

Mini-Project Report:

Saya ChatBot

**Guidance: Professor Shlomi Dolev
Mr. Michael Orlov**

**By: Aleksandra (Sasha) Kugel
Shira David**

**Ben-Gurion University of the Negev
Department of Computer Science
January 2008**

Table of contents:

1. Introduction

2. Architecture

3. Implementation

4. Scenarios

5. Conclusions and Future directions

6. Bibliography

1. Introduction:

SAYA is a robot that was purchased by the computer science department, at Ben-Gurion University. She functions as a receptionist and located in the entrance hall of “Alon” building. The robot was brought in order to improve her abilities and show its technology and abilities to the visitors.

SAYA is human-like and her ability of speech and utilization of her facial expressions to express feelings makes her communicatively capable.

SAYA was developed by Professor Hiroshi Kobayashi, a member at the Tokyo University of Science.

Ben-Gurion faculty members who made it possible to purchase SAYA are:

- Prof. Shlomi Dolev, the Chairman of Frenkel Center of Computer Science.
- Prof. Matya Katz, The Chairman of Computer Science Department, Ben Gurion University.
- Prof. Eyal Shimony, the Head of the Paul Ivanier Center for Robotics Research and Production Management.
- Dr. Ohad-Ben Shahar, Member of Computer Graphics research group, Computer Science Department, Ben Gurion University.

Our project involves SAYA's ability to communicate with a sense of logic. In order to perform her duty as a receptionist, SAYA must know how to answer whenever any human speaks to her. Our mission was to enlarge SAYA's vocabulary, and enable her respond appropriately to a large range of information obtained from the human, all of which allows her to conduct dialogs with humans.

This idea of a machine talking to a human was first explored by Joseph Weizenbaum. Weizenbaum was one of the fortunate few to join the embryonic MIT Artificial Intelligence Lab in the early 1960s. His most celebrated accomplishment was the development of ELIZA. ELIZA is a web robot and can conduct dialogs with humans through a web interface. ELIZA is based on very simple pattern recognition, and on a stimulus-response model.

It was ELIZA's development that inspired Richard Wallace to create a new chatbot named ALICE. ALICE's development began in 1995 and rewritten in Java at 1998. The current incarnation of the Java implementation is called Program D. Program D uses an XML Schema called AIML (Artificial Intelligence Markup Language) for specifying the heuristic conversation rules.

The original author of the project was Richard Wallace, but subsequent to the 2001 publication of an AIML specification, numerous other developers have taken up where Wallace left off, implementing open source AIML interpreters in a variety of programming languages, publishing AIML sets in various human languages, and continuing the spread of the technology as an open source venture.

This report will demonstrate our implementation of a package that enables SAYA to conduct dialogs based on AIML patterns, based on the example of ALICE.

2. Architecture

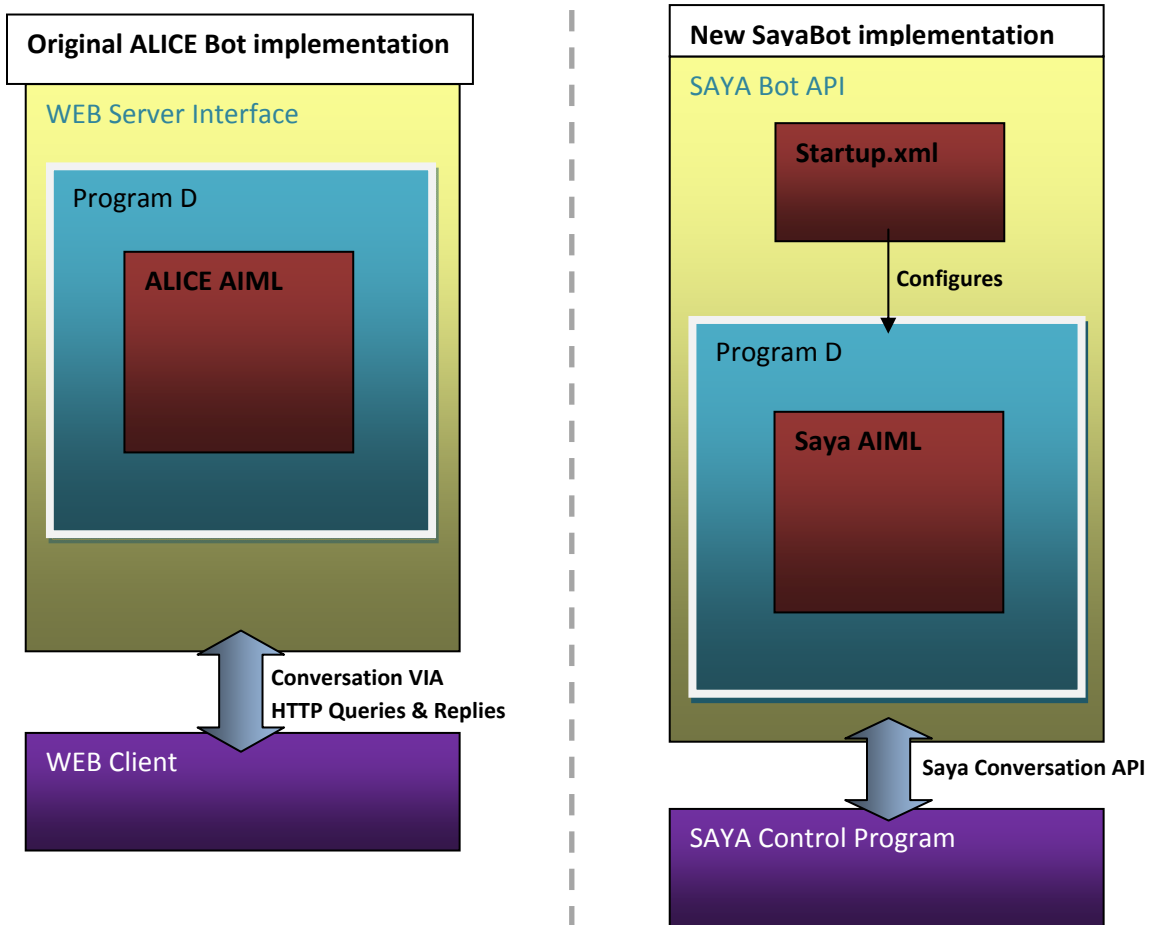
Since Program D and ALICE's AIML files are open sourced, we were able to make changes on those files, to suit SAYA.

We used ALICE's Program D and some of her AIML files. We changed the Program D Java code and wrapped it with our own "SayaBot" interface. We also built new AIML files to suit SAYA and her character.

Our AIML files contain information that is relevant for SAYA and is based on the ALICE AIML structure, modifying conversation topics and personality traits which were considered unique for ALICE's personality.

In order to start our program we defined an XML schema, called "startup.xml". This file contains configurative information and personality definitions for Saya (like her birthday, her favorite movie and the names of her friends), all of which is loaded as the program starts.

Following is a box diagram representing the different layers of the implemented SayaBot architecture vs. ALICE's:



3. Implementation

3.1 Changes that were made in Program D:

During a dialog between a human and SAYA, the sentence that the human says is translated from the sound into a string for SayaChatBot.

SAYA now needs to respond with an answer – to obtain a trustworthy string through a conversation API.

We changed the Program D so that it will all be wrapped in a new class named “SayaBot.java” that extends the required conversation API.

The conversation API:

- **public** SayaBot()
The constructor.
- **public** String getAnswer(String question)
This functions parameter is a string (what the human said) and the output is a string (Saya’s answer).
- **public void** closeBot()
This function closes the chatBot state and return Saya to her default status. Closing of the chatBot enables Saya to know who she talks to, and not store irrelevant information from the former person she talked with. (More on this is expended in Scenario chapter.)

3.2 Changes that were made in ALICE’s AIML files:

The main AIML file of ALICE (named Alice.aiml) had been changed to fit SAYA. All the AIML files are based on categories. Each category returns a response to one phrase. Adding a new category is based on the following platform:

- **<aiml>**: a tag that wraps an AIML document.
- **<category>**: a tag that marks a "unit of knowledge" in SAYA’s knowledge base.
- **<pattern>**: contains a simple pattern that matches a phrase that may be said to Saya.
- **<template>**: contains the response to be returned to the human.

An example for a category:

```
<category>  
<pattern>YOUR NEW NAME IS *</pattern>  
<template>No thanks I like the name SAYA.</template>  
</category>
```

3.3 Configuration file: startup.xml:

A global configuration file was added to enable easy personality related definitions for SayaBot administrators.

Currently, this file defines information like SAYA's favorite movie and actors, and other personality definitions that we require a simple changing schema without going into the core AIML code. These parameters can change every day, so this is a very friendly update interface.

An example for the information stored in the startup.xml:

```
<property name="name" value="SAYA"/>
<property name="master" value="COMPUTER SCIENCE DEPARTMENT, BEN GURION UNIVERSITY"/>
<property name="botmaster" value="Botmaster"/>
<property name="genus" value="robot"/>
<property name="location" value="Beer Sheva, Israel"/>
<property name="gender" value="Female"/>
<property name="species" value="chat robot"/>
<property name="size" value="128 MB"/>
<property name="birthday" value="January 30, 2007"/>
<property name="order" value="artificial intelligence"/>
<property name="party" value="Independent"/>
<property name="birthplace" value="Japan"/>
<property name="president" value="Shimon Peres"/>
```

Information about the current president changes every few years. Whenever a president is replaced, we go to the president line and change the value to the next president's name.

4. Scenarios

SAYA has the ability to remember things. For example: if a person tells her his name, then she will remember his name until he says otherwise or the conversation had ended. If a human talks to SAYA and ungracefully cease talking, then if SAYA doesn't receive new queries for a few minutes– the chatbot state returns to her default state by the controlling mainloop. The next time she talks to a human – she will no longer have the information about the former person (his name, favorite things, etc).

An example for a dialog between SAYA and a human:

SAYA: Hello!

HUMAN: hi there

SAYA: Hello JUDGE

HUMAN: my name is human

SAYA: Your name is human.

HUMAN: what is your name?

SAYA: My name is SAYA.

HUMAN: nice to meet you SAYA

SAYA: It's nice meeting you too human

HUMAN: tell me a joke SAYA

SAYA: It is better to be silent and be thought a fool, than to open your mouth and remove all doubt.

HUMAN: what is your favorite movie?

SAYA: My favorite movie is Hagiga basnuker. Have you seen it?

HUMAN: mine too

SAYA: Really yours too. That's unbelievable.

HUMAN: do you remember my name?

SAYA: Your name is human, seeker.

5. Conclusions and Future directions

We have created a package that enables Saya to conduct a full and maximally logic conversation with a human user, utilizing rich vocabulary of words and phrases and preserving a logic sequence of incoming information. Program D (the latest java open source incarnation) was wrapped into a single object, that can be manipulated from the mainloop of the robot. Wide knowledge stockpile written in AIML was included, providing a spacious range of information to be talked about with Saya. The core of SAYA's essence (personality, character, etc.) was stored in XML-written file, providing an easy access for the administrators. All in all, makes Saya a friendly robot, with a wide knowledge in different fields, that can conduct a conversation almost with any human being, talking to him about subjects as one wishes.

For future enrichment of SAYA's knowledge stockpile, the developer should write categories in the appropriate AIML form, placing it in the right knowledge file (Astronomy, Geography, etc.) or creating a new AIML file, and storing it in the "aiml" directory of the package. If changes to Saya personality required, they all should be made in the "startup.xml" file, stored in "conf" directory of the package. For new characteristics, new 'property' tags should be written in XML and added to the "startup.xml" file. Referring to the new characteristics in the AIML code is same as used in HTML.

6. Bibliography:

Sources:

ALICE web source: <http://www.alicebot.org/>

AIML files source: <http://www.alicebot.org/aiml/aaa/>

Program D source: http://aitools.org/Downloads#Program_D downloaded version: 4.1.5 zip source.

Program D documentation: <http://www.aitools.org/Documentation>

AIML overview: <http://www.pandorabots.com/pandora/pics/wallaceaimltutorial.html>