Experiences from Verbmobil

Norbert Reithinger  
DFKI GmbH  
Stuhlsatzenhausweg 3  
D-66123 Saarbrücken  
bert@dfki.de
Content

• Overview of Verbmobil project
• Scientific challenges and experiences
• Software technology challenges and experiences
• Management challenges and experiences
• Conclusion
Verbmobil - The Project

Some information for those who haven´t heard of Verbmobil recently

• speaker independent speech-to-speech translation system for appointment scheduling and travel planning:
  German ↔ English (10 175 words German, 6871 words English)
  German ↔ Japanese (2566 words Japanese)

• 69 modules, full configuration 3.5 GB
• 23 participating institutions (in Verbmobil II)
• over 900 full workers and students involved
• project duration: 1993 - 2000

scientific, software technology, and management challenges

N. Reithinger, DFKI
Verbmobil - The Partners

TU Dresden

RHEINISCHE FRIEDRICH WILHELM'S-UNIVERSITÄT BONN

SIEMENS

DAIMLERCHRYSLER

LMU

LUDWIG MAXIMILIANS UNIVERSITÄT MÜNCHEN

UNIVERSITÄT DES SAARLANDES

UNIVERSITÄT HAMBURG

UNIVERSITÄT TUBINGEN

EBERHARDT-KARLS UNIVERSITÄT ERLANGEN-NÜRNBERG

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG

TECHNISCHE UNIVERSITÄT MÜNCHEN

UNIVERSITÄT BIELEFELD

UNIVERSITÄT KARLSRUHE

UNIVERSITÄT STUTTGART

RUHR-UNIVERSITÄT BOCHUM

TU-BRAUNSCHWEIG

N. Reithinger, DFKI

RF, 12. Juli 2000, 4
A Remark about Project Duration

8 years is a long time, especially since the invention of Internet time

1993
  – “You will need special hardware!”
  – “1500 words speaker independent is impossible!”
  – “Aren’t your goals unrealistic?”

2000
  – “Does it run on my notebook?”
  – “Only 10 000 words?”
  – “Why can’t it also translate in the domains X, Y, and Z?”

but

it is a unique chance for
• large scale, continuous research and development
• training people, collaborating, gaining experience
• collecting and annotating data
Scientific Challenges

The goal

• Progress in the areas of speech translation

The situation 1993

• Speech understanding and translation system existed
• Knowledge distributed in various scientific fields
• There were only few interactions

The solution

• Collaborate in a project
• Cross fertilization: check the theories of the others
Scientific Results

There are over 600 refereed papers on the various aspects of and achievements in Verbmbil.
See also W. Wahlster (ed.): Verbmbil: Foundations of Speech-to-Speech Translation, Springer Verlag, to appear July 2000  ... at any shop near your office  :-)

Some highlights

- Speaker independent speech recognition over various channels
- Language ID
- Unknown words
- Prosodic information (segmentation, stress etc.) used in various modules
- Repair of hesitations, repetitions
- Combination of parser analysis fragments
- Semantic representation: VIT
- Context and dialogue knowledge supports translation
- Efficient semantic transfer
- Content to speech generation
- Word concatenative speech synthesis
- Dialogue minutes and summaries
- Large data collection with annotation on various levels (e.g. tree-banks, dialogue acts)
- ....
Collaboration for a New Functionality: Result Summaries

- Provide the users with a summary of the topics that were agreed

- Two benefits
  - have a piece of information to use in calendars etc.
  - control the translation

- Approach: exploit already existing modules for
  - content extraction
  - dialogue interpretation
  - planning the summary
  - generation
  - transfer
RESULT SUMMARY

Participants: Speaker B, Speaker A  
Date: 22.2.2000  
Time: 8:57 AM to 09:37 AM  
Theme: Appointment schedule with trip and accommodation

DIALOGUE RESULTS:

Scheduling:
Speaker B and speaker A will meet in the train station on the 1. of March 2000 at a quarter to 10 in the morning.

Travelling:
There the trip from Hamburg to Hanover by train will start on the 2. of March at 10 o'clock in the morning. The way back by train will start on the 2. of March at half past 6 in the evening.

Accommodation:
The hotel Luisenhof in Hanover was agreed on. Speaker A is taking care of the hotel reservation.
Multiple Approaches

• Mono-cultural approaches are dangerous
  – humans vs. viruses \downarrow diversity
  – Microsoft vs. ILOVEYOU and copycats \downarrow alternative software solutions

• Some sources of errors in a speech translation system
  – external
    • spontaneous speech: not well formed, hesitations, repairs
    • bad acoustic conditions
    • human dialogue behavior
  – internal
    • knowledge gaps in modules
    • software errors
    • probabilistic processing

⚠️ Use multiple engines, varying approaches on various stages of processing

N. Reithinger, DFKI

RF, 12. Juli 2000, 11
• Exclusive alternatives: three different German speech recognizers with various capabilities

• Competing approaches:
  – three parsers: HPSG, Chunk, Statistical
  – five translation tracks: case-based, dialog-act based, statistical, substring-based, linguistic (deep) semantic translation

• Needed: selection and combination of results from competing tracks
  – parsers: combination of partial analyses in the semantic processing modules
  – translation: preselection module
Multi-Engine for Translation (DOE)

- Large-Scale Web-based Evaluation: 25,345 Translations, 65 Evaluators
- Sentence Length 1 - 60 Words

<table>
<thead>
<tr>
<th>Translation Thread</th>
<th>Word Accuracy ≥ 50%</th>
<th>5069 Turns</th>
<th>Word Accuracy ≥ 75%</th>
<th>3267 Turns</th>
<th>Word Accuracy ≥ 80%</th>
<th>2723 Turns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case-based Translation</td>
<td>37%</td>
<td></td>
<td>44%</td>
<td></td>
<td>46%</td>
<td></td>
</tr>
<tr>
<td>Statistical Translation</td>
<td>69%</td>
<td></td>
<td>79%</td>
<td></td>
<td>81%</td>
<td></td>
</tr>
<tr>
<td>Dialog-Act based Translation</td>
<td>40%</td>
<td></td>
<td>45%</td>
<td></td>
<td>46%</td>
<td></td>
</tr>
<tr>
<td>Semantic Transfer</td>
<td>40%</td>
<td></td>
<td>47%</td>
<td></td>
<td>49%</td>
<td></td>
</tr>
<tr>
<td>Substring-based Translation</td>
<td>65%</td>
<td></td>
<td>75%</td>
<td></td>
<td>79%</td>
<td></td>
</tr>
<tr>
<td><strong>Automatic Selection</strong></td>
<td>57% / 78% *</td>
<td></td>
<td>66% / 83% *</td>
<td></td>
<td>68% / 85% *</td>
<td></td>
</tr>
<tr>
<td><strong>Manual Selection</strong></td>
<td>88%</td>
<td></td>
<td>95%</td>
<td></td>
<td>97%</td>
<td></td>
</tr>
</tbody>
</table>

* After Training with Instance-based Learning Algorithm
The Selection Problem

Selection is a hairy business

- confidence values are difficult to compare
  - probabilistic vs. knowledge based approaches
  - no bird’s eyes view possible
- re-training necessary after changes in the engines
- training data must be produced
Selection using Instance Based Learning


- **Software:** MLC++ toolkit
- **Test and training material** 25 345 translations with ratings
- **Classes**
  - STAT1 = statistical translation ok
  - STAT0 = statistical translation not ok
  - other tracks analogous ⇒ 10 classes
- **Used features:** length, number of fragments, mean of confidence values, minimal confidence value, number of missing translations
- **Cross validation on 10 disjoint test/training sets from the evaluation material**
- **Error:** \( \approx 22\% \) for word accuracy \( \geq 50\% \)
Experiences

• Researchers, naturally, like their own approach most
• Methods like statistical approaches spread in all fields
• Cross-fertilization worked
• New functionalities like dialog summaries through collaboration and reuse
• The multi-engine approach is a Good Thing™
• Collect data, and annotate it on various levels: indispensable for training, testing and evaluation

☐ Build ONE integrated system, where everybody is involved
Software Technology Challenges

The goal

• Build an integrated system

The situation

• Researchers do research
• Using different programming languages
• Researchers don’t want to be bothered with technical details

The solution

• Introducing: the System Group
• Maximal technical support for the researchers/developers
Support from the System Group

Integration framework (testbed) with

- common communication mechanism for all used programming languages (C, C++, Lisp, Prolog, Java, Fortran, Tcl/Tk)
- Narrow interface for all used programming languages
- Overall system control infrastructure
- Standards on various levels
  - Installation
  - Compilation
  - Communication formats between modules
  - ...
- Toolbox for recording, replaying, testing, inspecting data exchanged between modules, ...

N. Reithinger, DFKI
The Architecture

Verbmobil I

Multi-Agent Architecture

- Modules know all communication partners
- Direct communication between modules
- Reconfiguration difficult
- Software: ICE and ICE Master
- Basic Platform: PVM

Verbmobil II

Multi-Blackboard Architecture

- Modules know their I/O data pools
- No direct communication between modules
- 198 blackboards vs. 2380 direct comm. paths
- Reconfiguration easy
- Several instances of one module/functionality
- Software: PCA and Module Manager
- Basic Platform: PVM
Experience

• The System Group is a Good Thing™
• The multi blackboard architecture is a Good Thing™
• Crucial for the success of Verbmbobil
• Software foundation for (almost) hassle free module development

Controlled distributed development possible
Management Challenges

The goal

• Build an integrated system

The situation

• Partners distributed and pretty independent
• Great variation in project experience
• Adjustment of project plan and goals over time needed

The solution

• Define a flat management structure
• Create a group spirit
Project Organization

Scientific Management

- Scientific Head: W. Wahlster
- Deputy Scientific Head: A. Waibel
- Head of Project Management Group: R. Karger

Group of Module Managers

- Manager Module 1
- Manager Module n
- Module Coordinator: N. Reithinger
- Head of System Integration Group: A. Klüter

Verbmobil Advisory Board

German Federal Ministry for Research and Education

Verbmobil Consortium

Steering Committee

DLR G. Klein

N. Reithinger, DFKI
Module Managers

• Have technical hands on experience
• Responsible for one module, even if it is developed at different sites
• Volunteers (sort of ...)
• Meet regularly
• Define next milestones
• Define data and software integration plans

Module coordinator coordinates the efforts and is the link to the scientific management
Example: Optimization Schedule 2000

• 21.02. Delivery of CeBit system

• 21.02. - 30.04. Optimization phase
  – 15.03. - 28.04. End-To-End evaluation with feedback to developers
  – 27.03. - 07.04. Workshop Deep Processing

• 09.05. Delivery Verbmobil System 1.0
  • starting 09.05
    – speech recognizer evaluation
    – turn evaluation
Experience

- The group of module managers is a Good Thing™
- Common goals motivate
- Friendly peer pressure works most of the time
- Early problem detection and resolution in most cases
- Regular integration cycles focus and motivate

☑ Proactive consensus management (PCM)
Conclusion

• Shared some experiences from a large distributed project


• Just before Coling 2000 (http://www.coling.org)

• Experiences will be used in SmartKom (http://www.smartkom.org) and other projects
Thanks to ...  

- Wolfgang Wahlster (especially for a few slides :-)  
- Jan Alexandersson, Michael Kipp, and Ralf Engel  
- Reinhard Karger and Andreas Klüter  
- all student assistants