PLACAT: A user-friendly question answering system for smart speaker devices

Gabriel Luthier & Andrei Popescu-Belis
IICT / HEIG-VD / HES-SO
Yverdon-les-Bains, Vaud, Switzerland

19 June 2019
Chatterbots vs. smart assistants

• *Chat[ter]bots*
  - good interaction, well-formed replies | no useful task or knowledge

• *Smart assistants*
  - task-oriented [search / buy / navigate/ call] | fail when off-task

• Can we combine the advantages of both approaches?
  PLACAT project: a platform for designing conversational agents for accessing text-based knowledge (HES-SO P2)
Chatterbot sample from www.cleverbot.com

- Why do people build chatterbots?
  - Pass the Turing test
  - Win the Loebner prize
  - Simulate a therapist
  - Provide a conversation partner, e.g. for language learning

Cleverbot: Hi. What is your name?
User: Bob.
Cleverbot: Hello bob I'm Sal.
User: Hi Sal, do you like Star Wars?
Cleverbot: J love it.
User: What is your favorite character?
Cleverbot: I don't really have one.
User: Who are Ray's parents?
Cleverbot: Why do you want to know huh?
User: I cannot wait for the last episode.
Cleverbot: Remind me what it's called again.
User: The 8th one is The Last Jedi.
Cleverbot: Harry Potter and the deathly hallows.
User: No no no.
Cleverbot: Then what are you trying to tell me?
User: About Star Wars.
Cleverbot: Cool. I haven't seen rogue one yet.
Assessment of this chatterbot

😊 Positive first impressions
😊 Replies in good English, quite related to the conversation
😊 Some knowledge of the Star Wars domain

😞 No in-depth knowledge
😞 Canned replies for difficult situations, not related to the conversation
Smart assistants

• On smartphones or smart speakers
  • Google: Home + Assistant
  • Amazon: Echo + Alexa
  • Apple: HomePod + Siri
  • Microsoft: Echo + Cortana

➤ Tailored to user-specific tasks related to their makers’ main business: find, sell, talk
Assessment of a smart assistant: Google Assistant on Google Home

😊 Great speech recognition capacities
😊 Ability to perform common tasks: weather forecast, play music, find restaurant
😊 Pre-defined answers to typical chit-chat

😊 Reliance on the Google search engine
   😊 exact answers to some requests for information (from knowledge graph)
   😊 excerpts of web pages in other cases
   😊 web pages are often not natural answers to questions

😊 No more than one turn for small talk
😊 “Continuous conversation”: limited memory for several turns
Task: question answering over documents

Controller / dispatcher: dialogue act recognition

Interaction: chatterbot

Knowledge

User
Main idea: combine chatterbots and knowledge-based question answering

- Platform for designing conversational agents
  1. **user-friendly**: start/end, politeness, small talk
  2. **task-enabled**: here, question answering on Wikipedia
  3. integration with **smart speakers** (Google Home initially)
  4. **third-party + own code**
Voice-based front-end: smart speaker

- Google Home smart speaker allows user-created actions*

- PLACAT action created on https://console.actions.google.com/

- Action connects to DialogFlow*
  - forwards the user’s *transcribed* utterance to our server
  - receives the reply from our server
  - sends the words to the speaker for *speech synthesis*

* So does Amazon Echo with 'skills'

* DialogFlow (was API.AI, similar to WIT.AI) enables the creation of simple dialogue models and the use of NLP tools to detect intentions
Controller: utterance dispatcher

• Not a good idea to use knowledge to answer questions like *How are you?* or *Are you a robot?*

➢ Dispatch **interaction management** utterances (chit-chat, greetings) to a chatbot, and **task-related** utterances (informative questions) to a QA system

• Trained a dialogue act recognizer on labeled data from the chatbot and from the QA system
  • currently: binary decision (but could be refined)
  • Logistic Regression classifier trained on 25k utterances: 50% chitchat and 50% questions → 90% accuracy
Chatbot

• Deal with chit-chat, politeness, greetings, & off-task questions
  • relevance of replies may vary, but they tend to be socially acceptable

• Sequence-to-sequence neural model (Vinyals and Le, 2015)
  • RNN with LSTM implemented in PyTorch
    https://pytorch.org/tutorials/beginner/chatbot_tutorial.html
  • trained over 500M words from a movie dialogue corpus (OpenSubtitles EN)
Question Answering system

• Task = information access
  • now: information from English Wikipedia
  • can be adapted to a given field / company

• Components of our system
  • ElasticSearch: find relevant paragraphs from Wikipedia based on the question as a query
  • BERT: extract exact answers from paragraphs
    • Bidirectional Encoder Representations from Transformers (Devlin et al., 2018), BERT-base https://github.com/google-research/bert
    • trained on SQuAD 2.0 (Rajpurkar et al., 2018) https://rajpurkar.github.io/SQuAD-explorer/
      • 100k questions based on 500 WP paragraphs + 50k unanswerable questions

• Challenge: will ElasticSearch find the right paragraph for BERT?

• Design options that we optimized
  • question: filter out stopwords or keep all words
  • ElasticSearch: on full text, on opening, on titles
  • search results: keep 1-best or 5-best
  • answer extraction with BERT: from the opening text of the page, or from the entire page
Evaluation of the QA system

• SQuAD 2.0 questions were made to evaluate answer extraction
  • given a paragraph, workers have targeted more or less central facts
  • 1 question out of 3 is unanswerable (i.e. the correct answer is “I don’t know”)
  • best scores on hidden test set : 87% | our BERT-base scores on devset: 72.5%

• For PLACAT, we evaluate paragraph retrieval + answer extraction → 48%
  😊 correct on most unanswerable questions
  😞 incorrect on many answerable ones, often because ElasticSearch fails to find the right paragraph: when a question targets a minor fact, the right paragraph is hard to retrieve

➢ later: retrieve many paragraphs + merge answers

<table>
<thead>
<tr>
<th></th>
<th>Search on titles</th>
<th>Search on openings</th>
<th>Search on full texts</th>
</tr>
</thead>
<tbody>
<tr>
<td>BERT on openings</td>
<td>1st a. 5 a.</td>
<td>47.8</td>
<td>46.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>42.5</td>
<td>38.6</td>
</tr>
<tr>
<td>BERT on full texts</td>
<td>1st a. 5 a.</td>
<td>45.1</td>
<td>41.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>39.1</td>
<td>30.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>27.9</td>
</tr>
</tbody>
</table>
Perspectives [followed by a demo]

• Our intentions for the PLACAT platform
  • user-friendly chatbots giving access to corporate information
    • often available as large document sets and/or websites
    • evaluation methods tailored to domain and documents

• Alternative use of PLACAT
  • design a chatterbot with rich general knowledge
    • Amazon Alexa Prize: long general-purpose conversations
Demo: using PLACAT on Google Home

- Talk to Agent Placat
- Alright. Getting the test version of Agent Placat. Welcome! Ask me anything!
- What is bitcoin?
  - It is a decentralized digital currency without a central bank or single administrator
- What is a digital currency?
  - A money balance recorded electronically on a stored-value card or other devices
- Who is the creator of bitcoin?
  - Satoshi Nakamoto
- Thank you, bye
- Goodbye, come back any time!