.

REFERENCES

[1] 2019. Dialogflow. https://dialogflow.com. Accessed: 2019-04-17.

[2] 2019. IBM Bluemix. https://console.bluemix.net. Accessed: 2019-04-17.

[3] Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a Human-like Open-Domain Chatbot. arXiv:2001.09977 [cs.CL]

[4] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *Proceedings of The 33rd International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 48)*, Maria Florina Balcan and Kilian Q. Weinberger (Eds.). PMLR, New York, New York, USA, 173–182.

[5] Christopher Baber. 2002. Developing interactive speech technology. *Interactive speech technology: Human factors issues in the application of speech input/output to computers* (2002), 1–18.

[6] Matthias Baldauf, Stefan Ribler, and Peter Fröhlich. 2019. Alexa, I'm in Need! Investigating the Potential and Barriers of Voice Assistance Services for Social Work. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services* (Taipei, Taiwan) *(MobileHCI 2019)*. Association for Computing Machinery, New York, NY, USA, Article 50, 6 pages. https://doi.org/10.1145/3338286.3344397

[7] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. In *Advances in neural information processing systems*. 577–585.

[8] Phil Cohen, Adam Cheyer, Eric Horvitz, Rana El Kaliouby, and Steve Whittaker. 2016. On the Future of Personal Assistants. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI EA 2016)*. Association for Computing Machinery, New York, NY, USA, 1032—-1037. https://doi.org/10.1145/2851581.2886425

[9] Gregory Gelly and Jean-Luc Gauvain. 2018. Optimization of RNN-Based Speech Activity Detection. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.* 26, 3 (March 2018), 646–656. https://doi.org/10.1109/TASLP.2017.2769220

[10] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio Set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*. New Orleans, LA.

[11] Petko Georgiev, Sourav Bhattacharya, Nicholas D. Lane, and Cecilia Mascolo. 2017. Low-resource Multi-task Audio Sensing for Mobile and Embedded Devices via Shared Deep Neural Network Representations. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 50 (Sept. 2017), 19 pages. https://doi.org/10.1145/3131895

[12] Alex Graves and Navdeep Jaitly. 2014. Towards end-to-end speech recognition with recurrent neural networks. In *International conference on machine learning*. 1764–1772.

[13] Kun Han, Dong Yu, and Ivan Tashev. 2014. Speech emotion recognition using deep neural network and extreme learning machine. (2014).

[14] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. 2014. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567* (2014).

[15] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron Weiss, and Kevin Wilson. 2017. CNN Architectures for Large-Scale Audio Classification. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. https://arxiv.org/abs/1609.09430

[16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the Knowledge in a Neural Network. *arXiv e-prints*, Article arXiv:1503.02531 (March 2015), arXiv:1503.02531 pages. arXiv:1503.02531 [stat.ML]

[17] Inseok Hwang, Youngki Lee, Chungkuk Yoo, Chulhong Min, Dongsun Yim, and John Kim. 2019. Towards Interpersonal Assistants: Next-Generation Conversational Agents. *IEEE Pervasive Computing* 18, 2 (2019), 21–31.

[18] Mohit Jain, Ramachandra Kota, Pratyush Kumar, and Shwetak N Patel. 2018. Convey: Exploring the use of a context view for chatbots. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–6.

[19] Fahim Kawsar, Chulhong Min, Akhil Mathur, and Alesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (2018), 83–89.

[20] Qiuqiang Kong, Yong Xu, Wenwu Wang, and Mark D Plumbley. 2018. Audio set classification with attention model: A probabilistic perspective. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 316–320.

[21] Michael J Kuhn. 2015. Virtual game assistant based on artificial intelligence. US Patent 9,202,171.

[22] Nicholas D Lane, Petko Georgiev, and Lorena Qendro. 2015. DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 283–294.

[23] Gierad Laput, Karan Ahuja, Mayank Goel, and Chris Harrison. 2018. Ubicoustics: Plug-and-Play Acoustic Activity Recognition. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. ACM, New York, NY, USA, 213–224. https://doi.org/10.1145/3242587.3242609

[24] Gierad Laput, Karan Ahuja, Mayank Goel, and Chris Harrison. 2018. Ubicoustics: Plug-and-Play Acoustic Activity Recognition. In *The 31st Annual ACM Symposium on User Interface Software and Technology - UIST '18*. ACM Press, Berlin, Germany, 213–224. https://doi.org/10.1145/3242587.3242609

[25] Dawei Liang and Edison Thomaz. 2019. Audio-Based Activities of Daily Living (ADL) Recognition with Large-Scale Acoustic Embeddings from Online Videos. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 1, Article 17 (March 2019), 18 pages. https://doi.org/10.1145/3314404

[26] Hong Lu, Wei Pan, Nicholas D Lane, Tanzeem Choudhury, and Andrew T Campbell. 2009. SoundSense: scalable sound sensing for people-centric applications on mobile phones. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*. ACM, 165–178.

[27] Paul Luff, David Frohlich, and Nigel G Gilbert. 2014. *Computers and conversation*. Elsevier.

[28] Ewa Luger and Abigail Sellen. 2016. "Like Having a Really Bad PA": The Gulf Between User Expectation and Experience of Conversational Agents. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. ACM, New York, NY, USA, 5286–5297. https://doi.org/10.1145/2858036.2858288

[29] Lindsay C Page and Hunter Gehlbach. [n.d.]. How an Artificially Intelligent Virtual Assistant Helps Students Navigate the Road to College. ([n. d.]), 12.

[30] Salvatore Parise, Patricia J Guinan, and Ron Kafka. 2016. Solving the crisis of immediacy: How digital technology can transform the customer experience. *Business Horizons* 59, 4 (2016), 411–420.

[31] Nishant Piyush, Tanupriya Choudhury, and Praveen Kumar. 2016. Conversational commerce a new era of e-business. In *2016 International Conference System Modeling & Advancement in Research Trends (SMART)*. IEEE, 322–327.

[32] Mirco Rossi, Sebastian Feese, Oliver Amft, Nils Braune, Sandro Martis, and Gerhard Tröster. 2013. AmbientSense: A real-time ambient sound recognition system for smartphones. In *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. IEEE, 230–235.

[33] Mirco Rossi, Gerhard Troster, and Oliver Amft. 2012. Recognizing daily life context using web-collected audio data. In *2012 16th International Symposium on Wearable Computers*. IEEE, 25–28.

[34] Tara Sainath and Carolina Parada. 2015. Convolutional Neural Networks for Small-Footprint Keyword Spotting. In *Interspeech*.

[35] Sajad Shahsavari, Hossein Sameti, and Hossein Hadian. 2017. Speech activity detection using deep neural networks. *2017 Iranian Conference on Electrical Engineering (ICEE)* (2017), 1564–1568.

[36] Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural Responding Machine for Short-Text Conversation. *arXiv:1503.02364 [cs]* (March 2015). http://arxiv.org/abs/1503.02364 arXiv: 1503.02364.

[37] Heung-Yeung Shum, Xiao-dong He, and Di Li. 2018. From Eliza to XiaoIce: challenges and opportunities with social chatbots. *Frontiers of Information Technology & Electronic Engineering* 19, 1 (2018), 10–26.

[38] Jaisie Sin and Cosmin Munteanu. 2019. An Information Behaviour-Based Approach to Virtual Doctor Design. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services* (Taipei, Taiwan) *(MobileHCI 2019)*. Association for Computing Machinery, New York, NY, USA, Article 44, 6 pages. https://doi.org/10.1145/3338286.3344391

[39] Myrthe L. Tielman, Mark A. Neerincx, Rafael Bidarra, Ben Kybartas, and Willem-Paul Brinkman. 2017. A Therapy System for Post-Traumatic Stress Disorder Using a Virtual Agent and Virtual Storytelling to Reconstruct Traumatic Memories. *Journal of Medical Systems* 41, 8 (Aug. 2017), 125. https://doi.org/10.1007/s10916-017-0771-y

[40] Xi Yang, Marco Aurisicchio, and Weston Baxter. 2019. Understanding Affective Experiences with Conversational Agents. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 542, 12 pages. https://doi.org/10.1145/3290605.3300772

[41] Xi Yang, Marco Aurisicchio, and Weston Baxter. 2019. Understanding Affective Experiences with Conversational Agents. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. ACM, New York, NY, USA, Article 542, 12 pages. https://doi.org/10.1145/3290605.3300772

[42] Changsong Yu, Karim Said Barsim, Qiuqiang Kong, and Bin Yang. 2018. Multi-level attention model for weakly supervised audio classification. In *DCASE 2018 Workshop*.

[43] Jennifer Zamora. 2017. Rise of the Chatbots: Finding A Place for Artificial Intelligence in India and US. In *Proceedings of the 22Nd International Conference on Intelligent User Interfaces Companion* (Limassol, Cyprus) *(IUI '17 Companion)*. ACM, New York, NY, USA, 109–112. https://doi.org/10.1145/3030024.3040201

[44] Yu Zhang, William Chan, and Navdeep Jaitly. 2017. Very deep convolutional networks for end-to-end speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4845–4849.

[45] Li Zhou, Jianfeng Gao, Di Li, and Heung-Yeung Shum. 2020. The design and implementation of xiaoice, an empathetic social chatbot. *Computational Linguistics* 46, 1 (2020), 53–93.