Global Bots and User Assistance

Felix Sasaki, LambdaWerk GmbH (formerly W3C and DFKI), Berlin
Christian Lieske, SAP SE, St. Leon Rot

The full text for the conference proceedings has been included in the appendix to this presentation.
The bots we will be talking about ...

“Service” with a chat interface

Integrated into a messaging platform

Possibly, with voice input and output
About Us

Felix Sasaki – LamdaWerk GmbH
- Long term contributor to applications of web technologies for representation and processing of multilingual data
- Previously: Senior Researcher at DFKI (LT-Lab) and member of W3C, including W3C Office Germany / Austria
- Studied Japanese, linguistics, web technologies in Germany and Japan
- Details: http://sasakiatcf.com/felix/

Christian Lieske – SAP SE
- Knowledge Architect (SAP Language Services, Globalization Services)
- Topics: Multilingual production, natural language processing, internationalisation
- Contribution to World Wide Web Consortium (W3C), OASIS, Unicode Consortium
- Studied computer science and natural language processing (including machine translation and speech processing)
Overview

Implications for TechDoc
• What impact?

Focus on Language
• What are sample challenges?

Quiz
• What’s a chatbot?

Hints
• What to consider or watch out for?
Implications for TechDoc/How Chatbots and User Assistance connect

Solution/service quality and brand perception may get more heavily tied to language/content

User assistance/answers via chatbot vs. access via traditional user interface

- Value engineering (e.g. from information design models); discovery, curation, and amendment of content
- Writing (and meta data creation) at very fine-granular level
- Less focus on visual presentation

Conversational design

- People-centric approach
- Empathy
Quiz 1 – What sets chatbots apart?

Program
- explicitly started

Bot
- always on
- observing
- acting on behalf
- communicative

Chatbot
- conversational experience

Personal Assistant
- highly contextualized and adaptive
User: What is the translation of bank?

Bot: Sorry to ask back but: What is the target language you are interested in?

User: German

Bot: Cool. I save German as your default target language. Enter "Forget my current target language" in case this becomes inconvenient.

Bot: You know what? Translation alternatives from various domains exist. Are you interested in a particular domain?

User: Financials

Bot: Cool. I save Financials as your default domain. Enter "Forget my current domain" in case this becomes inconvenient.

Bot: In the domain of Financials, English bank translates to Bank in German.
Quiz 2 – Fineprint

- Accept anything
- Confirm and clarify
- Be user-centric
- Take turns
- Remember
## Quiz 3 – What’s needed?

### Content

**Intent recognition**
- “I want to make a reservation at a pizza place”: reserve a restaurant
- “I want to know the population of Potsdam”: satisfy a knowledge need

**Entity detection/identification**
- “I want to stay in Berlin”: entity of type *city*

**Language/response/answer generation**
- “I will check for Italian restaurants”
- “I found a nice hotel in the centre of Berlin”
Quiz 3 – Fineprint (1/4)

Stuttgart ... The capital of Baden-Württemberg ... Area code 0712 ... Nice sights include the Venusberg, ...

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stuttgart</td>
<td>is_capital_of</td>
<td>Baden-Württemberg</td>
</tr>
<tr>
<td>Stuttgart</td>
<td>has_area_code</td>
<td>0711</td>
</tr>
<tr>
<td>Bonn</td>
<td>has_sight</td>
<td>Venusberg</td>
</tr>
</tbody>
</table>

The population of Stuttgart is 623738.
Quiz 3 – Fineprint (2/4)

https://www.wikidata.org/wiki/Q161071

screwdriver (Q161071)

hand-tool

<table>
<thead>
<tr>
<th>Language</th>
<th>Label</th>
<th>Description</th>
<th>Also known as</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>screwdriver</td>
<td>hand-tool</td>
<td></td>
</tr>
<tr>
<td>German</td>
<td>Schraubenzieher</td>
<td>Handwerkzeug</td>
<td>Kreuzschlitzschraubenzieher</td>
</tr>
<tr>
<td>French</td>
<td>Tournevis</td>
<td>outil à main</td>
<td>1 tournevis détecteur de phase</td>
</tr>
<tr>
<td>Bavarian</td>
<td>No label defined</td>
<td>No description defined</td>
<td></td>
</tr>
</tbody>
</table>

All entered languages

Statements

instance of

- 0 references
  + add reference
  + add value

es | Destornillador
Quiz 3 – Fineprint (3/4)

Messaging platform (habitat)

[Speech-to-text/speech recognition]

Natural Language Understanding

Intent recognition

Entity detection/identification

Dialogue Management

Business Application

Natural Language/Response/Answer Generation

[Text-to-speech/speech synthesis]

Model(s)

- Hand-crafted
- Learned from data

Model training

- Hand-crafted
- Learned from data
## Quiz 3 – Fineprint (4/4)

<table>
<thead>
<tr>
<th>Natural Language Understanding</th>
<th>Dialogflow (formerly API.AI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intent recognition</td>
<td></td>
</tr>
<tr>
<td>Entity detection/identification</td>
<td></td>
</tr>
<tr>
<td>Dialogue Management</td>
<td>Dialogflow (formerly API.AI)</td>
</tr>
<tr>
<td>Business Application</td>
<td>Information Retrieval (node.js, knowledge source Wikidata/DBpedia)</td>
</tr>
<tr>
<td>Natural Language/Response/Answer Generation</td>
<td>Dialogflow (formerly API.AI)</td>
</tr>
</tbody>
</table>

- Messaging platform (habitat): Slack
- [Speech-to-text/speech recognition]
- [Text-to-speech/speech synthesis]

[Link to GitHub repository](https://github.com/fsasaki/myBot)
Quiz 5 – How to design a chatbot for translation?

fsasaki 11:15 PM
Lisa, what is the translation of Destornillador from Spanish to French

Lisa APP 11:15 PM
I will check, hold on.

The translation of Destornillador is Tournevis.
### Quiz 5 – Fineprint (1/2)

<table>
<thead>
<tr>
<th>Documents or document chunks (e.g. an article or paragraph in a knowledge base)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Utterance (e.g. a reply or even single word in the context of a chat)</td>
</tr>
<tr>
<td>Source and target language relate for example to the information consumer(s)</td>
</tr>
<tr>
<td>The &quot;quality&quot; of the input/source language differs (is for example usually lower in chat contexts)</td>
</tr>
<tr>
<td>Translation of speech needs a more comprehensive technology stack than translation of written text</td>
</tr>
<tr>
<td>Correct word/terminology choice is bound to the domain (e.g. &quot;bank&quot; in financials vs. &quot;bank&quot; in an outdoor setting)</td>
</tr>
<tr>
<td>Requirements on response time relate for example to the information consumption environment (in a chat context, a response time of one minute likely is inadequate)</td>
</tr>
<tr>
<td>Requirements on translation &quot;quality&quot; relate to the impact of information use (e.g. a wrong translation in a medical context may be fatal)</td>
</tr>
</tbody>
</table>
Requirements can be tackled in a variety of ways

- defaults (e.g. the users "preferred" target language)
- contextual knowledge (e.g. determining the user’s preferred language from his browser settings)
- dialogue modeling (e.g. requesting information about the domain of discourse in order to resolve an ambiguity)
Quiz 6 – How to get multilingual?
Conversation and the area of language, culture and local requirements

With bots, text may be all there is.

- Any understanding and control is related to language.
- There is nothing that can compensate language-related flaws (e.g. inaccuracies, ambiguities or errors).

Messengers/bots are cross-platform and cross-device.

Design of a “global” chatbot requires the same expertise as for other software products.

Special resources and considerations are needed for chatbot localization.
Sample Language Challenge Area – Intent Recognition

Sample phrases (with placeholders) related to a translation need in a chat/dialogue context

- Give me the translation of `<X>` from `<SourceLanguage>` into `<TargetLanguage>`?
- Give me the translation of `<X>`?
- Translate `<X>` from `<SourceLanguage>` into `<TargetLanguage>`?
- Translate `<X>`.
- What is the translation of `<SourceLanguage> `<X>` in `<TargetLanguage>`?
- What is the translation of `<X>`?
- How do you say `<X>` in `<TargetLanguage>`?
- What's the `<SourceLanguage>` word for `<X>` in `<TargetLanguage>`?
- What's `<X>` in `<TargetLanguage>`?
- What's `<X>` in `<TargetLanguage>` in the domain `<D>`?
Correct grammar (e.g. inflection) requires information about the bots personality traits (e.g. gender).

"I have canceled your current request."

- Male → Я отменил ваш текущий запрос.
- Female → Я отменила ваш текущий запрос.
- Neuter → Could be rephrased as ("impersonal"): „Your current request has been cancelled“ „Ваш текущий запрос отменен.“

"I created $possessivePronoun{ ${entityType/noun}}." (e.g. "I created your sales order.")

- Male → Я создал ваш заказ на продажу.
- Female → Я создала ваш заказ на продажу.
- Neuter → Could be rephrased as ("impersonal” and passive): „Your sales order has been created.“ „Ваш XXX был создан.

The verb depends on the gender of XXX.
Expressions are expected to be short. It should be possible to read and understand them immediately (e.g. on your smartphone’s lock screen).

Language may need to be abstract in the following sense: When you work with a bot from a tablet, the bot better not uses expressions such as “With the left mouse button pressed ...”.

You cannot be sure how for example the emoji you use is displayed/rendered (see for example what “hospital” can look like on different platform - [http://emojipedia.org/hospital/](http://emojipedia.org/hospital/)).
Design of a “global” chatbot requires the same localization-related expertise than for other software products

One cannot create a software product and treat translation as an afterthought. If the product design is flawed, going global will painful or even impossible.

The localization may need to take place on various levels, including the bot personality (tone, style, emotions), its gender, the natural language processing model, changing roles of icons or prompts.

In a localized bot, time zone and other elements of user context are of high importance, e.g. for a question like “When does the soccer game start?”
Indistinguishable placeholder/text variables make the reordering that is necessary in some languages impossible.

Create success: "I created $possessivePronoun{ ${entityType/noun}} near $possessivePronoun{ ${entityType/noun}}.""

"CL: In Japanese, the order of the placeholders/text variables may need to be switched."

→ All placeholders should be distinguishable (can for example be achieved via identifiers for placeholders).
Chatbots often deal with a large set of entities, e.g. place names. Speech recognition and correct pronunciation relies on the availability of corresponding multilingual data sets.

A truly multilingual bot may provide language detection, based on the user’s environment/context (e.g. browser settings).

Different input methods have different preferences depending on the target market. In some cultures, or certain situations in culture, speaking is preferred other typing, or vice-versa.

The adaption of closely related phrases can go wrong, if one does not consider domain specific terminological resources or general knowledge bases.
Avoid proprietary solutions/lock-in

The bot world requires artefacts (e.g. sample dialogue flows) to be “managed”.

Try to avoid that you will be locked-in.

Example: If you need to create 100 language-related rules to make your bot speak English for a limited domain such as leave requests, you may not want to redo this when you must switch to a different platform or framework.

Worth a look to avoid lock-in or reuse existing resources

- SRGS – Speech Recognition Grammar Specification Version 1.0
- SSML – Speech Synthesis Markup Language Version 1.0
- VoiceXML – Voice Extensible Markup Language
- SSML – Speech Synthesis Markup Language
- PLS – Pronunciation Lexicon Specification 1.0
- EmotionML – Emotion Markup Language
- Artificial Intelligence Markup Language
- Schema.org

In addition to best practices, standards or partial standard like the ones mentioned, existing software libraries and resources can be of great value.

Examples

- Frameworks to create bots, or realize necessary capabilities (e.g. user environment detection)
- Pre-defined resources for a given domain
- Libraries to handle timezones and general internationalization, e.g. moment.js or i18n-node in certain web application contexts
Expertise, Best Practices and Caveats – Example Application

Quality of processing differs between languages
- Influences especially intent recognition and entity identification
- Chatbot development needs to invest resources into providing language resources (intent training data, entity dictionaries)

One NLU component per language
- accessTokenAPIAI = {"en" : "accesstoken-for-english-agent", "de" : "accesstoken-for-german-agent"};

Externalize and localize strings
- waitmessage = {"en" : "I will check, hold on.", "de" : "Ich werde das überprüfen, einen Moment"};

Localize date expressions
- birthdatepicturestring = { "en" : "MMMM Do YYYY", "de" : "Do MMMM YYYY"};
Conclusions/Summary

Technical documentation will be affected by the rise of chatbots.

Get process/modelling experts, language experts, Natural Language Processing advisors, creatives (e.g. User Assistance experts), and more to facilitate a holistic view. Also, consider recommendations from the GUI world (e.g. related to Composite Messages).

Ask the question: What if the bot needs to “understand” and “speak” several languages? Agree on a scheme for multilingual bots.

Think about possibilities to automate (for example language-related style guides).

Related to all previous questions: think about workflows and formats for localizing content and pre-defined interactions.
Your opinion is important to us! Please tell us what you thought of the lecture. We look forward to your feedback via smartphone or tablet under http://lt02.honestly.de

or scan the QR code

The feedback tool will be available even after the conference!
All product and service names mentioned and associated logos displayed are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

This document may contain only intended strategies, developments, and is not intended to be binding upon the authors or their employers to any particular course of business, product strategy, and/or development. The authors or their employers assume no responsibility for errors or omissions in this document. The authors or their employers do not warrant the accuracy or completeness of the information, text, graphics, links, or other items contained within this material. This document is provided without a warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, or non-infringement.

The authors or their employers shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials. This limitation shall not apply in cases of intent or gross negligence.

The authors have no control over the information that you may access through the use of hot links contained in these materials and does not endorse your use of third-party Web pages nor provide any warranty whatsoever relating to third-party Web pages.
Top managers – not just from software vendors – are predicting that “(chat)bots will become the new interface”. The Graphical User Interface (GUI) as we now it thus will decrease in importance. This means that language – both written and spoken – will become the pivotal dimension of communication. Very often, the expectation is that bots will be able to “work” for any locale or language. Based on a sketch of important concepts related to the domain of bots, this contribution discusses some of the consequences of this development and derives a call-to-action related to the following insight: With bots, solution/service quality and brand perception will be tied to language more heavily. Thus, language experts, localization wizards, and creatives should get on their marks so that they can advise on bots that are “world-ready”. Furthermore, possibilities to automate production (e.g. via tools to check compliance with language-related style guides) should receive attention.

Chatbots will become the new interface

One way to look at so-called “bots” especially in messaging environments such as Skype, Twitter, Slack, Facebook messenger is the following: They are virtual contacts that enable the creation, consumption and interaction with content. That content often is provisioned by a technical service. Example: a bot that assists in scheduling an appointment may access calendars, and suggest suitable slots. After a shorter or longer “dialogue” between the user (“unfortunately, this is too tight for me”) and the bot (“How about this then?”). The expectation for the dialogue is that is “natural”: Users can use their – possibly even spontaneous and thus sometimes ungrammatical language – and bots know for example that “this” refers to a date that was mentioned earlier in the dialogue. Thus, bots become conversational User Interfaces - either voice- or text-based, or a combination of both.

It is thus understandable that top managers like Microsoft’s CEO Satya Nadella predict that “(chat)bots will become the new interface”. Since they are different from Graphical User Interfaces (GUIs), its high time to discuss requirements, prerequisites, and implications for anyone working on bots, bot platforms, or bot frameworks.

Chatbots and the language, culture and global market dimension

Where there is conversation, one dimension - possibly the pivotal one - is the area of language, culture and local requirements. Like in many new technologies, this dimension often is considered late, along the lines of “First we do an English version, other versions can be easily added later since the only thing that needs to be done is translation”. However, current discussions on chatbots reveal that the chatbot community has started to recognize flaws with this train of thought – or, depending on the point of view – opportunities to reach the next billion users. Amongst others, four areas are under discussion: text only, cross everything, design challenges, and localization considerations.
With bots, text may be all there is.
Implications (examples):
- Any understanding and control is related to language.
- There is nothing that can compensate language-related flaws (e.g. inaccuracies, ambiguities or errors).
- Solution/service quality and brand perception are tied to language more heavily.

Messengers/bots are cross-platform and cross-device.
Implications (examples):
- Expressions are expected to be short. It should be possible to read and understand them immediately (e.g. on your smartphone's lock screen).
- Language may need to be abstract in the following sense: When you work with a bot from a tablet, the bot better not uses expressions such as “With the left mouse button pressed …”
- You cannot be sure how for example the emoji you use is displayed/rendered (see for example what “hospital” can look like on different platform - http://emojipedia.org/hospital/)

Design of a “global” chatbot requires different expertise than for other software products.
Implications (examples):
- One cannot create a software product and treat translation as an afterthought. If the product design is flawed, going global will painful or even impossible.
- The localization may need to take place on various levels, including the bot personality (tone, style, emotions), its gender, the natural language processing model, changing roles of icons or prompts.
- In a localized bot, time zone and other elements of user context are of high importance, e.g. for a question like “When does the soccer game start?”

Special resources and considerations are needed for chatbot localization.
Implications (examples):
- Chatbots often deal with a large set of entities, e.g. place names. Speech recognition and correct pronunciation relies on the availability of corresponding multilingual data sets.
- A truly multilingual bot may provide language detection, based on the user's environment/context (e.g. browser settings).
- Different input methods have different preferences depending on the target market. In some cultures, or certain situations in culture, speaking is preferred other typing, or vice-versa.
- The adaption of closely related phrases can go wrong, if one does not consider domain specific terminological resources or general knowledge bases.

Best Practices
Looking at the sketch above, four calls to action seem to be adequate:
- Get process/modelling experts, language experts, Natural Language Processing advisors, creatives (e.g. User Assistance experts), and more to facilitate a holistic view. Also, consider recommendations from the GUI world (e.g. related to Composite Messages).
- Ask the question: What if the bot needs to “understand” and “speak” several languages? Agree on a scheme for multilingual bots.
- Think about possibilities to automate (for example language-related style guides).
- Related to all previous questions: think about workflows and formats for localizing content and pre-defined interactions.

**Avoid proprietary solutions/lock-in**

The bot world requires artefacts (e.g. sample dialogue flows) to be “managed”. Try to avoid that you will be locked-in. Example: If you need to create 100 language-related rules to make your bot speak English for a limited domain such as leave requests, you may not want to redo this when you must switch to a different platform or framework. Speech-related work from the World Wide Web consortium (W3C) which may be worth a look/help to avoid lock-in or reuse existing resources:

- SRGS – Speech Recognition Grammar Specification Version 1.0
- SSML – Speech Synthesis Markup Language Version 1.0
- VoiceXML – Voice Extensible Markup Language
- SSML – Speech Synthesis Markup Language
- PLS – Pronunciation Lexicon Specification 1.0

In addition to best practices, standards or partial standard like the ones mentioned, existing software libraries and resources can be of great value. Examples include:

- Frameworks to create bots, or realize necessary capabilities (e.g. user environment detection)
- Pre-defined resources for a given domain
- Libraries to handle timezones and general internationalization, e.g. moment.js or i18n-node in certain web application contexts

**Implications for technical documentation**

If the bot vision of top managers becomes real, consumption and production of technical documentation will change. Rather than doing a full text search for the item of interest, users will ask a bot “How do I change the oil of my car?”, and your bots’ interfaces points you to the right place (e.g. via Augmented Reality). Technical documentation itself might need to be enhanced to support the scenario. Feasibility, and viability will benefit from best practices, and non-proprietary solutions.

Contact:

felix@sasakiatcf.com
christian.lieske@sap.com