- URL: http://www.alicebot.org/anatomy.html  
- Title: The Anatomy of A.L.I.C.E  
- Authors: Wallace, Richard S  
- Notes:

##Review

The Anatomy of A.L.I.C.E is a paper that explains and represents the technical side of the Artificial Linguistic Internet Computer Entity, the winner of the Loebner Prize as “the most human computer”. The paper goes into detail on Turing’s 1950’s paper on the Original Imitation Game (OIG) where a man and woman are asked questions remotely, via text, by an interrogator. The man is instructed to lie and ensure the interrogator is not able to find out that the man is a woman. Turing proposes replacing the man with a robot, and so came the description of the Turing Test.

The paper goes on to criticise ALICE’s winning of the Loebner Prize, as it is purely designed to beat the Turing Test and is not designed for use in real world applications. ALICE consists of a huge database of AIML (XML style) elements, each combining questions and answers or stimulus and responses in an attempt to match any potential question with a reasonable response.

In this way, ALICE is based on the original ELIZA program, an earlier natural language processor in development from 1964-1966. The original ELIZA program was released under the guise of ‘Doctor’ for use by nontechnical staff in MIT. The issue with this was that nontechnical staff thought that this ‘Doctor’ was a real therapist and spent hours revealing personal problems to a script that merely either rewrote what they had said with the pronouns reversed, or answered with pre-prepared statements based on simple pattern matching. This, at the time, was thought to be rather dangerous.

The AIML elements are designed to symbolically reduce sentences into simpler formats to understand, or match elements and patterns. For example, ALICE programmers preferred to reduce sentences to the simplest possible form, which would be referred to as the template.

<category>  
<pattern>DO YOU KNOW WHO \* IS</pattern>  
<template><srai>WHO IS <star/></srai></template  
</category>

‘SRAI’ allows AIML to commit recursion to further simplify patterns.

It then attempts to “divide and conquer” sentences by reducing it to subsentences. For example, if a sentence begins with “Yes” (to answer a question), and then has more words, then both sections would be treated as subsentences as go down separate paths in the AIML tree.

The paper talks at length at how the ALICE bot continues to try and build its responses from the context of conversation, including previous responses, and how it is capable of remembering pronoun bindings using predicates. However, the paper does not conjecture on possible improvements to the ALICE bot nor what other works may be in process. It also does not describe the computational requirements or any examples of how the code that runs the AIML works, which is disappointing.

##Citation

Wallace, R. S. (n.d.). The Anatomy of A.L.I.C.E. Retrieved from http://www.alicebot.org/anatomy.html

- Title: Chatbots: Are they really useful?  
- Authors: Atwell, Eric; Shawar, Bayan Abu  
- Notes:

##Review

This paper is essentially a review of many different chat bots and their uses to date, many of which use AIML elements and are based on ALICE. It has basic explanation of ALICE’s pattern matching algorithm and how it deals with incoming input. For example, all input has punctuation removed, is split into multiple sentences as needed, and then will try and match word by word to obtain the longest pattern match, which would be expected to be the best one. It will then use those matched patterns to determine which templated response to use.

The current public-domain ALICE “brain” has been built up by Dr Richard Wallace, and contains more than 50,000 categories of possible patterns, but all these categories were hand coded, which is extremely time consuming if one were interested in building a chat bot for themselves. To combat this idea, the paper also includes information about a Java program that attempts to automatically turn text into AIML style scripting. However, this was found to not be satisfactory to trial users.

The paper also discusses Pandorabot, a website used to build and deploy chat bots using the AIML XML system, which could be useful.

While the paper does include some original work, much of the reviews and comparisons of the chatbot systems are references to the same author, and there is little evidence from other authors. The paper also rarely reflects upon bots that do not use the AIML system.

##Citation

Shawar, B. A., & Atwell, E. (2007). Chatbots: are they really useful? *LDV-Forum*.

- Title: Using dialogue corpora to train a chatbot  
- Authors: Shawar, Bayan Abu; Atwell, Eric  
- Notes:

##Review

This paper, related to the previous paper, presents two chatbot systems, ALICE and Elizabeth and attempts to compare and pattern matching techniques of each system. Using the Dialog Diversity Corpus, a database of formatted human texts that can be used for human interaction research, the researchers attempt to convert natural language into AIML format.

After a section explaining symbolic reduction and how the AIML format handles recursion, which interestingly seems to be taken straight from the original paper by Dr Richard Wallace, with no direct reference, the paper moves on to the Elizabeth chat bot.

Elizabeth is a much-improved adaptation of the Eliza program, but much simpler in terms of database management than ALICE. Elizabeth uses a single script to run the bot, and may contain at most four parts. The first part deals with welcoming the user, and what to do in the case of an empty message, or a message which it does not understand.

The second part of the script deals with transformations, such as “MUM -> mother”. This can be seen as a rudimentary form of sentence reduction, as it attempts to give the meaning of multiple words into one “intent”.

The third part of the script deals with output transformation. This, again, is a simplified version of pronoun reversal such as “my -> your” and “you is -> you are”.

The final part of the script deals with keyword transformation, which is effectively the picking out of keywords to attempt to create a cohesive answer. For example, if the script read:  
K I LIKE [string]ING  
R HAVE YOU [string]ED AT ALL RECENTLY?

(With K and R being keyword and response rule starts for the Elizabeth interpreter)  
The user would be able to enter “I like gaming” and the response would be “Have you gamed at all recently?”.

Elizabeth can get very complex, including implementing grammatical rules and advancing its pattern matching algorithm, but the fact remains that Elizabeth is unable to use any sort of “big data” to automatically train itself without assistance from a third-party application, nor can it stray particularly far from exact pattern matches without extremely advanced programming.

The paper ends with a list of problems encountered when dealing with their chosen text database, the Dialog Diversity Corpus, noting that there are no standards formats to distinguish between speakers, sub-standard annotation, and scanned images not being converted into text correctly.

##Citation

Shawar, B. A., & Atwell, E. (2003). Using dialogue corpora to train a chatbot.

- Title: Implementation of ALICE chatbot as domain specific knowledge bot for BRAC U (FAQ bot)  
- Authors: Rahman, Johan  
- Notes: This was a thesis and I sure as hell hope mine isn’t going to be this simplistic.

##Review

This paper follows the implementation of a chat bot with domain-specific knowledge about a university, essentially operating as a frequently asked question bot. Their goal is to show how accurate such a system can be, and how it can be improved based on a specific domain.

Domain-specific knowledge is a knowledge base relating to a particular system. In this case, frequently asked questions about BRAC University in Bangladesh. This information was formatted with a modified conversational base already implemented into ALICE, allowing the bot to have its conversation focused only on topics related to the domain.

Creating the bots knowledge bases consisted of extensive brainstorming and writing down as many questions in as many different formats or layouts as possible, to assist in the bot being able to intelligently match patterns, and to make it less likely that a relevant question would be misunderstood or ignored entirely. This included the use of wildcards, for example, “\* about cse370” would assume you were talking about a course, cse370, and would ignore other text.

The bot follows an interesting system for getting an idea about a topic. A user can advance into another topic by mentioning a certain thing, such as “admission”, and the bot will then move on to reading data from the “admission” AIML file instead of the “general” AIML file.

The paper compares between two potential bot setups. The first setup includes only very basic general responses, but the full range of FAQ responses. The second setup includes all general responses from the ALICE source, and the full range of FAQ responses. The first setup intentionally limits conversation from straying too far from the knowledge domain, but potentially could not seem as real or be as comfortable of a conversation for the end user. It was found that in fact the limited setup was preferred, as it came up with much more satisfactory answers a higher percentage of the time, as questions were more to do with domain specific knowledge.

The paper does not explore outside of the ALICE chatbot, nor does it look into any other ways of increasing the size of the knowledge base aside from manually writing new AIML.

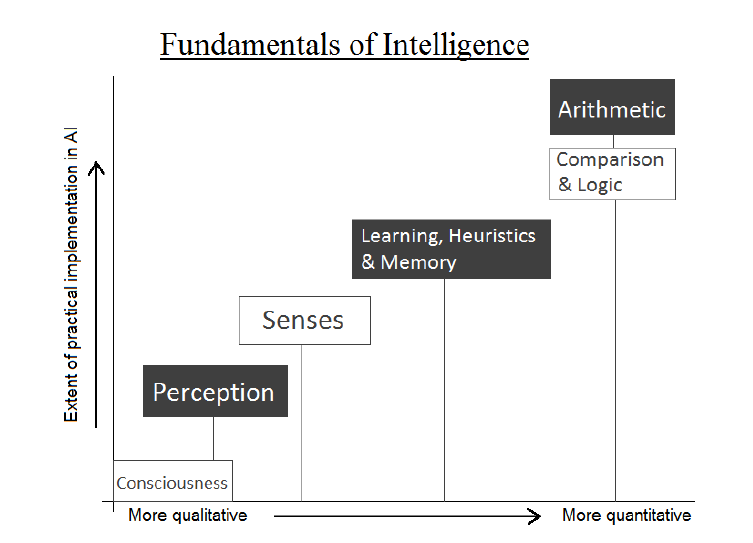
##Citation

Rahman, J. (2012). Implementation of ALICE chatbot as domain specific knowledge bot for BRAC U (FAQ bot).

- Title: A Study of Today’s AI through Chatbots and Rediscovery of Machine Intelligence  
- Authors: Khanna, A., Pandey, B., Vashishta, K., Kalia, K., Pradeepkumar, B., & Das, T  
- Notes:

##Review

According to this paper, the evolution of AI has been limited around some key points, such as databases being fed-in manually, or targeting beating the Turing test. Today’s AI systems only “pretend” to act like intelligent entities instead of actually being one. Rule based systems just follow a large set of if-else based logic, rather than applying actual reasoning.

To experiment, they created two chatbot programs. One using traditional AIML, which also gave them the advantage of also using ALICE’s AIML set directly, and one they created in C++ called FUTURE. FUTURE had an advantage over AIML – Being written in C++, it could reference other functions and programs when required, including offering basic arithmetic, trigonometric functions and even differentiating simple expressions.

It was stated that making the AIML bot was significantly easier and faster due to it already having many AI-oriented features. The AIML bot was tested for about 1500 queries, and gave suitable replies to around 1200, for an accuracy of 80%. However, the FUTURE bot was not tested in a similar fashion.

The end goal of the paper was to redefine “Machine Intelligence”. An “Intelligent” system must have all the fundamentals of intelligence (Figure to the right). A partially intelligent system could exhibit some of the fundamentals, such as Chat Bots, which can compare, have logic & reasoning, heuristics, and memory. It can even include the ability to perform arithmetic operations.

However, this papers conclusion could be quite controversial, as there are many competing papers on the definition of AI \*SOURCE\*.

##Citation

Khanna, A., Pandey, B., Vashishta, K., Kalia, K., Pradeepkumar, B., & Das, T. (2015). A Study of Today’s A.I. through Chatbots and Rediscovery of Machine Intelligence. *International Journal Science and Technology*, *8*(7), 277–284. https://doi.org/10.14257/ijunesst.2015.8.7.28

- Title: ICE: Enabling Non-Experts to Build Models Interactively for Large-Scale Lopsided Problems  
- Notes:

##Review

Human teachers guiding learning machines presents numerous benefits and challenges according to research by Microsoft. If a teacher can provide additional information and metadata as a learning task progresses, it becomes an *interactive learning* problem. The paper explains the need and benefits of ICE, a platform that accommodates the two-way communication channel needed for efficient interactive learning. The platform has two goals, two produce valuable & deployable models, and to support research on both learning and user interface challenges of the interactive learning problem.

A “lopsided” problem is defined as a problem where not all the data is available. For example, a book review not being correctly labelled as a book review. It could take hundreds of thousands, if not millions of pages for learning machine to correctly predict what a book review is, which is incredibly inefficient. To overcome the issue, they turned to interactive machine learning, which allows human input, training, scoring, and machine feedback in a real-time loop.

Using this method, a single teacher performs all the functions of a domain expert, the labeller and the machine learning expert. At each step, the teacher learns from the machine how the machine is functioning, and machine benefits from the human guidance. The feedback from the machine allows the teacher to gain expertise to best guide the training process without possessing machine learning expertise.

The paper goes on to describe the using of the ICE program, and how your average user would update labels and system states. The paper also describes system architectures, including the extremely powerful servers needed for sub-second delays for the teacher.

While the paper goes into detail on the inner-workings of their own system, they do not really compare to other systems that exist, even though they mention several, such as Information Retrieval. However, their argument is that ICE differs so greatly from other work that it would be erroneous to compare them, as it can be applied to effectively any data type.

##Citation

Simard, P., Chickering, D., Lakshmiratan, A., Charles, D., Bottou, L., Garcia, C., … Suh, J. (2014). ICE: Enabling Non-Experts to Build Models Interactively for Large-Scale Lopsided Problems.

Other upcoming reviews:

- Title: Structured Training for Neural Network Transition-Based Parsing  
- Authors: Weiss, David; Alberti, Chris; Collins, Michael; Petrov, Slav  
- Notes: Potentially way too complex for me to understand – Come back to later

##Review

##Citation

Weiss, D., Alberti, C., Collins, M., & Petrov, S. (2015). Structured Training for Neural Network Transition-Based Parsing.

- Title: Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS)  
- Notes:

##Review

##Citation

Williams, J. D., Kamal, E., Ashour, M., Amr, H., Miller, J., & Zweig, G. (2015). Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS).

- Title: Applying Chatbots to the Internet of Things: Opportunities and Architectural Elements  
- Notes:

##Review

##Citation

Kar, R., & Haldar, R. (2016). Applying Chatbots to the Internet of Things: Opportunities and Architectural Elements.