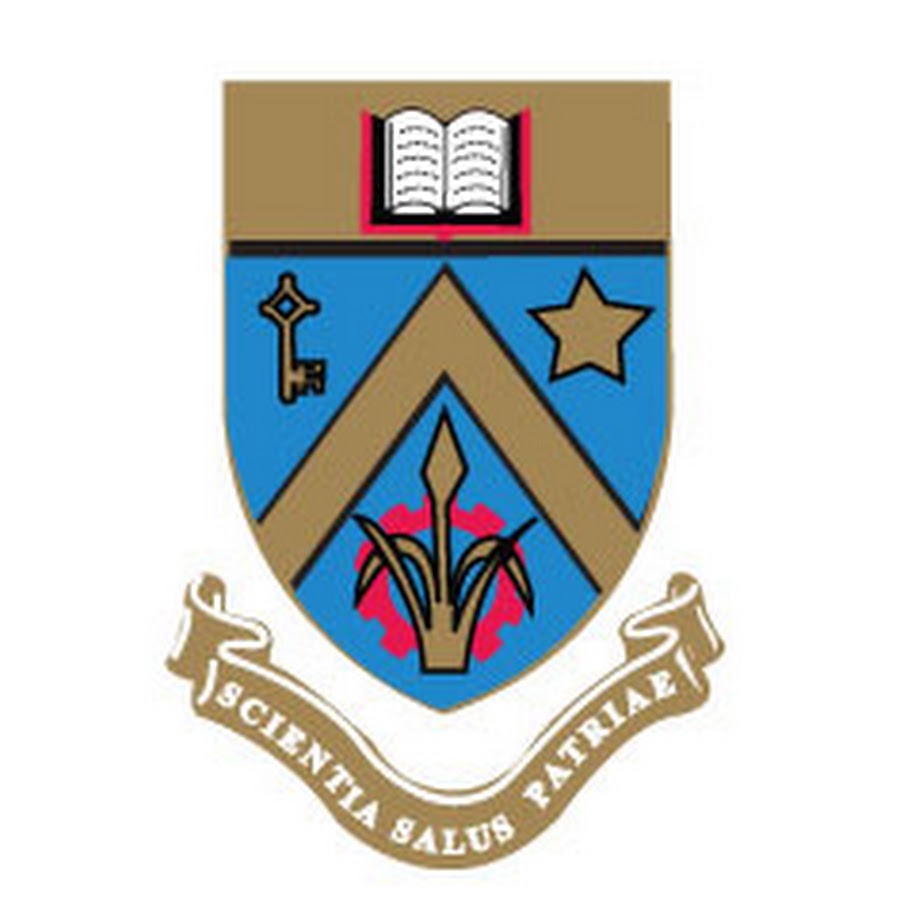
**Authors:**

Anastasia Lacrice

Laldeo Maroam

Nadeem Emritte

Intelligent Digital Coaching Assistant



MSc Applied Software Technologies

**March 2019**

**University of Mauritius**

Faculty of Information, Communication and Digital Technologies

**Artificial Intelligence**

Artificial Intelligence is a branch of computer science to construct machine that think. It includes games playing, expert systems, neural networks, natural language and robotics. Artificial Intelligence (AI) has significantly contributed to the advancement of Machine Learning techniques.

Today’s AI systems can interact with users, understand their needs, map their preferences and recommend an appropriate line of action with minimal or no human intervention. The most researched area of artificial intelligence is neural networks which are proving successful in several disciplines such as image recognition, voice recognition and Natural Language Processing (NLP).

Humans use natural language such as English, French etc. to communicate. One major challenge in computer science is concerned with creation of computers that can understand natural language. Natural Language Processing (NLP) is a field in computer science which helps to build intelligent chatbots.

The basic definition of chatbot is, it is a computer software program designed to simulate human conversation through text or audio messages. Since chatbots simulate an actual person, AI techniques are used to build them. One such technique within AI is Deep Learning which mimics the human brain. Basically, it finds pattern from the training data and uses the same patterns to process new data. Deep Learning is promising to solve long standing AI problems like Computer Vision, voice Understanding and Natural Language Processing (NLP).

The conversational agents available today, like Apple’s Siri, Microsoft’s Cortana, Google Assistant and Amazon’s Alexa cannot keep track of the context and suffer in long-ranging conversations. Another limitation of these chatbots is that they are designed specifically for helping a user with some specific problems, hence restricting their domain. They are unable to make a coherent and engaging conversation between human beings on popular topics such as recent news, politics and sports.

(Goodfellow, et al., 2016) has categorized AI into three approaches:

1. Knowledge Base
2. Machine Learning
3. Representation Learning
4. **Knowledge Base**

These AI models are powered by formal rules. Bots designed using this model are not very smart. They respond to very specific commands. Knowledge based systems have been helping to solve problems which are intellectually difficult, but easy for machines. (Cava, 2016)

1. **Machine Learning**

Then we have advanced version bots which use machine learning. Machine Learning tries to overcome the limitations of hard-coded rules of the Knowledge Base approach of AI. The basic function of Machine Learning is to extract patterns from data instead of relying on rules. Linear regression and Bayes methods are simple Machine Learning techniques that have been used to make simple models such as housing price prediction and spam email detection. They work by learning the correlation between features and the output class or value.

One major drawback of this AI model is that it is restricted to the features which are designed by the modeler. This essentially means that each entity should be represented as a set of features. For instance, if we take the problem of face detection, the modeler can associate a set of features such as shape and structure to a face, but it is difficult to model on a pixel-to-pixel basis.

Another shortcoming of this approach is that representation of data is very important. The best predictions can be achieved by using the process of feature engineering, which involves representing data for a model as a preprocessing step.

Both knowledge-base and machine learning approaches require us to have substantial domain knowledge and expertise. One way to overcome this problem is to map the representation to output and the representation itself. This is where representation learning comes into the picture.

1. **Representation Learning**

The rigidity of knowledge-base and machine learning approaches has led to the used of representation learning. Representation learning allow the model to be able to learn the representation of data itself. Learned representations often result in much better performance than can be obtained with hand-designed representations (Goodfellow, et al., 2016).

For example, we take the example of face detection again. As humans, we can recognize a face from different viewing angles, different lighting conditions, different facial features such as spectacles or beard. This representation of data is abstract and can be expressed as a hierarchy of simple to complex concepts which allow us to make sense of different data that we encounter. However, this type of information is almost impossible to model because of the randomness of the data.

Deep Learning tries to overcome this problem by expressing complex representations in terms of simpler representations (Goodfellow, et al., 2016).

Deep learning is a subset of representation learning, having multiple layers of neurons to learn representations of data with multiple levels of abstraction (Lecun, et al., 2015).

Representation learning models the human brain, with brain neurons analogous to computing units and the strength of connections between the neurons analogous to weights. Deep Learning architecture is like an Artificial Neural Network (ANN), but with more hidden layers (hence, more neurons) which allows us to model the more complex functions of our brain. This architecture is shown in Figure 1.



Figure 1: Deep Learning Architecture [3]

**Increase in Popularity of chatbots**

A chatbot is a piece of automated messaging software that uses Artificial Intelligence (A.I.) to converse with people (Cooper, 2019).

The term “ChatterBot” was originally coined by Michael Maudin who is the creator of the first *Verbot*[[1]](#footnote-1) named Julia in 1994 (Mauldin, 1994). Today, the trend is to develop chatbots for mobile messaging applications (Følstad & Brandtzae, 2017).

Most chatbots are accessed via virtual assistants on messaging apps such as Facebook Messenger or WeChat, or individual organisations’ apps and websites. Cortana (Microsoft 2015), Google Now (2012), Siri (Apple, 2011), Alexa (Amazon 2015) are among the top voice-enabled digital assistants.

The growing interest in chatbots is spurred by numerous developments in artificial intelligence (AI) and machine learning. Major technology companies see chatbots as the next popular technology; Microsoft CEO Satya Nadella said, “Chatbots are the new apps” (Cava, 2016).

In 2016, Facebook and Microsoft provided resources for creating chatbots to be integrated into their messaging platforms, Messenger and Skype. As of the 2018 *F8 Conference[[2]](#footnote-2)*, Facebook had 300,000 active Messenger bots which represented three times the amount in 2017 (Cooper, 2019).

**Tools and Models used by Microsoft to achieve speech recognition**

In October 2016, Microsoft researchers achieved speech recognition milestone. Microsoft created a technology that recognizes the words in conversation as well as a person does. A word error rate (WER) of 5.9% was reported which is about equal to that of people who were asked to transcribe the same conversation (Xiong, et al., 2017). It was the lowest WER ever recorded against the industry standard switchboard speech recognition task.

The accomplishment was achieved by the systematic use of neural network technology in all aspects of the system. Microsoft Cognitive Toolkit (CTNK), a system for deep learning was used by the research team to reach the human parity milestone. The system implements advanced optimizations that enable deep learning algorithms to run an order of magnitude faster than before.

The push that led the researchers to success was the use of neural language models in which words are represented as continuous vectors in space, and words like “fast” and “quick” are close together.

Deep neural networks work best when a large amount of data-called training sets- is used to train the computer system to recognize patterns from inputs such as images or sounds.

Both IBM and Microsoft cite the advent of deep neural network as the main reason for advances in speech recognition (Linn, 2016).

Another contributor that has helped Microsoft researchers to reach the human parity in speech recognition is a breakthrough for parallel training on graphics processing units, or GPUs.

Although GPUs are designed for computer graphics, researchers have discovered that they are the ideal tools for processing complex algorithms like the one used to understand speech.

**Artificial Neural Network (ANN)**

Artificial Neural Network (sometimes called connectionist systems, parallel distributed processing or adaptive networks) is an interconnected group of nodes that forms a computing system that is inspired by, but not identical to biological neurons in a brain. The system “learns” to perform tasks by using examples without programmed task-specific rules.

The root of this topic can be traced back to the 1950s and 1960s to research into Perceptron led by Frank Rosenblatt (Rosenblatt, 1957), inspired by earlier work by Warren Mc Culloch and Walter Pitts on MCP neuron (McCulloch & Pitts, 1943). The early neural-network researcher Bernard Widrow’s research into adaptive linear filters also marked the beginning of Artificial Neural Network (ANN) in the 1950s.

Rosenblatt’s major achievement has been to show that, by relaxing some of the MCP’s rules artificial neurons could actually learn from data. More importantly, he came up with a **Supervised Learning Algorithm** for this modified MCP neuron model – *The Perceptron*, that enabled the artificial neuron to figure out the correct weights directly from training data by itself. Supervised Learning is a type of Machine Learning used to learn models from labeled training data. It enables output prediction for future or unseen data.

Perceptron

A perceptron is a neural network unit (an artificial neuron) that does certain computations to detect features or business intelligence in the input data.

The Perceptron works by taking several binary inputs and produces a single binary output.

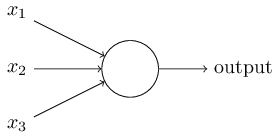


Figure 2: A simplified model of Perceptron

In the example above, the Perceptron has only three inputs, and produces a single binary output. In general, more input or fewer inputs are used.

The Perceptron Learning Algorithm is an algorithm for supervised learning of binary classifiers. Binary (or binomial) classification is the task of classifying the elements of a given set into two groups (e.g. classifying whether an image depicts a table or a chair) based on a prescribed rule. The Perceptron algorithm learns the weights for the input signals in order to draw a **linear** decision boundary. Rosenblatt’s perceptron can handle only classification tasks for linearly separable classes.

As discussed earlier, the major achievement of Rosenblatt was not only to show that his modification of the MCP neuron could actually be used to perform binary classification, but also to come up with a fairly simple and yet relatively efficient algorithm enabling the perceptron to learn the correct synaptic weights w from examples. This algorithm is given below:

The Perceptron Function

Perceptron Learning Rule states that the algorithm would automatically learn the optimal weight coefficients. The input features are then multiplied with these weights to determine if a neuron fires or not.

The Perceptron function is a function that maps its input “x” with the **learned** weight coefficient represented by “*w*” to express the importance of the respective inputs to the output. The neuron’s output value “f(x)”, is determined by whether the weighted sum  is greater than 0.

In the equation shown above

“w” is the vector of real-values weights

“b” is the bias (an element that adjusts the boundary away from origin without any dependencies on the input value)

“x” is the vector of input x values

“m” is the number of inputs of the Perceptron

A Boolean output is based on inputs such as salaried, married, age, past credit profile, etc. It has only two values: Yes and No or True and False. The summation function “∑” multiplies all inputs of “x” by weights “w” and then adds them up as follows:

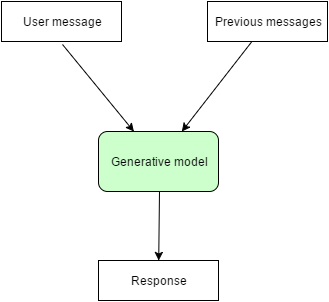
The output can be represented as “1” or “0”. It can also be represented by “1” or “-1” depending on which activation function is used.

**Architecture Model of a chatbot**

The architecture of a chatbot depends on the core purpose of development. A chatbot can respond in two ways: it can either generate a response from scratch as per machine learning models or it can select an appropriate response from a library of preset responses.

**Generative Model**

The generative model is used to build smart bots. However, these types of bots are rarely used as they are quite advanced in nature and requires the implementation of complex algorithms.



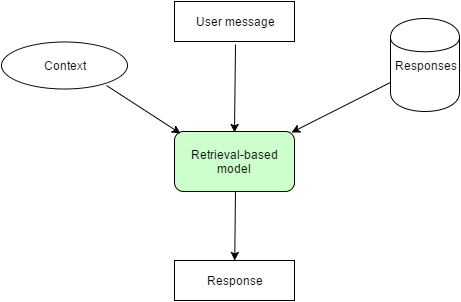
Training of generative bots is time consuming and requires a lot of effort by giving millions of examples.

This how deep learning model can engage in conversation. However, still, user are never sure what responses the model will generate.

**Retrieval-based Model**

Retrieval-based model is easier to build and is much more reliable that generative models. “Reliable” implies that we have an idea what types of possible responses will be delivered by the chatbot since the bot selects its response from a predefined list of messages.

Several APIs and algorithms are available for chatbots using this architectural model. The diagram below is a simple representation of the retrieval-based model. The retrieval-based model deliver response based on message and context of the conversation as well as the predefined list of messages available in the database.



**Response Generation Mechanism of chatbots**

**Pattern-Based Heuristics**

Chatbots can generate responses in two ways: by using if-else conditional logic or by using machine learning classifiers.

AIML is used for writing patterns and responses in chatbot development. Given a predefined rich pattern and a natural language processing pipeline, AIML can be used to build a smart chatbot.

These bots parse user messages find synonyms and concepts, tag parts of speech and find out which rule matches the user query. They do not use machine learning algorithms or other APIs unless specially programmed.

**Intent classification using machine learning**

Pattern-based heuristics is achieved by using machine learning technology. **Scikit-learn** is a popular machine learning library that helps in executing machine learning algorithms. With Scikit-learn, developers have the option to use one of cloud APIs among api.ai, wit.ai and Microsoft LUIS.

Pattern-based heuristics delivers good result. One shortcoming is that the pattern should be programmed manually. This is a tedious task because the chatbot must differentiate between hundred of intents for different scenarios.

Pattern-based chatbot requires a training set of at least thousand of examples that are most likely to be faced by the chatbot. The chatbot can be trained to pick up patterns of data and learn from it.

**High-Level Architecture of the chatbot**

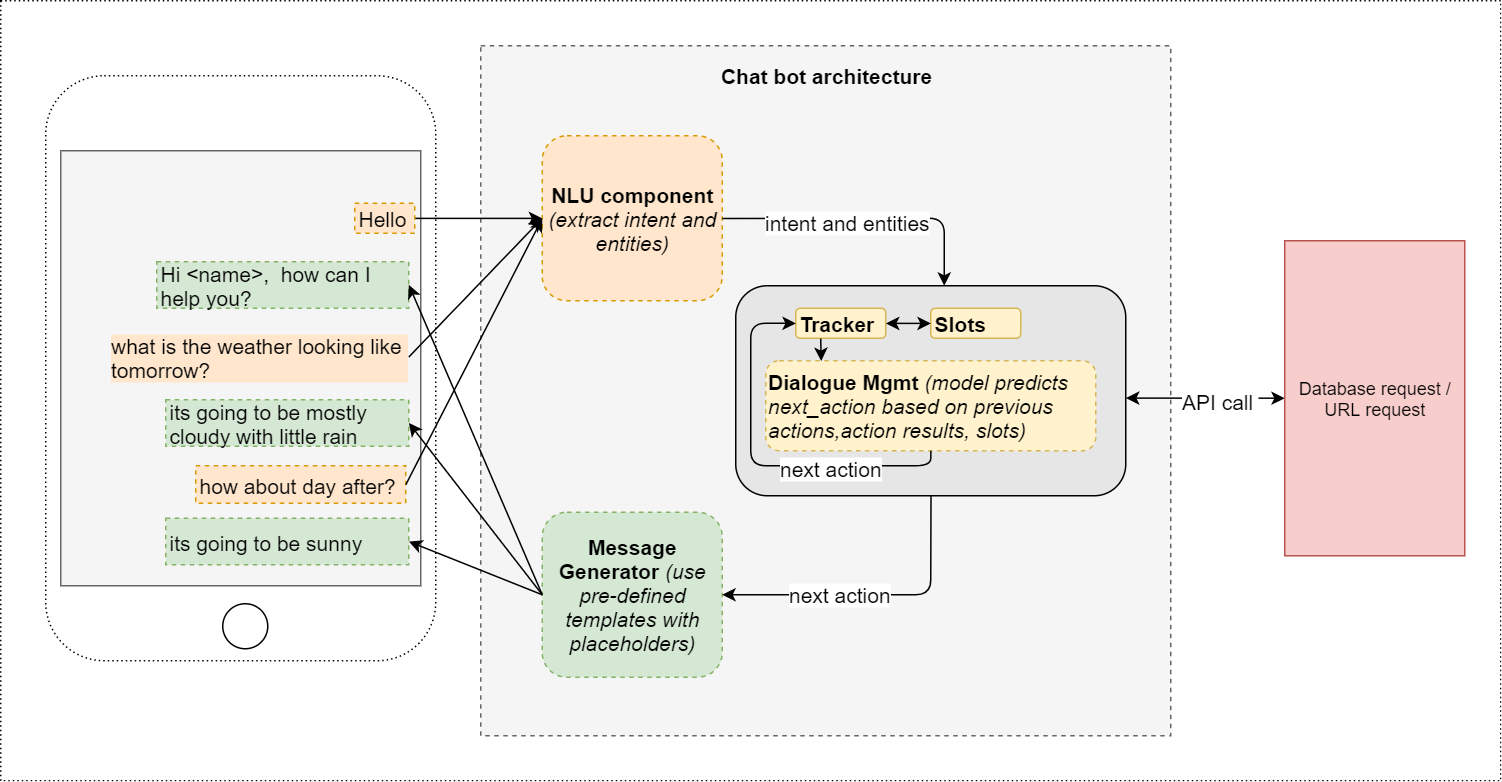


Figure 3: Chabot Architecture

An Artificial Intelligence/Machine Learning driven architecture: The model learns the action-based training data provided (unlike a traditional state machine architecture that is based on coding all the possible if-else conditions for each possible state of the conversation.

Languages that can be used are Node.js, Ruby, Python and Java. However, Java lacks native support for many platforms.

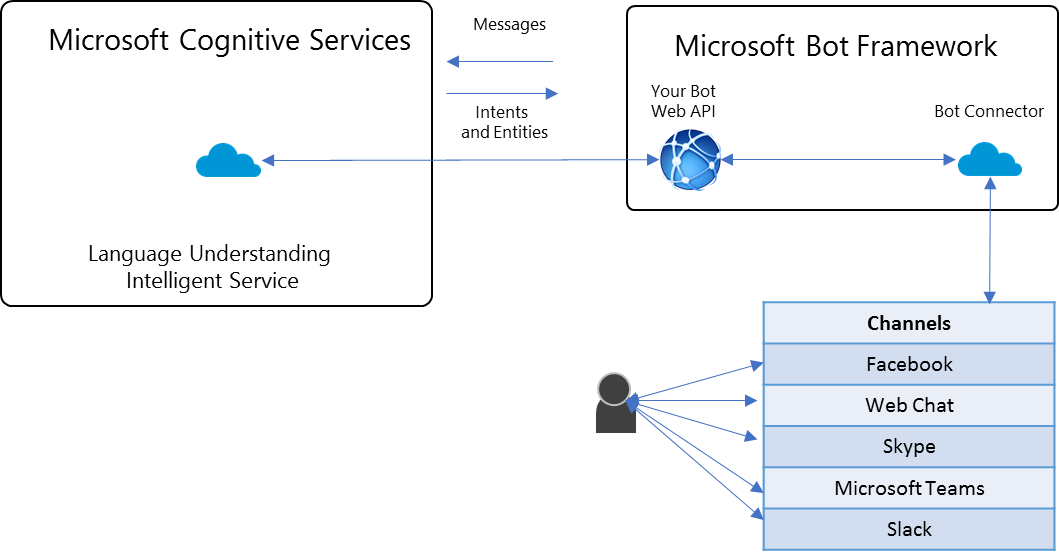
Natural Language Processing services that can be used to make the bot understand the messages sent by users are Facebook’s wit.ai, Google’s Dialog Flow or Microsoft Luis.

Recommended databases are MongoDB, PostgreSQL and MySQL.

**Microsoft Technologies**

Microsoft Bot Framework is a complete framework to build and deploy bots on various channels like Facebook, Skype and more.

Developers only need to build their web API using this framework. Azure and Bot Application can be used as server to host bots. Additional API from Microsoft Cognitive Services can be used to add intelligence to bots- LUIS, which helps parse messages into intents and entities.



The basic architecture of bots starts from the Microsoft cognitive services which helps to create AI based apps and provide various API like vision, speech, Language and Video.

The Microsoft cognitive Services provides the LUIS which helps to convert messages to the meaningful data.

1. **LUIS**

Luis is an intelligent service which reads the input messages from users and convert to the sample intents and entities by which you can extract what is the user’s intention and other related entities of message.

1. **Microsoft Bot Framework**

Bot Framework is a complete setup to building the bots. The framework manages the following:

1. The convention State
2. Message routing
3. Adding bots to channels
4. Registration of your bot
5. **Bot Connector**

Bot connector provides various services to enable communication between bot web API and user and including service bot registration, conversation and state management and channel management. Bot connector have a great feature-it gives a power to connect your bot with multiple channels like Facebook, Skype, Microsoft Teams and it is even adding more channels. So, without thinking about the channel, developers can focus on developing the conversation logic.

1. **Web API**

The web API is where the actual logic of the bot is written. The conversation is created using dialogs and forms. LUIS will then process the result and send the reply to the user.

Visual Studio Online templates provide bot application templates to create BOT web API. The BOT keys and LUIS keys and Microsoft Bot Framework emulator is also provided which help to test the bot.

When building bots, components like Dialogs and FormFlow can be used to build strong conversations including message replay and context to manage the message chain.

**Dialogs**

The dialog gives you most of the flexibility you need but it also requires you to manage message parsing and manage any state your dialog may need to reference. Dialog is the nice way to build conversation.

**FormFlow**

It provides a guided conversation with the ability to provide options, do input validation, and confirm user responses. They are good choices for replacing existing web forms with something more like natural conversation. The dialog can be called with another dialog, and this feature allows you to build conversations that can be reused. You can also have an option to call a form within a dialog. This allows you to build mixed features of both.

CHECK TENSORFLOW

# **References**

Cava, M. d., 2016. *USA Today: Microsoft CEO Nadella: "Bots are the new apps".* [Online]   
Available at: https://www. usatoday.com/story/tech/news/2016/03/30/microsof-ceo-nadella-bots-new-apps/82431672/  
[Accessed 10-13 July 2019].

Cooper, P., 2019. *Hootsuite: The Complete Guide to using Facebook Messenger Bots for Business.* [Online]   
Available at: https://blog.hootsuite.com/facebook-messenger-bots-guide/  
[Accessed 12 August 2019].

Dale, R., 2016. Natural Language Engineering. In: *The return of the chatbots.* s.l.:s.n., pp. 811-817.

Følstad, A. & Brandtzae, P., 2017. The return of the chatbot. *Interactions,* 24(4), pp. 38-42.

Goodfellow, I., Bengio, Y. & Courville, A., 2016. *Deep Learning.* s.l.:MIT press.

Lecun, Y., Bengio, Y. & Hinton, G., 2015. Deep Learning. *Nature,* 521(7553), p. 436.

Linn, A., 2016. *Microsoft Official Blog: The AI Blog Historic Achievement: Microsoft researchers reach human parity in conversational speech recognition.* [Online]   
Available at: https://blogs.microsoft.com/ai/historic-achievement-microsoft-researchers-reach-human-parity-conversational-speech-recognition/  
[Accessed 12 August 2019].

Mauldin, M. L., 1994. *ChatterBots, TinyMuds, and the Turing Test: Entering the Loebner Prize Competition.* Pittsburg, AAAI Press.

McCulloch, W. S. & Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics,* 5(4), pp. 115--133.

Rosenblatt, F., 1957. *The perceptron, a perceiving and recognizing automaton Project Para.* s.l.:Cornell Aeronautical Laboratory.

Rosenblatt, F., 1960. Perceptron Simulation Experiments. *Proceedings of the IRE,* 48(3), pp. 301-309.

Xiong, W. et al., 2017. *Achieving human parity in conversational speech recognition,* s.l.: Microsoft Research.

Xu, A., Guo, Y., Sinha, V. & Akkiraju, R., 2017. *A new chatbot for customer service on social media.* s.l., Proceedings of the ACM Conference on Human Factors in Computing Systems.

1. Verbot is basically a shorter term for verbal bots. Maudin used this term to describe the conversational nature of Julia. [↑](#footnote-ref-1)
2. The F8 conference is an annual conference hosted by Facebook where developers, innovators, creators and entrepreneurs from around the world come together for the future of technology. [↑](#footnote-ref-2)