Contextual Flow In Chatbot Conversations

Arijit Datta

Y5111

Guide: Prof Amitabha Mukherjee

CS365, Spring 2008

Introduction

As part of this project, I have done a study of the ALICE model of chatbot and Bayesian Conversation Architecture (Quartet Architecture) developed by Microsoft Research. I hereby present a proposal of a chatbot model. The underlying strategy is weaved over ideas picked up from the ALICE and MSR Group, and also tries to use the internet and search engines.

Chatbots as agents in Human Technology Interaction

Human-Technology interaction is at present a field with good momentum and prospects. Research in this field over the past several years have made various forms of interactions possible between man and machine. One such interface would be through conversations. Conversation agents or simply chatbots are computer programs that engage with the human in turn-by-turn conversations in natural language. However, the 'naturalness' of such conversations is far from satisfactory.

Stupid Chatbots

Chatbots are restricted to the knowledge that is manually handcrafted into their knowldedge base by their authors. The knowledge bases are thus time-consuming and tedious to build up and are quickly exhausted. The chatbot currently with the largest knowledge database would be ALICE where about 50000 patterns have been handcoded and more are being added. However, this paradigm can not have long term potential. An automatic process has to be devised and a non-static database has to be implemented. This will improve the naturalness of the chats. Observe, however, that if we compare a chatbot to a typical person, it is ‘natural’ for the chatbot to know only a few interesting areas and be aware of the most popular topics only.

Approaches to chatbots

The pattern-matching technnique was introduced by MIT labs way back in 1955 when they designed ELIZA. This approach still remains to be the most popular approach till date. The method is very simple and gives the most natural feel to the chatbots. There is minimum error in conversation. However, the technique also has limitations. Contextual or semantic nature can not be imparted very well to the conversations.

Markov Models were implemented in MegaHal. This chatbots was able to come up some really novel replies to user inputs. The chatbot won a lot of praise from the judges in Loebner Contest and affirmed that a machine learning approach was worthwhile. However, the chatbot was not producing grammatically correct sentences.

Using the free labour of millions of users avaialble on the net seems to be a greedy approach. However, there is the problem of screening out garbage-givers.

NLP is still in infancy and is of little help in the field of chatbots.

Bayesian Networks seems to be the most promising approach to take chatbots to the next level. Bayesian Networks in chatbots is heavily researched by Microsoft Research Team, Redmond. MSR team is involved in building intelligent interface agents. A Quartet Architecture has been designed that provides a natural language chat interface to users over various domains. The conversation architecture is pretty sophisticated, incorporating cutting edge technology from NLP, Artificial Vision, Voice Analysis; all weaved together elegantly into a Bayesian model of conversation.

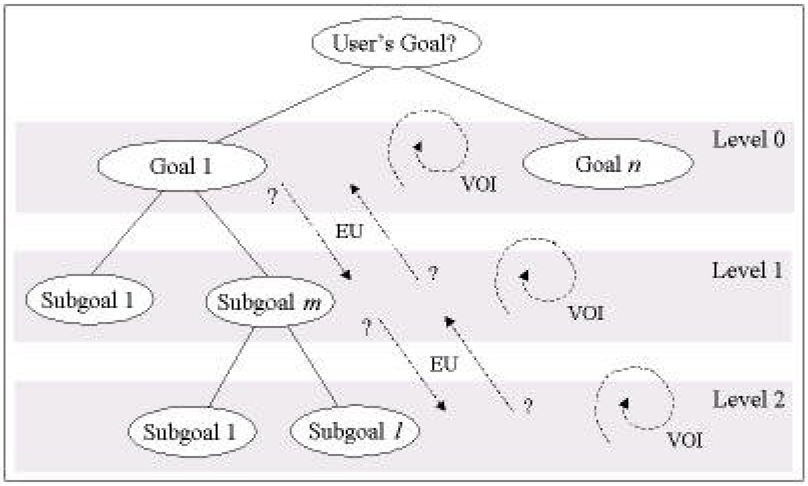

Figure 1. Schematized view of an overall model for guiding diagnosis of goals via Bayesian inference, computation of value of information (VOI), and expected utility analysis (EU) for navigation between levels of detail for progression and backtracking in conversation.

Ref: A computational Architecture for conversation, MSR, E Horvitz, T Paek

Some Guidelines

Maintaining the general nature of chatbot is very desirable. Also care should be taken to cause minimum pollution in the chatbot knowledge base. Simplicity is the single most important criteria when developing a new model. It is important that we dont get lost in the complexity of contextual and semantic richness of natural human conversations.

The ALICE Model

AIML (Artificial Intelligence Markup Language) is an XML-compliant language that's easy to learn. AIML, describes a class of data objects called AIML objects and partially describes the behavior of computer programs that process them. AIML objects are made up of units called topics and categories, which contain either parsed or unparsed data. The ALICE model of the chatbot consists of an AIML brain and an AIML interpreter. The AIML brain is a collection of all the AIML files and contains the entire knowledge base of the bot. The AIML interpreter is a programs that provides the user an interface, takes an input, matches through the AIML files for an output and gives the user back the output. The interpreter maybe loaded with features through which the user can select a particular bot personality or teach the bot a new answer etc.

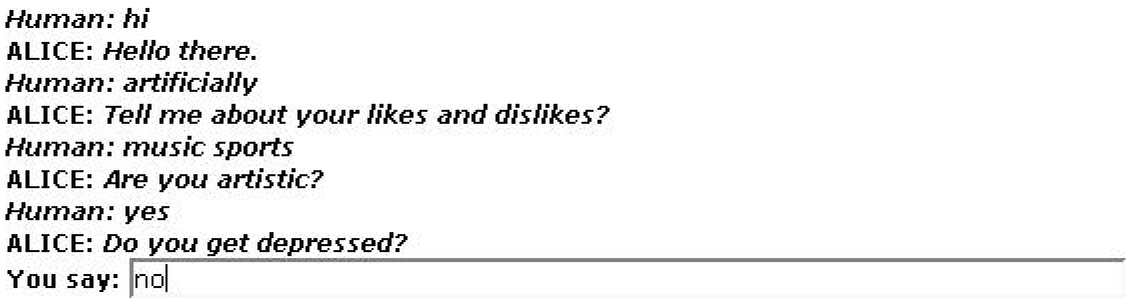

Fig: A piece of chat with ALICE

Extending the Chatbots

The first step to adding a contextual nature to the conversations is to ground the conversation over a topic. Identifying the “topic” from the user input then becomes the first question. Having a non-static database of “topics” is desirable. When we target a non-static database of topics, we must find a way to automatically find “responses” for the new “topics”.

Potential of the web

Human conversation ranges over all possible topics. A general purpose chatbot should be able to talk on topics as diverse as music, politics, technology, philosophy and what not. Obtaining training corpora for all domains/topics of conversation is practically possible.

The Web is a seemingly infinite source of data. Everything that goes on stays in the web. We have smart search engines. Every problem can at least theoritically be reduced to a search problem. Research by Search Engine giants have shown that how the information is organized holds the key to the utility value of the information.

Previous Proposals

Jizhou Huang1, Ming Zhou, Dan Yang discussed their idea of extracting chatbot knowledge from forums. forums are a place of online conversation and hence followed the expectation to mine conversation data over specific topics. However as they showed in their paper, forums have too high a level of noise. And finding good sentences itself becomes a tricky thing.

FAQs were proposed because they did not have the noise. FAQs are written in a specific easy-to-follow neat and clean format. But then FAQs are not always avialable for all topics.

Wikipedia

Wikipedia, the free encyclopedia. Wikipeida has become a household term in the last few years. Wikipedia is probably the richest and densest source of information on the web. It is rich in content. The information is well organized into articles. There is an wikipedia page for almost every imaginable topic.

From the perspective of chatbots, Wikipedia holds the potential to surpass the limitations of both forums and FAQs and yet combine the promises of the two. There is minimum pollution or noise in Wikipedia. A very high percentage of the sentences are "good". Article are equivalent to topics. The pages come in a definite format. And mining for good sentences should be relatively easy.

Fixing topics

The user input can be almost anything. Every possible user input has to be indentified with a topic. One possible solution is to do a keyword search. The user input is first processed to weed out all common high-frequency words.

i | you | he | she | they | we | them | everybody | somebody | etc etc

The list will include all pronouns, articles, conjunctions etc. The topic will typically be conveyed by the noun phrase of the user input. The least frequent words in English are the ones with highest information about the context of the conversation.

Google Search within wikipedia is an interesting trick. Forming the search query is trivial because Google uses GET method. If "input1 input2" is the user input, the search query is as follows

http://www.google.co.in/search?hl=en&q=input1+input12+site%3Aen.wikipedia.org&btnG=Search&meta=

The query returns a Google results page with a known format. Mining the URLs from the html corpus is again easy. The first URL returned is simply taken as the result. The URL itself can be processed to extract a phrase for the topic. The string that follows the last "/" character is cut out and all special characters are replaced by spacebars. We thus have a topic name.

Finding Responses

We thus have a way to map every user input into an Wikipedia article. The Wikipedia article now has to be processed for possible responses.

One simple and naive solution would be to pick up arbitrary sentences from the page. All sentences in an Wikipedia article are grammatically legible. The sentences are informative, concise with lot of information. We could exploit this desirable feature of content richness of Wikipedia pages to avoid any complications.

However, any arbitrary sentences from the page may not make sense in absence of the context of the article. How good the naive approach might have worked remains to be seen experimentally.

An alternative is to process the article with an automatic text summarization software. The summarization software can then pick out a handful of the good informative sentences. www.extractorlive.com is a satisfactory suggestion bacause of two reasons. Firstly, it produces a good enough summarization output. Secondly, it picks up important "keyphrases" from the page which then directly serves as the list of topics related to the current topic.

Both approaches should be tested with to see which works better.

Enhancing the responses

If the chatbot is always mining sentences from the Wikipedia page, over time it will tend to come off as an all knowing quizmaster at the risk of losing the natural touch to the conversation. To maintain the balance towards the naturalness, we make a simple observation.

There are general responses that can be blindly added to every topic. So if the topic is X, we can add these responses blindly to the database for X. For eg,

Ahh, X. I have heard of it.

What do you know about X?

Do you have interests in X?

I do know some facts about X.

Generating the AIML file

Once we have a pile of responses for a particular topic, we can easily automate the process a simple AIML and add it to the chatbot database. Writing the responses into AIML format is now only a matter of adding simple tags at the right places.

```
<category>
<pattern>X</pattern>
<template><random>
    <li> (responses) </li>
</random></template>
```

</category>

This AIML file is then added into the database.

```
<!-- Last modified July 26, 2007 -->
– <category>
    <pattern>PROTESTANT *</pattern>
  – <template>
    – <random>
      – <li>
          There are significant differences between Protestant sects.
        </li>
        <li>There are Calvinists, and there are Lutherans.</li>
        <li>We musn't forget the Catholic counter-reformation.</li>
      </random>
    – <think>
      – <set name="it">
          <set name="topic"> Protestants </set>
        </set>
      </think>
    </template>
  </category>
– <category>
    <pattern>ARE YOU A PROTESTANT *</pattern>
    <template>Yes, that is my religion.</template>
  </category>
– <category>
    <pattern>WHERE IS THE _ BIBLE</pattern>
    <template>Try reading the twenty third Psalm.</template>
  </category>
– <category>
    <pattern>* RELIGION</pattern>
```

Fig: A sample AIML file religion.aiml

Relating Topics

As the conversations go on, the chatbot obtains and 'learns' more topics. It is necessary to impart a structure and organization to these topics. A Bayesian Netwok over the topics is tempting at this point.

We try to store the topics in a graphical ddata structure. Each topic corresponds to a node in the graph. Related nodes are joined by an weighted edge. The weight between two nodes can then reflect how strongly the two nodes are connected.

Assigning the prioris

Assigning weights to the graph poses a question.

A way of assigning probablities to the arcs from one topic to another would be from the link analysis of wikipedia pages. For example, consider Maths and Groups. The topics are somewhat related and a conversation on one topic can be directed to the other. Let M be the total number of user clicks from the Maths page to any other page within Wikipedia domain. Let mg be the total number of clicks from the Maths page into the Groups page. Consider the ratio (mg+gm)/(M+G). This fraction can be used to assign probabilites to the network. The numbers mg, gm, M and G are all a property of the page and the ration can be calculated in fixed time.

Search Engines actually do keep track of all these data. These data are crucial to search engine algorithms.

We can put a threshold value on this probabilty, and in case the value is less than the preassigned, no arc is assigned.

There a lot of ups and downs that come along with this (crude?) model of assigning weights.

Two topics stand the chance of being ever related if only there is a direct link from one page to the other. Looking at a general article, we can see a large number of links to other Wikipedia article and it seems viable that this shortcoming will not come in the way.

How good is the ratio a reflection of the true nature of the contextual flow of a conversation? This is something that we can comment on only after the chatbot is developed, tested with a sufficiently large number of users and the feedback collected.

However, the data is not publicly available.

We can simply use google search feature. The idea is to query Google with "Maths", "Groups" and "Maths+Groups" and use the fraction to assign weightage to the graph. This would however be severely off and crude because the Google numbers are a very bad reflection of what the ratios should actually be.

We consider a third alternative. Since we relate only topics that are returned by the summarization software, we may assume all such topics to be strongly related and assign a constant ratio to the weights. We can then keep updating the ratios everytime conversation flows over the topics and thus over time, collaterally, develop a good reflection of how the user moves from one topic to another.

The bot can now be able to guide the conversation from one topic to another. Conversation can then be navigated from one topic to another over the graph and we will have certain "smoothness". Moving from one topic to a different is then a matter of finding possible paths from the destination topic to the final topic and then gradually "pushing" the context from one topic to other, preferably staying on some topics for a certain period of time.

The ALICE brain into a graph

The ALICE brain is freely available under the GNU GPL licence and initially contains 58 AIML files. These 58 files can easily be mapped manually to 58 nodes in the graph. This part of the graph then forms the "core" brain of the chatbot and is static (we do not delete nodes from this part). So we dont start with an empty graph either. Prioris can be assigned manually or the same general technique can be uses.

A good idea would be to transform the ALICE brain into a bayesian graph and see how well the contextual flow of the conversation can be achieved before taking the strategy any further.

We therefore develop a chatbot that grounds conversation over the Wikipedia articles and can be thought of having an extended “Wikipedia” brain over the original ALICE brain. In addition the brain is imparted a Bayesian kind of structure.

Limiting the graph and the database

Wikipedia provides over a million articles. Theoritically, the graph grow into millions of nodes over time. Such a large knowledge base could turn out to be more than is desirable. We may have to restrict the graph to a simple upper bound of nodes. Simple speaking, we maintain a score for every node. This score is a reflection of how often the conversation was grounded over this node. We give the nodes a +1 score everytime conversation reaches that node. When the upper bound of the no of nodes has been reached and a new node needs to be added, we simple truncate and scrap away a fixed number of nodes from the graph that had the lowest number of scores.

Viewing the chatbot as an information retrieval system

The model suggested above can be alternatively viewed a sort of information retrieval system over the internet from wikipedia. This is like an utility value of the chatbot. When an user talks with the chatbot long enough over a specific domain, he can obtain a good bit of information of what wikipedia has to say. All he has to do is keep the chatbot talking.

An example

The user input: "i like linkin park"

Weeding out common words: "linkin park"

URL Query: "http://www.google.co.in/search?hl=en&q=linkin+park+site%3Aen.wikipedia.org&btnG=Search&meta="

Wikipedia URL: "http://en.wikipedia.org/wiki/Linkin_Park"

Topic Name: "Linkin Park" (New Node to be added to the graph)

Result of summarization:

Keyphrases: album, band, awards, rock, artists, studio album

Highlights:

“Alternative rock”

“Linkin Park is a rock band from Agoura Hills, California.”

“Since their formation in 1996, the band has sold more than fifty million albums and won two Grammy Awards”

“They achieved mainstream success with their debut album, Hybrid Theory, which was certified Diamond by the RIAA in 2005.”

“Their following studio album, Meteora, continued the band's success, topping the Billboard 200 album charts in 2003, and was followed by extensive touring and charity work around the world.”

“Awards and nominations”

“During this time, Linkin Park received many invitations to perform on many high-profile tours and concerts including Ozzfest, Family Values Tour and KROQ Almost Acoustic Christmas.”

“The band also formed their own tour, Projekt Revolution, which featured other notable artists such as, Cypress Hill, Adema, and Snoop Dogg .”

“Within a yearâ€™s stretch, Linkin Park had performed at over 320 concerts.”

“The experiences and performances of the precocious band were documented in their first DVD, Frat Party at the Pankake Festival, which debuted in November 2001”

Total: 9 responses

As is observed from the example, the summarization software is not always doing an excellent job. The data returned is split at every period (.) into individual responses.

Conclusion

This is a general strategy for extending the chatbots that should be interesting and worthwhile to implement and experiment with.

Bibliography

1. Predicting Evidence of Understanding by Monitoring User's Task Manipulation in Multimodal Conversations Yukiko I. Nakano Yoshiko Arimoto Kazuyoshi Murata Yasuhiro Asa Mika Enomoto Hirohiko Sagawa

2. A Computational Architecture for Conversation Eric Horvitz (MSR) Tim Paek (Stanford)

3.Continuous Listening for Unconstrained Spoken Dialog Eric Horvitz (MSR) Tim Paek (Stanford) and Eric Ringger (MSR)

4.Principles of Mixed-Initiative User Interfaces Eric Horvitz (MSR)

5.Uncertainty, Utility, and Misunderstanding: A Decision-Theoretic Perspective on Grounding in Conversational Systems Eric Horvitz (MSR) Tim Paek (Stanford)

6.DeepListener: Harnessing Expected Utility to Guide Clarification Dialog in Spoken Language Systems Eric Horvitz (MSR) Tim Paek (Stanford)

7.Harnessing Models of Users' Goals to Mediate Clarification Dialog in Spoken Language Systems Eric Horvitz (MSR) Tim Paek (Stanford)

8.Inferring Informational Goals from Free-Text Queries:A Bayesian Approach David Heckerman and Eric Horvitz

9.Grounding Criterion: Toward a Formal Theory of Grounding Eric Horvitz (MSR) Tim Paek (Stanford) 2000

10.Conversation as Action Under Uncertainty Eric Horvitz (MSR) Tim Paek (Stanford)

11.Empirical Methods for Evaluating Dialog Systems Tim Paek (Stanford)

12.A Corpus Based Approach to Generalising a Chatbot System Bayan A Shawar PhD Thesis (2002) University of Leeds UK

13.The Point Process Paradox: Where Should We Extend the Conversation? Nozer D. Singpurwalla 2000

14.Using dialogue corpora to train a chatbot B A Shawar Eric Atwell

15.Uncertain reasoning and forecasting P. Dagum ab' , , A. Galper a, E. Horvitz c, A. Seiver Stanford

16.A Chat-Bot based Multimodal Virtual Guide for Cultural Heritage Tours A. Santangelo1, A. Augello1, A. Gentile G. Pilato2, S. Gaglio

17.Common Sense Conversations: Understanding Casual Conversation using a Common Sense Database Nathan Eagle, Push Singh, Alex (Sandy) Pentland MIT Labs

18.A two-layered approach to communicative artifacts 2007 Yong Xu Tatsuya Hiramatsu Kateryna Tarasenko Toyoaki Nishida Yoshiyasu Ogasawara Takashi Tajima Makoto Hatakeyama Masashi Okamoto Yukiko I. Nakano

19.The Trouble with Chatbots: social skills in a social world , Peter Wallis Emma Norling

20.Tracking Conversational Context for Machine Mediation of Human Discourse 2000 Tony Jebara, Yuri Ivanov, Ali Rahimi, Alex Pentland MIT Labs

21.A Two-Stage Bayesian Network for Effective Development of Conversational Agent Jin-Hyuk Hong and Sung-Bae Cho Yonsei Universoty

22.Extracting Chatbot Knowledge from Online Discussion Forums* Jizhou Huang1, Ming Zhou2, Dan Yang1

23.A chatbot system as a tool to animate a corpus B A Shawar Eric Atwell

24. An interdisciplinary review of turing test AYSE PINAR SAYGIN1, ILYAS CICEKLI2 & VAROL AKMAN