Game of Characters System Report

1.Motivation

* purpose of system, motivation for use

The purpose of our system is to help users who are interested in the famous TV series “Game of Throne” to learn about the background stories of each character and the relationships between them, when the users find themselves not so familiar with the whole story, want to get a gist of what’s going on, or just forget about what happened before.

With our system at hand, TV fans don’t need to take time to open the browser, type in the question “Who are Arya’s siblings”, or “Who’s Benjen Stark?” and browse the results one by one filtering out all the irrelevant contents. Instead, they only need to ask the computer out loud, saying “Who is Tywin Lannister to Tyrion Lannister?” and the simple and accurate answer, “Parent”, will be returned immediately.

It is not only fun to use, fast and accurate, but also the first dialog system implements on TV show storyline queries.

2. Capability

* scope of dialogue system and plans for future work

Currently, the system mainly answers for 4 kinds of questions:

(1) Backstory of a character X by user asking, “Who is Arya?”

(2) Or the relationship between 2 characters X and Y by user asking, “What’s the relationship between Ned Stark to Jon Snow?”

(3) Or questions about a character’s relative: “Who are Tyrion’s parents?”

(4) Or the name of the actor who portrayed a role, by user asking, “Who portrayed Benjen Stark?”

The system supports up to 1155 characters that currently appeared in the Game of Thorne series, users may get generic information about the characters’ age, status, which season they first appeared in by asking the name of them. Also, the relationship of each connected character like their children, spouse and parents can also be obtained by querying the system.

As the system contains more detailed profiles of each character, like the religion and origin, than just their names and relationships, it could be further developed to add more types of queries, like things the characters were doing at some location at certain period of time, or the particular feelings one character holds toward another.

In the future, our system could be expanded to cover more than one TV series, like the Arrow, Breaking Bad, etc. And, for those shows that are not ended yet, we will be updating the system to add newly appeared characters and stories to ensure the database is up-to-date.

3. Technical Used

·description of technical details (ASR, programming languages, platforms used, etc)

The system is using Alexa Skills Kit, it is Amazon's intelligent voice recognition and natural language understanding service. The major components of the dialogue system, like automatic speech recognition (ASR) and natural language understanding (NLU), are handled by Alexa in the cloud. It also recognizes and responds to user’s voice requests instantly, greatly improve the quality of the user experience.

We built the system for an Alexa skill using AWS Lambda, which runs the code only when it’s needed and scales automatically, so there is no need to continuously run servers. We used Python to collect data from a set of data links, like extracting each character’s information, and we also built a linked list for the relationship tree among all the characters. To apply the code into Alexa, we uploaded the code into Lambda function, which executes the code and responses to the Alexa’s voice interactions, it will also automatically manage the compute resources.

4. Context For Use

·description of context for use (mobile, web, desktop)

As some of dialogue system components like Automatic Speech Recognition and Natural Language Understanding are handled in the cloud, the system can be implemented in any Alexa-enabled device that has a microphone and speaker, as long as it connects to the Internet and the Alexa hardware. It can be used in a laptop, a smart phone, or even a car.

Currently we built and execute our system on the laptop.

5. Dialogue System Architecture

Our system is using frame-based, mixed initiative dialogue manager, where the NLU and the ASR modules are handled by Alexa in the cloud. Users can ask questions about any character’s parents, siblings, spouses and grandparents, or any general information of a character stored in the database. The diagram of dialogue system architecture is shown as below.

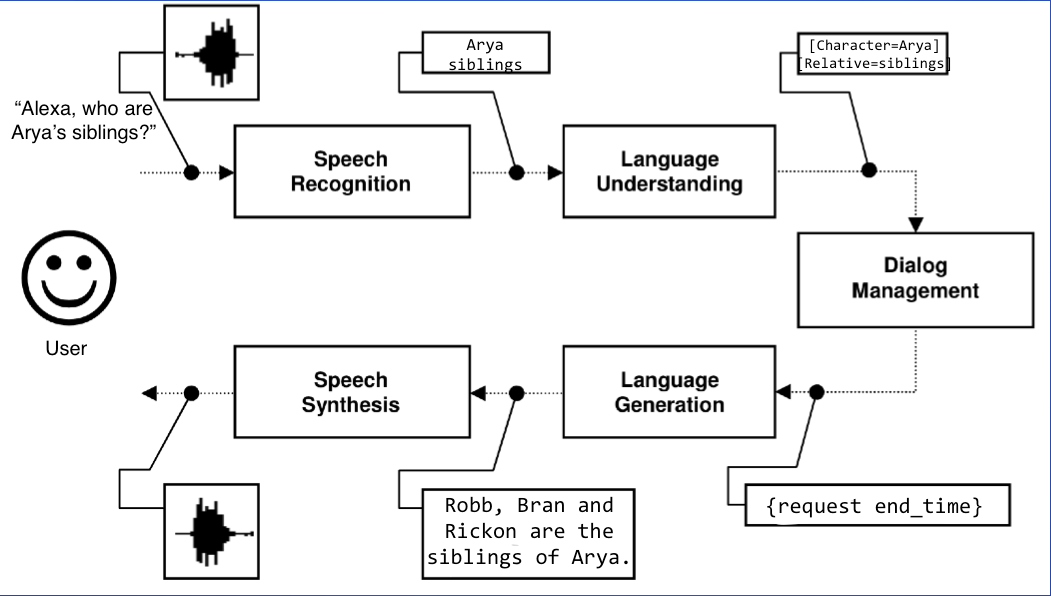


Figure 1. Diagram of dialogue system architecture

6. Dialogue Manager

* description of dialogue manager including reasoning for that approach (ex finite-state, frame-based, statistical, ...)

In the Alexa Skills toolkit, we are using frame-based approach for the dialogue manager, the reasons for doing this is that frame-based DM has higher ASR results, it monitors the dialogue while filling in each slot, and refers to that slot for information. Also, as frame-based DM has no absolute sequences for which slot to fill first, users may ask questions in whatever order they like, or contain multiple concepts (names, relation) in one turn.

The dialogue manager decides what to do next after receiving and interpreting the users input, it will ground inputs like names of the characters and output the information of the stories and character relationships that has been queried.

First we specified a mapping schema between users’ spoken input and the intents, which represent actions that fulfills a user’s spoken request. There are currently four different intents in our system: ‘WhoIsX’, ‘CharactersRelative’, ‘RelationOfXandY’ and ‘WhoPortrayedX’.

For each intent there are: (1) Intent property that gives the name of the intent, like the ‘CharactersRelative’. (2) And the slots property that lists the slots associated with that intent, like the ‘Character’ slot and the ‘Relation’ slot. When a user says “Alexa, who are Tyrion’s parents?”, and the Alexa service sends the ‘CharactersRelative’ intent with the value ‘Tyrion’ in the ‘Character’ slot and ‘parent’ in the ‘Relation’ slot in written format. On receiving the ‘CharactersRelative’ intent, the system can then look up the list of characters for the word ‘Tyrion’ and the list of relations for the word ‘parents’, then sends back text to convert to speech.

There are also different types defined in each slot. For instance, the ‘Character’ slot is defined as type ‘LIST\_OF\_CHARACTERS’, in order for the system referring to a list of character name values provided for the slot (Tyrion, Arya, etc.), and the slot type can be used for multiple slots, like the type ‘LIST\_OF\_CHARACTERS’ can also be used in ‘WhoIsX’, ‘RelationOfXandY’ or ‘WhoPortrayedX’ slots.

The syntax of the DM that contains the intent and slots are shown as below:

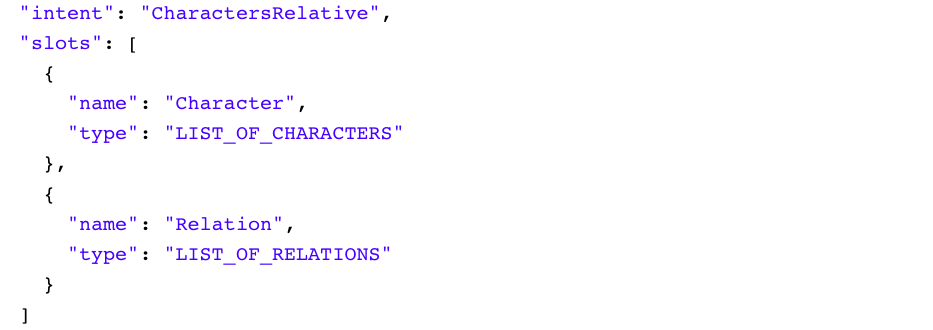


Figure 1. Intent Schema of the Dialogue Manager

7. Persona Design

As our system only intended for basic information queries, the persona of the system only contains the personality (like the language, tone, speed of the Alexa) and backstory of the system, with no physical manifestation in design.

Our system implicitly contains persona backstory of Alexa, that she is a female from America, and she speaks American English in medium speed with a pleasant intonation, which indicates she is a very helpful ‘person’, who are delightful to help, make the user feel like willing to talk to her.

8. Round Robin Feedback

* comments/feedback from waypoint round robin and iterations/changes made to address them

9. Sample Dialogue History

* sample dialogue history, accompanying UI

10. User Satisfaction Analysis

* user experience survey with results and analysis of what features of the system are indicative of user satisfaction

2 main predictors: Task completion, ASR results/conceptual accuracy

11. System Limitations

* challenges faced, system limitations

The key challenge of the system is its low recognition result in noisy backgrounds, as its shown in the round robin survey class, when the system constantly returns restart utterance due to speech recognition time out. Sometimes when the noise is added in, the ASR results could be lower than usual that the system returns wrong answers like information of the character other than what the user had asked for.

Other challenges like context sensitivity, tapered prompts of the system, and the ability to respond when the user barge-in in the middle of system speaking are also complex issues that we are currently facing.

12. Contribution Breakdown

Project proposal, Implementation and modification

Theoretical, evaluation and report

**Presentation:**

All of the above, but need not be explicit slides for each one. E.g.: you could cover both "diagram of dialogue system architecture" and "description of dialogue manager including reasoning for that approach" at the same time, with one *slide that shows* a diagram of the dialogue system, over which you *talk about* your dialogue manager and reasoning.