Black-Box Attacks via the Speech Interface Using Linguistically Crafted Input

Mary K. Bispham(✉), Alastair Janse van Rensburg, Ioannis Agrafiotis, and Michael Goldsmith

Department of Computer Science, University of Oxford, Oxford OX1 3QD, UK
{mary.bispham,alastair.rensburg,ioannis.agrafiotis, michael.goldsmith}@cs.ox.ac.uk

Abstract. This paper presents the results of experiments demonstrating novel black-box attacks via the speech interface. We demonstrate two types of attack that use linguistically crafted adversarial input to target vulnerabilities in the handling of speech input by a speech interface. The first attack demonstrates the use of nonsensical word sounds to gain covert access to voice-controlled systems. This attack exploits vulnerabilities at the speech recognition stage of handling of speech input. The second attack demonstrates the use of crafted utterances that are misinterpreted by a target system as a valid voice command. This attack exploits vulnerabilities at the natural language understanding stage of handling of speech input.

Keywords: Cyber security · Voice-controlled digital assistant · Human-computer interaction

1 Introduction

Speech interfaces as implemented in voice-controlled systems such as Google Assistant and Amazon Alexa represent a new type of attack surface that can be exploited by attackers seeking to gain unauthorised access to a system. Attacks via a speech interface that are not easily detectable by human listeners are particularly pernicious in their potential effects. Various attacks of this nature have been demonstrated in prior work. For example, Carlini et al. [1] have presented results showing it is possible to hide malicious commands to voice-controlled digital assistants in white noise, whereas Zhang et al. [2] have shown that it is possible to hide commands in frequencies that are above the human-audible range. In this paper, we present two new types of attacks via the speech interface that are not detectable by legitimate users of voice-controlled devices. The first

This work was supported by a doctoral training grant from the Engineering and Physical Sciences Research Council (EPSRC).

P. Mori et al. (Eds.): ICISSP 2019, CCIS 1221, pp. 93–120, 2020.
https://doi.org/10.1007/978-3-030-49443-8_5

of these attack types is an attack using nonsensical word sounds to exploit unintended functionality in speech recognition in a voice-controlled system. Specifically, our experimental work demonstrates an attack on speech recognition in Google Assistant using nonsensical word sounds to trigger a set of target commands. The second attack type is an attack targeting unintended functionality in natural language understanding in a voice-controlled system. Specifically, our experimental work demonstrates that it is possible to mislead natural language understanding functionality in Amazon Alexa Skills and in an open-source natural language understanding system to trigger a target action, using utterances that appear to human listeners to have a meaning that is unrelated to the target action. These adversarial utterances are crafted by embedding homophones of target command words in a different sense context.

This paper is an extended version of an earlier paper that presented the results of a pilot experiment and of a proof-of-concept study [3]. The pilot experiment presented in the earlier paper represented initial results on attacks on speech recognition in Google Assistant using nonsensical word sounds. The work presented in the current paper builds on the results of the pilot experiment by generating a new set of results on this type of attack using a refined methodology that achieves a higher attack success rate. The proof-of-concept study presented in the earlier paper represented feasibility tests that demonstrated the potential for attacking natural language understanding in Amazon Alexa Skills using adversarially crafted utterances. The work presented in the current paper builds on the results of the proof-of-concept study by generating a more substantial set of results on this type of attack. The proof-of-concept study presented in the earlier paper included both attacks based on word substitution, in which a word in a target command is replaced with an unrelated word, as well as attacks based on embedding alternate meanings of target command words in new utterances. As stated above, the work presented in the current paper focusses solely on the latter attack method, which we term a 'word transplant' attack. Word transplant represents a novel method for attacking natural language understanding that has to the best of our knowledge not been explored in prior work.

The remainder of the paper is structured as follows. Section 2 describes experimental work demonstrating the feasibility of attacks using nonsensical word sounds. Section 3 describes experimental work demonstrating the feasibility of attacks using unrelated utterances. Section 4 makes some suggestions for future work and concludes the paper.

2 Nonsense Attacks on Google Assistant

2.1 Description and Context

This section presents experimental work showing that it is possible to hide malicious voice commands to the voice-controlled digital assistant Google Assistant in word sounds that are perceived as meaningless by humans. We term this type of attack a 'nonsense' attack, in accordance with a taxonomy published in a previous paper [4], which categorises attacks via the speech interface according

to human perceptual categories. The attack can also be characterised as a black-box adversarial learning attack. The idea for this work was inspired by the use of nonsense words to teach phonics to primary school children.[1]

In prior work, Papernot et al. [5] have shown that a sentiment analysis method could be misled by input that was 'nonsensical' at the sentence level, i.e. the input consisted of a nonsensical concatenation of real words. By contrast, the work described here examines whether voice-controlled digital assistants can be misled by input that consists of nonsensical word sounds. Whilst the attack by Papernot et al. targeted a text-based natural language understanding functionality, the attack based on nonsensical word sounds presented here targets the automatic speech recognition component of a voice-controlled digital assistant. The attacks described here represent the first example of an attack of this type targeting speech recognition in a voice-controlled system.

Nonsensical word sounds as understood here are sounds that are composed of the sound units that are used in a given language, but to which no meaning is allocated within the current usage of that language. Sound units used to form words in a given language are known as 'phonemes'.[2] English has around 44 phonemes.[3] The line between phoneme combinations that carry meaning within a language and phoneme combinations that are meaningless is subject to change over time and place, as new words evolve and old words fall out of use (see Nowak and Krakauer [6]). The space of meaningful word sounds within a language at a given point in time is generally confirmed by the inclusion of words in an established reference work, such as, in the case of English, the Oxford English Dictionary.[4] Word sounds that are outside this space can be described as nonsense words. Nonsense words are a grey area between non-speech, i.e. noise, and meaningful speech.

The aim of the experimental work was to develop a novel attack based on nonsensical word sounds that have some phonetic similarity with the words of a relevant target command, using a systematic methodology. Specifically, we tested Google Assistant's response to English word sounds that were outside the space of meaningful word sounds in English, but that had a 'rhyming' relationship with meaningful words recognised as commands by Google Assistant. The term 'rhyme' is used to refer to a number of different sound relationships between words (see for example McCurdy et al. [7]), but it is most commonly used to refer to a correspondence of word endings.[5] For the purposes of the experimental work, rhyme was defined according to this commonly understood sense as words that share the same ending phoneme.

[1] See The Telegraph, 1st May 2014, "Infants taught to read 'nonsense words' in English lessons".

[2] See for example https://www.britannica.com/topic/phoneme.

[3] See for example https://www.dyslexia-reading-well.com/44-phonemes-in-english.html.

[4] See for example https://blog.oxforddictionaries.com/press-releases/new-words-added-oxforddictionaries-com-august-2014/.

[5] See https://en.oxforddictionaries.com/definition/rhyme.

The hypothesis behind the experimental work was that nonsensical word sounds represent a category of unexpected input for which current speech recognition systems lack an appropriate handling mechanism, and that this is in contrast to the processing of such input by humans, who perceive such input as having no meaning. Current speech recognition technologies are machine learning-based classifiers that use Hidden Markov Models (HMMs) to map acoustic features in a speech signal to a most likely sequence of words to have produced them (see for example McTear [8]). It was hypothesised that some sequences of nonsensical word sounds with sufficient similarity to a target command might be accepted as that target command at a confidence level higher or equal to the level required for recognition of speech input by the target system's speech recognition system as a legitimate command. It can be assumed that the confidence level required for recognition of speech input will have been set during training of a system such as Google Assistant to achieve optimal recall and precision measures on a test dataset. Setting a higher confidence threshold in order to prevent acceptance of nonsensical word sequences as legitimate commands might therefore lead to rejection by the system of legitimate input, implying an inevitable trade-off between usability and security. The attacks demonstrated in this experimental work thus exploit a vulnerability created by a focus on usability in the implementation of current systems. The attack concept is illustrated in Fig. 1, which shows the alignment of a dummy dataset of nonsense commands and legitimate commands to a higher and to a lower confidence threshold. The figure shows that as some of the nonsense commands are accepted by the system as valid commands with a higher level of confidence than some legitimate commands, it is not possible to prevent acceptance of all nonsense commands whilst ensuring acceptance of all legitimate commands. Implementing the higher

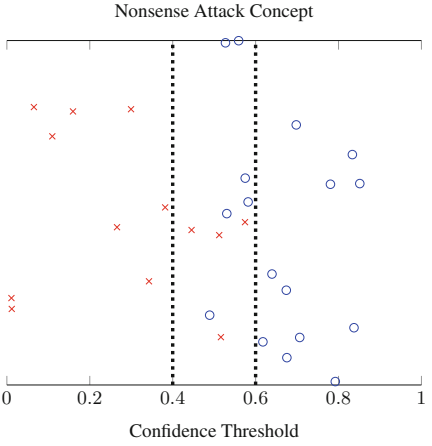

Fig. 1. x = nonsense commands; o = legitimate commands; dummy confidence threshold for ensuring acceptance of legitimate input = 0.4; dummy confidence threshold for ensuring rejection of nonsensical input = 0.6

confidence threshold will result in rejection of some legitimate commands, whereas implementing the lower threshold will result in acceptance of some nonsense commands.

The attacks presented here exploit three related features of speech recognition in voice-controlled systems. One of these features is the delineation of the space of word sounds that the Assistant has been trained to recognise as meaningful. The space of word sounds that a voice assistant such as Google Assistant can transcribe is much larger than the number of words that it can 'understand' in the sense of being able to map them to an executable command. In order to be able to perform tasks such as web searches by voice and note taking, a voice-controlled digital assistant must be able to transcribe all words in current usage within a language. It can therefore be assumed that the speech recognition functionality in Google Assistant is trained to recognise all words in current English usage. Whereas earlier speech recognition systems were vulnerable to potential confusion between out-of-vocabulary words that they did not have a capacity to recognise, on account of only a limited set of in-vocabulary words being included in their phonetic dictionary (see for example Hazen and Bazzi [9]), the potential for this type of confusion has been minimised in current systems. Earlier systems were also vulnerable to confusion between speech and non-speech sounds, but potential for this type of confusion has also been minimised in systems used for wake word detection that are trained using a noise model (see for example Raju et al. [10]). However, whilst the problem of delineating out-of-vocabulary words and non-speech sounds from in-scope words has been minimized in current systems, nonsensical word sounds still represent a type of out-of-scope input that speech recognition functionalities struggle to delineate from in-scope input. The inability of the Assistant to distinguish meaningful from meaningless word sounds is one of the features exploited in the attacks demonstrated here.

Another feature of speech recognition in voice assistants that is exploited in an attack using nonsense syllables is the influence of the language model used in speech recognition. Modern speech recognition technology combines acoustic modelling and language modelling components as parts of HMM-based speech recognition. The acoustic modelling component computes the likelihood of the acoustic features within a segment of speech having been produced by a given word. The language modelling component calculates the probability of one word following another word or words within an utterance. The acoustic model is typically based on Gaussian Mixture Models (GMMs) or deep neural networks (DNNs), whereas the language model is typically based on n-grams or recurrent neural networks (RNNs). Google's speech recognition technology as incorporated in Google Assistant is based on neural networks.[6] The words most likely to have produced a sequence of speech sounds are determined by calculation of the product of the acoustic model and the language model outputs. The language model is intended to complement the acoustic model, in the sense that

[6] See Google AI blog, 11th August 2015, 'The neural networks behind Google Voice transcription', https://ai.googleblog.com/2015/08/the-neural-networks-behind-google-voice.html.

it may correct 'errors' on the part of the acoustic model in matching a set of acoustic features to words that are not linguistically valid in the context of the preceding words. This assumption of complementary functionality is valid in a cooperative context, where a user interacts via a speech interface in meaningful language. However, the assumption of complementarity is not valid in an adversarial context, where an attacker is seeking to engineer a mismatch between a set of speech sounds as perceived by a human, such as the nonsensical speech sounds generated here, and their transcription by a speech-controlled device. In an adversarial context such as that investigated here, the language model may in fact operate in the attacker's favour, in that if one 'nonsense' word in an adversarial command is misrecognised as a target command word, subsequent words in the adversarial command will be more likely to be misrecognised as target command words in turn, as the language model trained to recognise legitimate commands will allocate a high probability to the target command words that follow the initial one.

A third feature of speech recognition in voice assistants exploited in covert attacks using this kind of input is the difference between machine and human processing of meaningless speech sounds. Like speech recognition by machines, speech recognition by humans is known also to reference an internal 'lexicon' to match speech sounds to words (see for example Roberts et al. [11]). However, unlike machines, humans also have an ability to categorise speech sounds as nonsensical. This discrepancy between machine and human processing of word sounds was the basis of the attack methodology for hiding malicious commands to voice assistants in nonsense words. Outside of the context of attacks via the speech interface, differences between human and machine abilities to recognise nonsense syllables have been studied for example by Lippmann et al. [12] and Scharenborg and Cooke [13]. Bailey and Hahn [14] examine the relationship between theoretical measures of phoneme similarity based on phonological features, such as might be used in automatic speech recognition, and empirically determined measures of phoneme confusability based on human perception tests. Machine speech recognition has reached parity with human abilities in terms of the ability correctly to transcribe meaningful speech (see Xiong et al. [16]), but not in terms of the ability to distinguish meaningful from meaningless sounds. The inability of machines to identify nonsense sounds as meaningless is exploited for security purposes by Meutzner et al. [15], who have developed a CAPTCHA based on the insertion of random nonsense sounds in audio. This experimental work explores the opposite scenario, i.e. the possible security problems associated with machine inability to distinguish sense from nonsense, and, conversely, human inability to recognise nonsensical input as meaningful.

2.2 Methodology

The experimental work comprised three key stages. The first stage involved generating, from a set of target commands, a set of potential adversarial commands consisting of nonsensical word sequences. These potential adversarial

commands were generated using a mangling process that involved replacing consonant phonemes in target command words to create a rhyming word sound, and then determining whether the resulting rhyming word sound was a meaningful word in English or a 'nonsense word', using the Unix word list as a proxy for the space of meaningful words in English. This was done so as to identify nonsensical word sounds that had an acoustic relationship to target command words and thus could be used to create potential adversarial commands. Word sounds identified as rhyming nonsense words were concatenated to create potential adversarial commands. Audio versions of these potential adversarial commands were created using speech synthesis technology. The second stage of the experimental work was to test the response of the target system to the potential adversarial commands, i.e. to test machine 'comprehension'. This was done both via audio file input and via over-the-air input of potential adversarial commands. The third stage of the experimental work was to test the human comprehensibility of adversarial commands that were successful in triggering a target action in the target system. The three key stages of the experimental work are shown in Fig. 2.

Fig. 2. Nonsense Attacks Experimental Stages.

The target system for the experiment was the voice-controlled digital assistant Google Assistant. The Google Assistant system was accessed via the Google Assistant Software Development Kit (SDK).[7] Target commands used in both experiments were selected to represent the generic types of action that can be performed by voice-controlled digital assistants. A voice-controlled digital assistant such as Google Assistant typically performs three generic types of action, namely information extraction, control of a cyber-physical action, and data input. The data input category may overlap with the control of cyber-physical action category where a particular device setting needs to be specified, eg. light colour or thermostat temperature. For this experiment, six target commands corresponding to the three types of action were used. The target commands were:

- "What's my name" (target action: retrieve username, action category: information extraction)
- "Who am I" (target action: retrieve username, action category: information extraction)
- "Turn on light" (target action: turn light on, action category: control of cyber-physical action)
- "Turn off light" (target action: turn light off, action category: control of cyber-physical action)
- "Turn light red" (target action: turn light to red, action category: data input)
- "Turn light blue" (target action: turn light to blue, action category: data input)

In addition to six specific target commands, a further command targeted in the experiments was the wake phrase "Hey Google" used to activate the Assistant.

[7] See https://developers.google.com/assistant/sdk/.

Adversarial Command Generation. Potential adversarial wake phrases and commands were created by replacing words in the original wake phrase or target command with a rhyming nonsense word. A set of rhyming nonsensical word sounds for each original word in the wake phrase and in each of the target commands was generated using a word mangling process. This mangling process was based on replacing consonant phonemes in the target command words to generate nonsensical word sounds that rhymed with the original target command word.[8] The target commands were first translated to a phonetic representation in the Kirshenbaum phonetic alphabet[9] using the 'espeak' functionality in Linux. The starting consonant phonemes of each word of the target command were then replaced with a different starting consonant phoneme, using a Python script and referring to a list of starting consonants and consonant blends.[10] Where the target command word began with a vowel phoneme, a starting consonant phoneme was prefixed to the vowel. The word sounds resulting from the word mangling process were checked for presence in a phonetic representation of the Unix word list, also generated with espeak, to ascertain whether the word sound represented a meaningful English word or not. If the sound did correspond to a meaningful word, it was discarded. This process generated for each target command word a number of rhyming nonsensical words to which no English meaning was attached. In the case of the wake phrase 'Hey Google', in addition to replacing the starting consonants 'H' and 'G', the second 'g' in 'Google' was also replaced with one of the consonants that are found in combination with the '-le' ending in English.[11]

For the audio file input tests, potential adversarial wake phrases and potential adversarial commands were generated separately. Original words in the wake phrase and target commands were replaced with one of the rhyming nonsense words for that word identified in the word-mangling process described above. Audio of the potential adversarial wake phrases and commands was created using Amazon Polly speech synthesis.[12] The potential adversarial wake phrases and commands generated for the audio file input tests included both potential adversarial wake phrases and commands in which all of the original words were replaced with nonsense words, as well as potential adversarial wake phrases and commands in which only some words were replaced. Specifically, the experiment included potential adversarial wake phrases and commands in which only one of the original words was replaced, potential adversarial commands in which only two of the three original words were replaced, and potential adversarial wake phrases and commands in which all of the original words were replaced. As the space of potential adversarial wake phrases and commands was quite large,

[8] Our approach was inspired by an educational game in which a set of nonsense words is generated by spinning lettered wooden cubes - see https://rainydaymum.co.uk/spin-a-word-real-vs-nonsense-words/.

[9] See http://espeak.sourceforge.net/phonemes.html.

[10] See https://k-3teacherresources.com/teaching-resource/printable-phonics-charts/.

[11] See https://howtospell.co.uk/.

[12] See https://aws.amazon.com/polly/.

a process of filtering and random sampling was used in generating potential adversarial wake phrases and commands in which more than one of the original words was replaced, as described in more detail below. Thus the potential adversarial commands generated for testing covered only a subspace of the full space of potential adversarial commands. The size of the full space of potential adversarial commands in which all of the original words are replaced is shown in Table 1.

Table 1. Space of potential adversarial commands for wake phrase and target commands.

Target Command	No. of Rhyming Nonsense Words	Space of Potential Adversarial Commands
Hey Google	'Hey': 17; 'Google': 395	6715
Who am I	'Who': 18; 'am': 27; 'I': 20	9720
What's my name	'What's': 27 ; 'my': 20 ; 'name': 35	18900
Turn on light	'turn': 40 ; 'on': 38 ; 'light': 28	42560
Turn off light	'turn': 40 ; 'off': 41 ; 'light': 28	45920
Turn light red	'turn': 40 ; 'light': 28 ; 'red': 25	28000
Turn light blue	'turn': 40 ; 'light': 28 ; 'blue': 18	20160

For the over-the-air and human comprehensibility tests, random samples of adversarial wake phrases and adversarial commands that had been successful in the audio file input tests were concatenated in different combinations to generate potential full adversarial commands, i.e. adversarial commands that would both activate the Assistant and trigger a specific target command. Potential full adversarial commands for each of the target commands were generated at different mangling levels. These levels were fully mangled commands, commands in which four of the original words had been mangled (two in the wake phrase and two in the specific target command, or one in the wake phrase and three in the specific target command), commands in which three of the original words had been mangled (one in the wake phrase and two in the specific target command, or two in the wake phrase and one in the specific target command), and commands in which two of the original words had been mangled (one in the wake phrase and one in the specific target command). Two potential adversarial commands were generated at each of the partial mangling levels, one in which the word 'Google' was one of mangled words, and one in which 'Google' was not mangled. This was in order to test the effect of the presence of the unmangled word 'Google' on machine and human comprehensibility of partially mangled adversarial commands.

Machine Comprehensibility Tests. The Google Assistant SDK was integrated in a Ubuntu virtual machine (version 18.04). The reason for accessing the

Google Assistant system via the Google Assistant SDK was that, unlike in the case of accessing Google Assistant using commercial devices such as the Google Home device, this allowed the Assistant's transcriptions of speech input to be retrieved. The transcriptions that could be retrieved using the Google Assistant SDK integrated in a virtual machine included both interim and final transcriptions of speech input to the Assistant. Two separate versions of Google Assistant SDK were integrated in the virtual machine; the Google Assistant Service, and the Google Assistant Library. The Google Assistant Service is activated via keyboard stroke and thus does not require a wake phrase, and voice commands can be inputted as audio files as well as over-the-air via a microphone. The Google Assistant Library, on the other hand, does require a wake phrase for activation, and receives commands via a microphone only. The Google Assistant Service could therefore be used to test adversarial commands for target commands and for the wake phrase separately via audio file input rather than via a microphone. The Google Assistant Library could be used to test the activation of the Assistant and the triggering of a target command by an adversarial wake phrase and adversarial command in combination over the air, representing a more realistic attack scenario. The Assistant's response to plain-speech versions of each target command was tested first to confirm that these target commands triggered the relevant target action.

Audio File Input Tests. For the audio file input tests, the target system's responses to audio versions of potential adversarial wake phrases and commands created using Amazon Polly were tested separately via command line input. The audio file input tests were performed at different levels of mangling using a filtering process for generating potential adversarial wake phrases and commands that built on successes found at a previous level of mangling. The testing process was automated using a Python script that first tested all possible potential adversarial wake phrases and commands in which only one of the original words had been mangled. Potential adversarial wake phrases and commands that were successful at this first level were then combined with one another to create a second level of potential adversarial wake phrases and commands in which two words had been mangled, with potential adversarial wake phrases and commands that were not successful at the first level being discarded. In the case of potential adversarial commands, a third level was also tested consisting of combinations of successful adversarial commands from the first level and successful adversarial commands at the second to generate fully mangled adversarial commands. At the mangling levels subsequent to the first level, the Python script tested up to a maximum of 150 potential adversarial commands at each level using random sampling, with a target of maximum 100 successes. This random sampling process was followed due to the large space of potential adversarial commands. A target action was considered to have been triggered if the Assistant's final transcription of adversarial input matched the target command.

Over-the-Air Tests. For the over-the-air tests, a random sample of adversarial wake phrases and adversarial commands that had been successful in audio

file input tests at different levels of mangling were concatenated to form full potential adversarial commands for five of the six target commands (the target command "turn light red" was not included due to a lack of successful adversarial commands being identified in the audio file input tests at higher levels of mangling). These potential full adversarial commands were tested via microphone input using Google Assistant Library. Table 2 shows the concatenations of randomly selected successful adversarial wake phrases and commands at different mangling levels for the target commands that were tested in over-the-air tests.

Table 2. Samples of successful adversarial wake phrases and commands concatenated for over-the-air and human comprehensibility tests.

Target command w. condition	fully mangled command	Level 4	Level 3	Level 2
Hey Google who am I (Google unmangled first)	Z'eI l'Uk@L spl'u: bl'am str'aI ("zhay lookle sploo blam strai")	v'eI g'u:g@L spl'u: bl'am str'aI ("vay Google sploo blam strai")	v'eI g'u:g@L v'u: T'am 'aI ("vay Google voo tham I")	v'eI g'u:g@L h'u: T'am 'aI ("vay Google who tham I")
as above (Google unmangled last)	as above	Z'eI l'Uk@L v'u: T'am 'aI ("zhay lookle voo tham I")	Z'eI l'Uk@L h'u: T'am 'aI ("zhay lookle who tham I")	h'eI g'Ud@L h'u: T'am 'aI ("Hey goodle who tham I")
Hey Google what's my name (Google unmangled first)	T'eI gl'u:s@L D'0ts sn'aI z'eIm ("thay gloosle thots snai zame")	Z'eI g'u:g@L D'0ts sn'aI z'eIm ("zhay Google thots snai zame")	Z'eI g'u:g@L w'0ts gr'aI Z'eIm ("zhay Google what's grai zhame")	Z'eI g'u:g@L w'0ts bl'aI n'eIm ("zhay Google what's blai name")
as above (Google unmangled last)	as above	h'eI w'u:b@L D'0ts sn'aI z'eIm ("Hey wooble thots snai zame")	h'eI w'u:b@L w'0ts gr'aI Z'eIm ("Hey wooble what's grai zhame")	h'eI w'u:b@L w'0ts bl'aI n'eIm ("Hey wooble what's blai name")
Hey Google turn on light (Google unmangled first)	Z'eI fl'Uk@L D'3:n f'0n D'aIt ("zhay flookle thurn fon thight")	Z'eI g'u:g@L D'3:n f'0n D'aIt ("zhay Google thurn fon thight")	Z'eI g'u:g@L t'3:n tr'0n p'aIt ("zhay Google turn tron pight")	Z'eI g'u:g@L br'3:n '0n l'aIt ("zhay Google brurn on light")
as above (Google unmangled last)	as above	h'eI k'u:s@L D'3:n f'0n D'aIt ("Hey koosle thurn fon thight")	Z'eI fl'Uk@L br'3:n '0n l'aIt ("zhay flookle brurn on light")	h'eI k'u:s@L br'3:n '0n l'aIt ("Hey koosle brurn on light")
Hey Google turn off light (Google unmangled first)	v'eI g'u:t@L g'3:n bl'0f j'aIt ("vay gootle gurn blof yight")	v'eI g'u:g@L g'3:n bl'0f j'aIt ("vay Google gurn blof yight")	v'eI g'u:g@L pr'3:n b'0f l'aIt ("vay Google prurn bof light")	v'eI g'u:g@L tr'3:n '0f l'aIt ("vay Google trurn off light")
as above (Google unmangled last)	as above	h'eI k'u:z@L g'3:n bl'0f j'aIt ("Hey koozle gurn blof yight")	v'eI g'u:t@L tr'3:n '0f l'aIt ("vay gootle trurn off light")	h'eI k'u:z@L tr'3:n '0f l'aIt ("Hey koozle trurn off light")
Hey Google turn light blue (Google unmangled first)	Z'eI gl'u:p@L pl'3:n g'aIt v'u: ("zhay gloople plurn gight voo")	T'eI g'u:g@L pl'3:n g'aIt v'u: ("thay Google plurn gight voo")	T'eI g'u:g@L fl'3:n v'aIt bl'u: ("thay Google flurn vight blue")	T'eI g'u:g@L t'3:n Z'aIt bl'u: ("thay Google turn zhight blue")
as above (Google unmangled last)	as above	Z'eI gl'u:p@L fl'3:n v'aIt bl'u: ("zhay gloople flurn vight blue")	h'eI bl'Uk@L fl'3:n v'aIt bl'u: ("Hey blookle flurn vight blue")	h'eI bl'Uk@L t'3:n Z'aIt bl'u: ("Hey blookle turn zhight blue")

Human Comprehensibility Tests. The human comprehensibility tests used the same concatenations of adversarial wake phrases and adversarial commands as were used in the over-the-air tests shown in Table 2. Participants in the human comprehensibility tests were presented with potential full adversarial commands in descending order of mangling on a spectrum from fully mangled adversarial commands to adversarial commands in which only one word in the wake phrase and one word in the target command had been mangled. This approach was taken so as to provide an indication of how many words would need to be mangled in an adversarial over-the-air command in order to escape human comprehensibility. Subjects were asked to indicate whether they had identified any meaning in the audio. If they had identified meaning, they were asked to indicate what meaning they heard. After hearing all of the potential adversarial commands, participants were also presented with a plain-speech version of the full target command, which provided a baseline for the comprehensibility tests, and also served as an attention test.

The potential full adversarial commands were separated into two sets for each target command. In the first set, "Google" was the first word to be revealed to the listener in plain-speech, whereas in the second set, "Google" was the final word to be revealed. The separation of these two conditions enabled an assessment of whether the presence of the specific word "Google" affected listeners' ability to detect the presence of a voice command, and to realise its possible content. Each set of five wake phrase and command combinations for each target command under each of the two conditions was played to six different participants.

Participants were recruited using the survey website Prolific Academic.[13] The experiments with human subjects received ethics clearance through the Departmental Research Ethics Committee of the Department of Computer Science at the University of Oxford. All subjects were native speakers of English.

2.3 Results

Machine Comprehensibility Tests

Audio File Input Tests. Table 3 shows the overall ratio of successes to failures in the audio file input tests, as well as the number of successes for adversarial wake phrases and commands at each level of mangling. The differences between success rates of potential adversarial wake phrases and commands at different levels of mangling are shown to be not very significant. This suggests that the approach of limiting the pool of potential adversarial wake phrases and commands tested at each mangling level to combinations of adversarial wake phrases commands successful at the previous level is effective in maximising the success rates of attacks at each level. With the exception of the "turn light red" target command, successful adversarial commands could be generated for all target commands at all mangling levels, as was also the case for the target wake phrase. Overall success rates for target commands apart from the "turn light red" command

[13] See https://prolific.ac/.

ranged from 29.9% to 63.8%. The "turn light red" target command appeared to be an outlier in terms of success rates for potential adversarial commands, with a success rate of only 3.2%; no clear reason for this was apparent. The overall success rate for the adversarial wake phrase was 14.4%.

Figure 3 shows an example of the output to the command line produced by a successful fully mangled wake phrase and two successful fully mangled adversarial commands, showing both interim and final transcriptions of the adversarial input by the Assistant.

Table 3. Success rates of adversarial commands in audio file input tests.

Target command	Overall success rate	Level 1 successes	Level 2 successes	Level 3 successes
Hey Google	14.4%	52	18	n.a.
Who am I	29.9%	46	21	18
What's my name	55.4%	56	52	57
Turn on light	49.2%	44	46	65
Turn off light	56.7%	52	50	83
Turn light red	3.2%	3	none	none
Turn light blue	63.8%	41	62	63

```
ADVERSARIAL WAKE PHRASE FOR "Hey Google": v'eI g'u:t@L ("vay gootle")

WARNING:root:Transcript of user request: "V".
WARNING:root:Transcript of user request: "wake".
WARNING:root:Transcript of user request: "Virgo".
WARNING:root:Transcript of user request: "very good".
WARNING:root:Transcript of user request: "viagogo".
WARNING:root:Transcript of user request: "hey Google".
WARNING:root:Transcript of user request: "hey Google".
WARNING:root:Transcript of user request: "hey Google".
WARNING:root:Playing assistant response.
WARNING:root:Expecting follow-on query from user.
WARNING:root:Finished playing assistant response.
RESPONSE TRANSCRIPTION: hi what can I do for you

ADVERSARIAL COMMAND FOR "who am I": f'u: D'am z'aI ("foo tham zai")

WARNING:root:Transcript of user request: "true".
WARNING:root:Transcript of user request: "through the".
WARNING:root:Transcript of user request: "who am".
WARNING:root:Transcript of user request: "fu Fareham".
WARNING:root:Transcript of user request: "who am I".
WARNING:root:Transcript of user request: "who am I".
WARNING:root:Transcript of user request: "who am I".
WARNING:root:Playing assistant response.
WARNING:root:Finished playing assistant response.

ADVERSARIAL COMMAND FOR "turn off light": n'3:n T'0f j'aIt ("nurn thoff yight")

WARNING:root:Transcript of user request: "no".
WARNING:root:Transcript of user request: "9".
WARNING:root:Transcript of user request: "turn off".
WARNING:root:Transcript of user request: "turn off the".
WARNING:root:Transcript of user request: "turn off  the".
WARNING:root:Transcript of user request: "turn off  my".
WARNING:root:Transcript of user request: "turn off  light".
WARNING:root:Transcript of user request: "turn off light".
WARNING:root:Transcript of user request: "turn off light".
WARNING:root:Playing assistant response.
WARNING:root:Finished playing assistant response.
```

Fig. 3. Audio File Input Tests - Successes.

Figure 4 shows some examples of the output to the command line produced by an unsuccessful fully mangled wake phrase and two unsuccessful fully mangled adversarial commands, showing both interim and final transcriptions of the adversarial input by the Assistant. The unsuccessful examples share one nonsensical word sound with the corresponding successful example in Fig. 3, demonstrating that the success or failure of adversarial wake phrases and target commands in triggering a target action was influenced not only by the probabilities allocated to individual word sounds by the acoustic model used in the Assistant's speech recognition, but also by the probabilities allocated to utterances as a whole by the Assistant's language model.

Over-the-Air Tests. Table 4 shows the results of tests of the Assistant's response to input via microphone of audio versions of the partially mangled and fully mangled adversarial commands listed in Table 2. Specifically, the complete target action was activated by the adversarial commands for the 'what's my name' target command at the fourth, third and second levels of mangling under the condition of the word Google being revealed last, and by the adversarial command for 'turn on light' at the second level of mangling under the condition of the word Google being revealed last. There were also instances where although the target command itself was not triggered, the adversarial command did activate the Assistant by triggering the wake phrase.

Human Comprehensibility Tests. Table 5 shows the results of tests of human comprehensibility of audio versions of the partially mangled and fully mangled full adversarial commands listed in Table 2. The results are summarised according to whether a simple majority of participants identified no meaning, part of the target command meaning, or the full target command meaning in the adversarial audio input. Where different results are identified by an equal number of participants, this is indicated in the table. There were four instances where participants returned a blank test result. In these cases, results are given out of five participants instead of six, as detailed in the table.

A consistent result across all the tests was that, with one sole exception, none of the participants identified any meaning in the fully mangled adversarial wake phrase and target command combinations. Otherwise the results from this small-scale test represent a more mixed picture. Some participants did not hear any meaning in the audio clips prior to hearing the plain-speech command, whereas others picked up some of the adversarial wake phrase and target command words at the lower levels of mangling prior to hearing the plain-speech command. Some participants identified words in adversarial commands that were not actually present in the wake phrase or target command. A few participants believed that they had heard a different language, or tried to transcribe some of the nonsensical word sounds. A couple of participants identified the entire meaning of a target command prior to hearing the plain-speech version in some instances. The condition as to whether the word 'Google' was revealed first or last did not appear to significantly affect the participants' ability to detect the content of the

```
 ADVERSARIAL WAKE PHRASE FOR "Hey Google": v'eI gl'u:f@L ("vay gloofle")

WARNING:root:Transcript of user request: "V".
WARNING:root:Transcript of user request: "wake".
WARNING:root:Transcript of user request: "vehicle".
WARNING:root:Transcript of user request: "fake love".
WARNING:root:Transcript of user request: "The Gruffalo".
WARNING:root:Transcript of user request: "The Gruffalo".
WARNING:root:Transcript of user request: "The Gruffalo".
WARNING:root:Playing assistant response.
WARNING:root:Finished playing assistant response.

ADVERSARIAL COMMAND FOR "who am I": spl'u: bl'am z'aI ("sploo blam zai")

WARNING:root:Transcript of user request: "screw".
WARNING:root:Transcript of user request: "play".
WARNING:root:Transcript of user request: "volume".
WARNING:root:Transcript of user request: "who do I am sorry".
WARNING:root:Transcript of user request: "who do I am sorry".
WARNING:root:Transcript of user request: "volume three".
WARNING:root:Finished playing assistant response.


ADVERSARIAL COMMAND FOR "turn off light": n'3:n v'0f tS'aIt ("nurn voff chight")

WARNING:root:Transcript of user request: "no".
WARNING:root:Transcript of user request: "Night by".
WARNING:root:Transcript of user request: "new bar".
WARNING:root:Transcript of user request: "new bath".
WARNING:root:Transcript of user request: "buy a".
WARNING:root:Transcript of user request: "bye bye".
WARNING:root:Transcript of user request: "turn both tried".
WARNING:root:Transcript of user request: "9 Bath Street".
WARNING:root:Playing assistant response.
WARNING:root:Finished playing assistant response.
```

Fig. 4. Audio File Input Tests - Losses.

Table 4. Results of Over-the-Air Tests.

Condition	fully mangled command	Level 4	Level 3	Level 2	Target command
Google unmangled first	wake phrase activated	unsuccessful	unsuccessful	unsuccessful	Hey Google who am I
Google unmangled last	as above	wake phrase activated	unsuccessful	unsuccessful	as above
Google unmangled first	unsuccessful	unsuccessful	unsuccessful	unsuccessful	Hey Google what's my name
Google unmangled last	as above	successful	successful	successful	as above
Google unmangled first	unsuccessful	unsuccessful	unsuccessful	unsuccessful	Hey Google turn on light
Google unmangled last	as above	unsuccessful	unsuccessful	successful	as above
Google unmangled first	unsuccessful	unsuccessful	unsuccessful	unsuccessful	Hey Google turn off light
Google unmangled last	as above	unsuccessful	unsuccessful	unsuccessful	as above
Google unmangled first	unsuccessful	unsuccessful	unsuccessful	unsuccessful	Hey Google turn light blue
Google unmangled last	as above	unsuccessful	unsuccessful	unsuccessful	as above

entire command. A notable result was that the adversarial commands for the 'what's my name' target command at the fourth and third levels of mangling under the condition of the word 'Google' being revealed last that had been effective in triggering the target action in the over-the-air tests were identified as also being successful in evading human comprehensibility. Thus these two partially mangled adversarial commands represent fully effective covert attacks on the target system.

As regards transcription of the plain-speech target commands, these were transcribed correctly by a large majority of participants. There were three instances where transcription of the plain-speech command was incomplete, one where it was incorrect, and one where transcription of the plain-speech was missing.

Table 5. Results of Human Comprehensibility Tests.

Condition	fully mangled command	Level 4	Level 3	Level 2	Target command
Google unmangled first	no meaning (5/6 participants)	no meaning (5/6 participants)	no meaning/partial meaning (3/6 participants)	partial meaning (4/6 participants)	Hey Google who am I
Google unmangled last	as above	no meaning (4/6 participants)	no meaning (4/6 participants)	partial meaning (4/6 participants)	as above
Google unmangled first	no meaning (6/6 participants)	no meaning (4/6 participants)	partial meaning (4/6 participants)	partial meaning (5/6 participants)	Hey Google what's my name
Google unmangled last	as above	partial meaning (4/6 participants)	no meaning/partial meaning (3/6 participants)	partial/full meaning (2/5)	as above
Google unmangled first	no meaning (6/6 participants)	no meaning (5/6 participants)	no meaning/partial meaning (3/6 participants)	partial meaning (5/6)	Hey Google turn on light
Google unmangled last	as above	no meaning (4/6 participants)	partial meaning (4/6 participants)	partial meaning (4/6 participants)	as above
Google unmangled first	no meaning (6/6 participants)	no meaning (4/6 participants)	no meaning/partial meaning (3/6 participants)	partial meaning (5/6 participants)	Hey Google turn off light
Google unmangled last	as above	no meaning (5/6 participants)	no meaning/partial meaning (3/6 participants)	partial meaning (5/6 participants)	as above
Google unmangled first	no meaning (6/6 participants)	no meaning (5/5 participants)	no meaning (3/5 participants)	partial meaning (6/6 participants)	Hey Google turn light blue
Google unmangled last	as above	no meaning (5/6 participants)	partial meaning (5/6 participants)	partial meaning (5/6 participants)	as above

2.4 Discussion

The combined results from the machine response and human comprehensibility tests confirm the hypothesis that voice-controlled digital assistants are potentially vulnerable to covert attacks using nonsensical word sounds. The key finding is that voice commands to voice-controlled digital assistant Google Assistant can be triggered by nonsensical word sounds in some instances, whereby the same nonsensical word sounds are perceived by humans as either not having any meaning at all, or as having a meaning only partially related to the voice commands to the Assistant. This supports the hypothesis that adversarial input consisting of nonsensical word sounds having an acoustic relationship with target command words may be recognised as legitimate commands at a confidence level equal to or higher than that required for speech recognition by the Google Assistant target system as trained for optimal performance in recognition of legitimate commands. The findings further show that it is not always necessary to replace all of the original words in a target command in order to generate an adversarial command that is successful in triggering a target action in a target system. Particularly with regard to over-the-air attacks, replacing only some rather than all of the target command words with nonsense words may increase the success rate of adversarial commands, whilst still preserving the covert nature of the attacks in terms of being hidden from human understanding. This is based on the finding that partially mangled adversarial commands were successful both in triggering a target action over-the-air and in hiding from human recognition in some instances.

The results confirm the influence of the three features of speech recognition in current voice-controlled systems in enabling this type of attack via the speech interface, as discussed above. These three features were thus shown to represent security vulnerabilities in the current generation of voice-controlled digital assistants.

The first of these features was the target system's inability to recognise the true nature of nonsensical word sounds. As envisaged, the attacks demonstrated in this experimental work exploit a vulnerability in the speech recognition functionality of the Google Assistant target system of being unable to recognise nonsensical word sounds as meaningless. In the results of the experimental work, the Google Assistant target system always either indicated incomprehension or attempted to match the nonsensical sounds to real words, rather than transcribing the nonsense word sound. This confirms that the Assistant is vulnerable to being fooled by word sounds that are perceived by humans as obviously nonsensical. The findings are in accord with the hypothesis behind these experiments that as a grey area between speech and non-speech, nonsensical word sounds represent a part of the input space to a voice-controlled system that the current generation of voice-controlled digital assistants struggles to handle appropriately. Whilst the Assistant does reject some of the input from this grey area as incomprehensible, in other instances input from this grey area is treated as meaningful input.

The second of these features was the influence of the language model in enabling the success of some of the attacks. The examples found in the experiment of the same nonsensical word sounds being present in both successful and unsuccessful adversarial wake phrases and commands confirms that the triggering of a target action by adversarial input may be influenced by probabilities allocated by the language model used in speech recognition to an utterance as a whole, as well as by probabilities allocated to individual word sounds by the acoustic model. Thus the aim of language modelling of 'correcting' possible incorrect word recognitions may have the opposite effect in an adversarial context of enabling the success of attacks based on nonsensical word sounds in some instances.

The third feature shown to be exploited in the attacks was discrepancy in human and machine processing of nonsensical input. The machine and human responses to nonsensical word sounds in general were comparable, in that both machine and humans frequently indicated incomprehension of the sounds, or else attempted to fit them to meaningful words. However, in the specific instances of nonsensical word sound sequences that triggered a target command in Google Assistant, human listeners did not hear a Google Assistant voice command in the nonsensical word sounds that had triggered a target command in the majority of instances. In addition to either indicating incomprehension or transcribing the nonsensical sounds as real words, human subjects on occasion attempted to transcribe the nonsensical word sounds phonetically as nonsense syllables. This superior ability of humans to recognise nonsensical word sounds as meaningless paradoxically prevented human listeners from detecting the presence of a malicious voice command, thus enabling the covert attacks.

A notable feature of the results of human comprehensibility tests is their variability between individual experimental subjects. Thus the covert nature of these attacks depends to some extent on individual human perception, i.e. whereas some individuals may hear target command words in an adversarial command based on nonsensical word sounds, others may not. This was seen in the variable results of the human comprehensibility tests described above. Human perception of word sounds is known to be unstable in some instances, seen for example in a widely shared audio recording in which some listeners heard the word "Yanny" whereas others heard the word "Laurel".[14]

3 Missense Attacks on Amazon Alexa and RASA NLU

3.1 Description and Context

This section presents experimental work demonstrating that it is possible to gain unauthorised access to a voice-controlled system using utterances that are

[14] See for example The Guardian, "Laurel or Yanny explained: why do some people hear a different word?", 17th May 2018, https://www.theguardian.com/technology/2018/may/16/yanny-or-laurel-sound-illusion-sets-off-ear-splitting-arguments.

accepted by the system as a target command despite having a different meaning to the target command in terms of human understanding. We term this type of attack a 'missense' attack, in accordance with a taxonomy of attacks via the speech interface published in a previous paper [4], which categorises attacks via the speech interface according to human perceptual categories. The attack can also be characterised as a black-box adversarial learning attack. The aim of the attacks of generating adversarial utterances that trigger a target command but that are unrecognisable as such to humans was realised by embedding alternate meanings of target command words in an unrelated utterance to create an adversarial utterance. As stated above, these attacks are termed 'word transplant' attacks.

In prior work, Carlini and Wagner [17] have used crafted audio recordings of speech that is unrelated to voice commands to attack a speech transcription system. Whereas the attacks by Carlini and Wagner target speech recognition functionality, the attacks presented here target natural language understanding. There have been no comparable attacks targeting natural language understanding in voice-controlled systems reported in prior work. There have been some examples of attacks on natural language understanding in related areas, such as sentiment analysis (see for example Kuleshov et al. [18]). However, these attacks have used different attack methods based on word substitution. Word transplant attacks have not been demonstrated in any prior work, and thus represent a novel attack concept.

Linguistically plausible adversarial examples that trigger an action in a voice-controlled system with an utterance of apparently unrelated meaning are difficult to generate using automated, mathematical approaches. As noted by Papernot et al. [5], adversarial learning in the context of natural language understanding technologies that take as input a sequence of words is not a differentiable problem. Papernot et al. concede that their own work on fooling a sentiment classifier with 'nonsensical' sentences generated using a mathematical method has some limitations, in that the nonsensical nature of the adversarial sentences is easily noticeable by humans. They point to the need in future work to address grammar and semantics in adversarial sentence generation, in order to make sentences indistinguishable from innocent utterances by humans. The attacks demonstrated here attempt to do this using a manual, non-mathematical approach for generating adversarial voice commands by manipulating linguistic parameters such as syntactic structures and word meanings, rather than mathematical parameters such as acoustic features or word embedding vector values.

Natural language understanding in voice-controlled systems involves a process of semantic parsing for mapping transcriptions of spoken utterances to a formal representation of the utterances' meaning that the system can use to trigger an action. This usually involves some form of machine learning such as Conditional Random Fields (CRFs) or RNNs (see for example Mesnil et al. [19]). The process of semantic parsing may take into account the syntactical structure of an utterance as well as the presence of individual words to determine the most appropriate action to take in response to a natural language command (see for

example McTear [8]). The state-of-the-art in machine natural language understanding is known to fall far short of human abilities (see for example Cambria and White [20]).

The hypothesis behind the attacks presented here is that the deficiencies of natural language functionality in the current generation of voice-controlled digital assistants may render such systems vulnerable to being misled by adversarially crafted input that triggers a target action in the system, whilst being perceived by humans as unrelated to that target action. Specifically, it is hypothesised that word transplant attacks will exploit deficiencies in out-of-domain detection, that is the ability to reliably distinguish between relevant and irrelevant speech input (see for example Tür et al. [21]), as well as deficiencies in word-sense disambiguation, that is the ability to reliably determine the correct meaning of a word in context (see for example Stolk et al. [22]). Current systems identify speech input as in-domain or out-of-domain based the presence or absence of a combination of linguistic features, with word sense disambiguation being performed as part of this process based on co-occurrence of words in a given context. Such methods may be misled by crafted adversarial commands that retain some elements of a target command, as is the case in the word transplant attacks demonstrated here that reuse content words from a target command in a different sense context. Crafted adversarial input of this type is likely to thwart the system's ability to understand the intent of an utterance based on combinations of features, and to determine the intended meaning of individual words based on the context of neighbouring words. Given the crudity of current methods in natural language understanding for distinguishing valid from invalid input, as in the case of the attacks on speech recognition using nonsensical word sounds described above, applying higher confidence levels for the determination of user intent in natural language understanding to thwart such attacks may result in non-acceptance by the system of legitimate input and thus damage usability of the system.

The deficiencies in the current state-of-the-art in natural language understanding in distinguishing relevant from irrelevant input necessitate an assumption in the design principles for systems such as voice-controlled digital assistants of a genuine intent between user and device to communicate as conversation partners. In other words, such systems have no choice but to assume that any speaker interacting with them intends to communicate a relevant meaning. The guidelines for developing Google Conversation Actions, for example, recommend applying a set of conversation rules known as 'Grice's Maxims', the first of which is "only say things which are true".[15] In an adversarial setting, the assumption of shared context does not hold, and thus puts the system at risk of being misled by malicious input in missense attacks.

The covert nature of the attacks depends on unrelated utterances being used for adversarial purposes not being detected as a trigger for a voice-controlled action by human listeners. It is in fact unlikely that human listeners will detect unrelated utterances as covert voice commands, as humans are for the most part

[15] See https://developers.google.com/actions/downloads/be-cooperative.pdf.

so proficient at the language comprehension task that a large part of human natural language interpretation is performed automatically without conscious consideration. Miller [23] states that the alternative meanings of a word of which the meaning in context is clear will not even occur to a human listener, claiming the humans hearing the sentence "He nailed the board across the window", for example, will not even notice that "board" has more than one meaning: "Only one sense of "board" (or of "nail") reaches conscious awareness." This suggests that the very proficiency of humans in natural language understanding may hinder victims in identifying attacks that seek to exploit the limitations of automated systems in performing the same task.

The attacks described here were demonstrated on two specific natural language understanding functionalities. The first of these was the natural language understanding functionality behind Amazon Alexa Skills. Skills are third-party applications that can be incorporated in the Alexa digital assistant. Developers of Amazon Alexa Skills can make use of generic templates for actions to be performed by the Skill that are made available in the Amazon Developer Console, the so-called Built-in Intents, and/or create their own Custom Intents using the tools provided in the developer environment (see Kumar et al. [24]). Alexa Skills share speech recognition and natural language understanding functionalities with the core Alexa digital assistant. The natural language understanding functionality in Amazon Alexa uses as a meaning representation structure the so-called Alexa Meaning Representation Language (AMRL), which consists of graph-based structures representing the actions that can be performed by Alexa on different types of entities (see Kollar et al. [25]). Built-In Intents for Alexa Skills are based on pre-existing AMRL structures. Custom Intents in Alexa Skills do not make use of pre-existing AMRL structures as such, however, they do make use of the same natural language understanding models for mapping natural language utterances to meaning representation made available in the developer environment for Alexa Skills, as explained by Kumar et al. As stated by Kumar et al., Alexa's natural language understanding functionality will generate a semantic representation of the Custom Intent based on the sample utterances provided by the user. Various models are used to map natural language utterances to meaning representation in Amazon Alexa Skills, including CRFs and neural networks (see Kumar et al.). Kumar et al. explain that the process of mapping natural language utterances to the semantic representation of an intent, i.e. semantic parsing, has both a deterministic and stochastic element. The deterministic element ensures that all of the sample utterances provided by the user will be reliably mapped to the intent, whereas the stochastic element ensures some flexibility in the parsing of previously unheard utterances.

The second target system used in the experiment was an open source natural language understanding functionality named RASA NLU. RASA NLU is a natural language understanding library made available for use by non-specialist developers.[16] The RASA NLU target system was implemented using the 'spacy sklearn' pipeline option, which incorporates pre-existing generic word embed-

[16] https://rasa.com/docs/nlu/.

dings, which are used in combination with training data provided by the user to train a classifier to recognise the intents specified by the developer (as detailed by Bocklisch et al. [26]). This enables users to create bots using a relatively small amount of training data.

The specific setting of the envisaged attacks is a voice assistant used for personal banking. The use of digital assistants in financial services is becoming more common, with some suggestion that such systems are seen as providing better customer service than human agents (as reported by Qi and Xiao [27]). In his book entitled 'Bank 4.0', Brett King claims that voice assistants will assume great signficance in banking and financial advice services in future development of the industry [28].

3.2 Experiment

Methodology. Two target systems were created for the purposes of the experiment. These were an Alexa Skill and a bot based on RASA NLU. Both systems were dummy banking assistants that mimic the capabilities of a real Alexa Skill made available by Capital One bank to its customers.[17] The Capital One Skill enables three types of intents that can be expressed by their customers via voice command, namely Check Your Balance, Track Your Spending, and Pay Your Bill. The dummy assistants created for the purposes of the experiment included mock versions of these three intents, as well as mock versions of two further intents, namely to reset a password that a user had forgotten, and to block a credit card that had been lost or stolen.The dummy Alexa Skill also implemented the pre-built FallBackIntent available in the Amazon Developer Console, which represents a confidence threshold for acceptance of valid input by the Skill. Without implementation of a confidence threshold via the FallbackIntent, an Alexa Skill will treat any utterance as relevant and match the utterance to one of its actions. The RASA NLU target system implemented the same five target intents as the dummy Alexa Skill, and also implemented five generic intents, namely a greeting intent, a thanks intent, a goodbye intent, an affirmation intent, and an intent to provide a name. The generic intents were implemented to improve robustness of the RASA NLU target system. The RASA NLU system further implemented a 'nonsense' intent that was intended to be representative of out-of-scope input, performing a similar function to the FallBackIntent in the Amazon Alexa Skill.

Training data for the five target intents was the same for both the dummy Alexa Skill and for the RASA NLU bot. The training utterances represented a combination of example commands publicised by Capital One for their real Alexa Skill, publicly available training data examples for a third-party banking assistant bot[18], and self-generated training data. The training datasets contained 30 utterances for the account balance, recent transactions and pay bill intents, and 15 utterances for the reset password and block card intents. The five generic

[17] https://www.capitalone.com/applications/alexa/.

[18] This was a template for a banking assistant bot made available by IBM at https://github.com/IBM/watson-banking-chatbot.

intents in the RASA NLU target system were trained with sample utterances made available to developers by RASA NLU. The nonsense intent in the RASA NLU target system was trained with a large set of unrelated utterances made available by a third-party developer of another banking bot.[19]

Table 6. Target systems' response to target commands.

Test/Target Intent	Test/Target Command	RASA NLU Test Result	Alexa Skill Test Result
get account balance	tell me the current balance	target intent triggered	target intent triggered
get recent transactions	show me all my transactions	target intent triggered	target intent triggered
pay bill	pay a bill for electricity	target intent triggered	target intent triggered
reset password	can't recall my password	target intent triggered	target intent triggered
block card	think my card is stolen	target intent triggered	target intent triggered

Table 7. Target systems' response to out-of-scope commands.

Control Intent	Control Command	RASA NLU Test Result	Alexa Skill Test Result
be back	I'll get back to you in a moment	nonsense intent triggered	FallBackIntent triggered
be back	be back in 5 min	nonsense intent triggered	FallBackIntent triggered
be back	I'll be back	nonsense intent triggered	FallBackIntent triggered
be back	I promise to come back	nonsense intent triggered	FallBackIntent triggered
be back	I'll be back in a few minutes	nonsense intent triggered	FallBackIntent triggered

After training of the target systems, the systems' responses to utterances not seen in training were tested with respect to both in-scope and out-of-scope utterances. Input to the target systems was text-based. A test utterance for each of the specific intents for triggering the five target actions was inputted. The test utterances were utterances that had not been used in training, but that were clearly within the scope of the given intent. In order to test the systems' ability to reject non-malicious out-of-scope input, the tests also assessed the systems' responses to five other utterances that were unrelated to any of the actions within the scope of the Alexa Skill target system or the RASA NLU target system (these were five training utterances for a 'be back' intent that was part of the sample training data made available to developers by RASA NLU). Details of the tests of the systems' responses to in-scope and non-malicious out-of-scope input are shown in Tables 6 and 7 respectively. The tests confirmed that the target systems were robust in their handling of in-scope input not seen in training and non-malicious out-of-scope input, with all of the test utterances triggering the appropriate intent in both systems, and all of the control utterances triggering the nonsense intent in the RASA NLU target system and the FallBackIntent in the Alexa Skill. The test utterances were thus used as target commands for the missense attacks. Testing of the dummy Alexa Skill was performed in a sandbox environment in the Amazon Developer Console only and was not deployed in the Alexa cloud. Testing of the RASA NLU system was performed locally via a terminal.

[19] https://github.com/Twanawebtech/bank-chatbot.

Potential adversarial utterances were generated using the following process. First, a list of content words from the sample utterances for each Custom Intent was extracted (content words are words that give meaning to a sentence or utterance, as distinguished from function words that contribute to the syntactical structure of the sentence or utterance rather to its meaning, examples being prepositions such as 'of', determiners such as 'the', pronouns such as 'he' etc.). This was done automatically using a Python script implementing the Natural Language Toolkit (NLTK).[20] Second, a dictionary API[21] was used to automatically retrieve different word meanings and usage examples for the content words in the target commands. This enabled the identification and use of unusual and outdated word meanings for the target command words, which might be expected to increase the probabilities of successfully misleading natural language understanding systems such as that implemented in an Alexa Skill or RASA NLU bot, which are likely to have been trained to handle only common and current meanings of words. Following the extraction of content words and alternate word meanings, potential adversarial utterances for each Custom Intent were then generated manually, by embedding alternate meanings of words from the target command in new utterances, using as few new content words as possible, to create a potential adversarial command with a different meaning to the target command. The response of both target systems to each potential adversarial utterance was then tested.

Results. Table 8 shows the results of the word transplant attacks. The Amazon Alexa Skill target system was seen to be more vulnerable that the RASA NLU system. All but one of the word transplant attacks on the Alexa Skill target system were successful. On the RASA NLU target system, word transplant attacks were successful in only two out of five instances.

Table 8. Target systems' response adversarial commands generated by word transplant.

Target Intent	Target Command	Adversarial Command (Word Transplant)	RASA NLU Test Result	Alexa Skill Test Result	original content words / total content words
get account balance	tell me the current balance	I kept my balance in the current	target intent triggered	target intent triggered	2 out of 3
get recent transactions	show me all my transactions	the transactions were for show	nonsense intent triggered	target intent triggered	2 out of 2
pay bill	pay a bill for electricity	bill of an anchor	nonsense intent triggered	target intent triggered	1 out of 2
reset password	can't recall my password	we can't recall our product	nonsense intent triggered	FallBackIntent triggered	1 out of 3
block card	think my card is stolen	your card is an ace	target intent triggered	target intent triggered	1 out of 2

[20] https://www.nltk.org/.
[21] https://developer.oxforddictionaries.com/.

3.3 Discussion

The results of the experiment confirm the hypothesis that natural language understanding functionality in systems such as Amazon Alexa Skills and RASA NLU is vulnerable to being misled by malicious actors using utterances that are accepted by the system as a valid action trigger, but are unrelated to the relevant target command in terms of their meaning as understood by humans. The results of the experiment support concerns surrounding the implementation of voice control in sensitive areas such as banking.[22]

The results confirm that, whilst measures for enabling voice-controlled systems to reject irrelevant input, such as the FallbackIntent in Alexa Skills or the nonsense intent in the RASA NLU banking bot, do prevent such systems from simply accepting any utterance directed towards them as valid commands, this is not sufficient to prevent voice-controlled systems from accepting irrelevant utterances that have been crafted maliciously so as to mislead natural language understanding functionality. In the case of the Alexa Skill target system, some adversarial commands were identified as the target command with a sufficiently high level of confidence to avoid triggering of the FallBackIntent, whereas in the case of the RASA NLU target system, some adversarial commands were identified as the target command with a higher confidence level than the confidence level assigned to the nonsense intent. The success of some adversarial commands in triggering the target command indicates that the capacities of natural language understanding functionality in current voice-controlled systems to distinguish valid from invalid input and to identify the correct meaning of words in a given context can be easily undermined. These issues represent significant security vulnerabilities, in that they may enable a malicious actor to gain control of a system using utterances that are unlikely to be recognised by the system's human users as a voice command to their system. A notable characteristic of these attacks is that they have the potential to be plausibly deniable, in that a target system's execution of a target action in response to an unrelated utterance vocalised in its environment might be easily explained as being due to an error on the part of the target system, rather than to malicious intent on the part of the source of the utterance.

A clear limitation of the attacks demonstrated here with respect to the Alexa Skill target system is that they do not take into account the need for an attacker to activate the Alexa assistant and the target Skill using a wake-word or activation phrase. However, this limitation should not be viewed as one which cannot be overcome in future work. Due to the known presence of false positives with respect to wake-word recognition, it might be possible to trigger activation of the wake-word by using a single natural language word, other than the wake-word itself, as part of an unrelated utterance, in order to subsequently be able to execute an adversarial learning attack targeting natural language understanding to trigger a specific target command. This possibility was in fact demonstrated in

[22] See for example phys.org, 20th June 2018, 'Banking by smart speaker arrives, but security issues exist', https://phys.org/news/2018-06-banking-smart-speaker-issues.html.

an incident in which an Amazon Alexa device misinterpreted a word spoken in a private conversation as the wake-word 'Alexa', and subsequently misinterpreted other words in the conversation as commands to send a message to a contact, resulting in a recording of a couple's private conversation in their home being sent to a colleague.[23] Whilst this transmission of private information occurred as a result of error rather than malicious intent, it highlights the potential for spoofing of wake-word recognition and the inadequacy of wake-word recognition as a security measure.

4 Future Work and Conclusions

The experimental results presented here consolidate initial results presented in our earlier paper, confirming that speech recognition in voice-controlled systems is vulnerable to being misled by adversarial input consisting of nonsensical word sounds that are perceived by legitimate users of voice-controlled systems as having no meaning, and that natural language understanding functionality in voice-controlled systems can be manipulated using crafted utterances that retain elements of a target command, but that are perceived by naive listeners as being unrelated to the action that an attacker is seeking to trigger.

Future work should seek to demonstrate the types of attacks investigated here on different systems and on a broader set of target commands. With respect to the 'nonsense' attacks targeting speech recognition, whilst the target commands used in the experiment performed here were real commands actually executable by Google Assistant, the methodology applied in the experiment described here of assessing the target system's response based on transcription of audio input, rather than an actual performed action, potentially expands the range of target commands beyond actions that are within the scope of a target system's actual capabilities to actions that are currently hypothetical. Therefore it would be possible to investigate the vulnerability of hypothetical target actions to attacks of this type before the actions are actually implemented as part of a live system. The ultimate focus of future work should be to develop defence mechanisms that can make voice-controlled systems more robust to nonsense and missense attacks at a general level.

References

1. Carlini, N., et al.: Hidden voice commands. In 25th USENIX Security Symposium (USENIX Security 2016), Austin, TX (2016)
2. Zhang, G., Yan, C., Ji, X., Zhang, T., Zhang, T., Xu, W. (2017). DolphinAttack: inaudible voice commands. arXiv preprint arXiv:1708.09537
3. Bispham, M. K., Agrafiotis, I., Goldsmith, M.: Nonsense attacks on Google Assistant and missense attacks on Amazon Alexa. In: Proceedings of International Conference on Information Systems Security and Privacy (2019)

[23] See BBC News, 24th May 2018, "Amazon Alexa heard and sent private chat", https://www.bbc.co.uk/news/technology-44248122.

4. Bispham, M.K., Agrafiotis, I., Goldsmith, M.: A taxonomy of attacks via the speech interface. In: Proceedings of Third International Conference on Cyber-Technologies and Cyber-Systems (2018)
5. Papernot, N., McDaniel, P., Swami, A., Harang, R.: Crafting adversarial input sequences for recurrent neural networks. In: Military Communications Conference, MILCOM 2016–2016 IEEE, pp. 49–54. IEEE (2016)
6. Nowak, M.A., Krakauer, D.C.: The evolution of language. Proc. Nat. Acad. Sci. **96**(14), 8028–8033 (1999)
7. McCurdy, N., Srikumar, V., Meyer, M.: RhymeDesign: a tool for analyzing sonic devices in poetry. In: Proceedings of the Fourth Workshop on Computational Linguistics for Literature, pp. 12–22 (2015)
8. McTear, M., Callejas, Z., Griol, D.: The Conversational Interface. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-32967-3
9. Hazen, T.J., Bazzi, I.: A comparison and combination of methods for OOV word detection and word confidence scoring. In: Proceedings of 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing, p. 1. 397–400 (2001)
10. Raju, A., Panchapagesan, S., Liu, X., Mandal, A., Strom, N.: Data augmentation for robust keyword spotting under playback interference (2018). Proceedings of arXiv preprint arXiv:1808.00563
11. Roberts, A.C., Wetterlin, A., Lahiri, A.: Aligning mispronounced words to meaning: evidence from ERP and reaction time studies. Mental Lexicon **8**(2), 140–163 (2013)
12. Lippmann, R.P., et al.: Speech recognition by machines and humans. Speech Commun. **22**(1), 1–15 (1997)
13. Scharenborg, O., Cooke, M.: Comparing human and machine recognition performance on a VCV corpus (2008)
14. Bailey, T.M., Hahn, U.: Phoneme similarity and confusability. J. Memory Lang. **52**(3), 339–362 (2005)
15. Meutzner, H., Gupta, S., and Kolossa, D.: Constructing secure audio captchas by exploiting differences between humans and machines. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 2335–2338. ACM (2015)
16. Xiong, W., et al.: Achieving human parity in conversational speech recognition (2016). arXiv preprint arXiv:1610.05256
17. Carlini, N., Wagner, D.: Audio adversarial examples: targeted attacks on speech-to-text (2018). arXiv preprint arXiv:1801.01944
18. Kuleshov, V., Thakoor, S., Lau, T., Ermon, S.: Adversarial Examples for Natural Language Classification Problems. OpenReview submission OpenReview:r1QZ3zbAZ (2018)
19. Mesnil, G., et al.: Using recurrent neural networks for slot filling in spoken language understanding. IEEE/ACM Trans. Audio Speech Lang. Process. (TASLP) **23**(3), 530–539 (2015)
20. Cambria, E., White, B.: Jumping NLP curves: a review of natural language processing research. IEEE Comput. Intell. Mag. **9**(2), 48–57 (2014)
21. Tur, G., Deoras, A., Hakkani-Tür, D.: Detecting out-of-domain utterances addressed to a virtual personal assistant. In: Proceeding of Fifteenth Annual Conference of the International Speech Communication Association (2014)
22. Stolk, A., Verhagen, L., Toni, I.: Conceptual alignment: how brains achieve mutual understanding. Trends Cogn. Sci. **20**(3), 180–191 (2016)
23. Miller, G.A.: WordNet: a lexical database for English. Commun. ACM **38**(11), 39–41 (1995)

24. Kumar, A., Gupta, A., Chan, J., Tucker, S., Hoffmeister, B., Dreyer, M.: Just ASK building an architecture for extensible self-service spoken language understanding (2017). arXiv preprint arXiv:1711.00549
25. Kollar, T., et al.: The Alexa meaning representation language. In: Proceedings of Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 3, pp. 177–184 (2018)
26. Bocklisch, T., Faulkner, J., Pawlowski, N., Nichol, A.: Rasa: open source language understanding and dialogue management (2017). arXiv preprint arXiv:1712.05181
27. Qi, Y., Xiao, J.: Fintech: AI powers financial services to improve people's lives. Commun. ACM **61**(11), 65–69 (2018)
28. King, B.: Bank 4.0: Banking Everywhere, Never at a Bank. Marshall Cavendish Business, Singapore (2018)