NLU vs. Dialog Management: To Whom am I Speaking?

Dirk Schnelle-Walka
Harman International
Germany
dirk.schnelle-walka@harman.com

Stefan Radomski
TU Darmstadt
Germany
radomski@tk.informatik.tu-darmstadt.de

Benjamin Milde
TU Darmstadt
Germany
milde@lt.informatik.tu-darmstadt.de

Chris Biemann
TU Darmstadt
Germany
biem@lt.informatik.tu-darmstadt.de

Max Mühlhäuser
TU Darmstadt
Germany
max@informatik.tu-darmstadt.de

ABSTRACT
Research in dialog management and natural language understanding are both approaching voice-based interaction. Coming from different perspectives they emphasize different components in the spoken dialog system processing chain. Although each approach is suitable to provide a satisfiable user experience, a combined approach could potentially improve towards a more convincing natural interaction with the user as discussed in this vision paper.

Author Keywords
natural language understanding; dialog management; intelligent personal assistants; user experience

ACM Classification Keywords
H.5.m User Interfaces: Miscellaneous

DIALOG MANAGEMENT
Speech is considered to provide an efficient and pleasant way to interact with smart objects [29]. Historically, these systems were built along a processing chain to actually initiate actions based on the user's utterance and/or produce spoken output in return. A general architecture, according to Kunzmann [13], of these system is shown in Figure 1. Pieraccini describes the components in [21] as follows: The Automated Speech Recognition (ASR) component converts the raw audio input into a sequence of words (or the n-best results). This is forwarded to a Natural Language Understanding (NLU) component to extract the semantics of the utterance. This is used by the dialog manager (DM) to decide upon the action to take according to the employed dialog strategy. The DM may use stored contextual information derived from previous dialog turns. One of the actions, a DM may take is the generation of spoken output. Therefore, the response generation (RG) generates text as an output which is passed to the text-to-speech engine (TTS) component to be synthesized into an utterance.

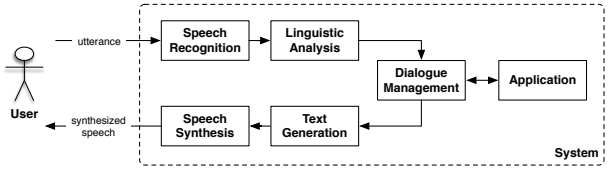

Figure 1. General architecture of a spoken dialog system

Research has been centered around DM for many years. One of the main efforts was the development of suitable dialog strategies for a more natural user experience. Radomski provides a thorough analysis of the related terms in [22] for *dialog*, *dialog management* and *dialog strategy*. Based on various definitions throughout literature, e.g. by Traum [27] or Rudnicky [23] he comes the following definitions for multimodal dialogs. We adapted them to voice user interfaces.

Definition 1 *A **dialog** is a sequence of interleaved, communicative events between a human and a computer to convey information aurally.*

Definition 2 *A **Dialog Manager** is a software component responsible for maintaining the dialogs state and driving the interaction by mapping relevant user input events onto system responses as output events. Performing these responsibilities is also referred to as dialog management.*

IUI 2016 Workshop: Interacting with Smart Objects, March 10th, 2016, Sonoma, CA, USA

DOI: 10.13140/RG.2.1.1928.4247

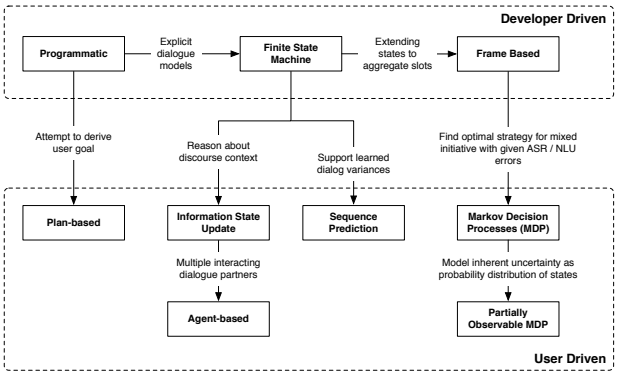

Figure 2. Pattern Language for Dialog Management

Definition 3 *A **Dialog Strategy** is a conceptualization of a dialog for an operationalization in a computer system. It defines the representation of the dialogs state and respective operations to process and generate events relevant to the interaction.*

Schnelle-Walka et al. [24, 25] developed a pattern language thereof as shown in Figure 2. They identified the following strategies: (i) Programmatic Dialog Management, (ii) Finite State Dialog Management, (iii) Frame Based Dialog Management, (iv) Information State Update [14], (v) Plan Based, (vi) Markov Decision Process [16] and (vii) Partially Observable MDP [31]. Each strategy has its strengths and weaknesses. Some are more restricted while others allow for less constrained user input. Generally, the system used to define the degree of freedom that users have while interacting with the system. They all share the DM-centered perspective regarding the NLU to be some input into the system while the decision upon subsequent interaction is being handled in this component.

This concept has also been applied to multimodal approaches to DM, like PAC-AMODEUS [6], TrindiKit [15], Jaspis [28] or MIMUS [2] as well as high level architectures [17].

NATURAL LANGUAGE UNDERSTANDING

Natural language understanding (NLU) is a subtopic of natural language processing in artificial intelligence that deals with machine reading [8] comprehension. NLU targets the automatic comprehension of entire documents without anticipating their content.

In the past years, performance of NLU increased dramatically as sketched by Cambria [5] and shown in Figure 3. Today's NLU moves away from the *Syntactical Curve* to the *Semantic Curve*. While the previous one focuses on processing of documents at a syntax level, like keyword or word co-occurrence count, newer concepts "rely on implicit denotative features associated with natural language text" [5].

Research in NLU usually learns from large document sets. One application that demonstrates the level of understanding is to query for information, also known as *Question Answering*. This is, where interaction with the user comes into play.

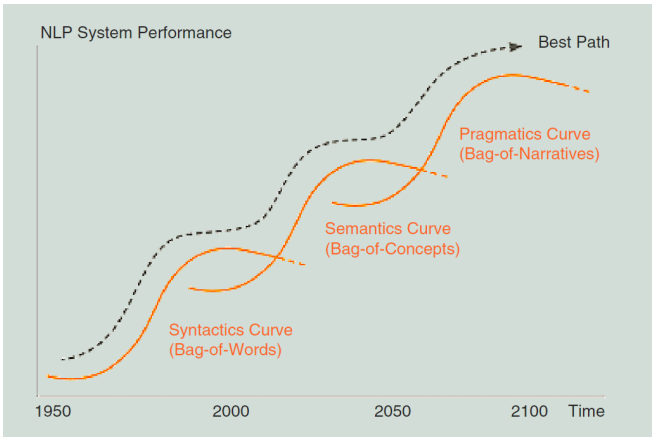

Figure 3. Envisioned evolution of NLU research through different eras or curves

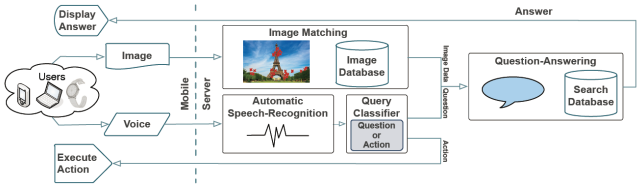

Figure 4. Pipeline of Sirius as an example for an intelligent personal assistant

Hence, a typical example is seen in the development of intelligent personal assistants (IPA). One example of such an IPA is the open source IPA Sirius from Hauswald et al. [10] as shown in Figure 4. Other examples of IPAs that expose their API to developers include IBM Watson [9] and LUIS [1] from Microsoft. While the first IPAs were only able to cope with a single dialog turn, newer systems also establish dialog context. Thus, they are able to also refer to previously entered input and, e.g. iteratively refine query results by adding or removing parameters as needed. This way, they are adopting tasks, such as maintaining the conversational state, that researchers in dialog management see as one of the core tasks of a DM.

CONTRASTING NLU AND DM

While the AI community usually focuses on NLU, the spoken dialog community focuses on the DM as the central point in this chain. Both have good reasons for their approach and are able to deliver convincing results.

DM-centered systems are principally constrained because they anticipate the users input as plans to help them to achieve their goal. Depending on the implemented dialog strategy they allow for different degrees of flexibility.

NLU-centered systems see the central point in the semantics of the utterance, which should also be grounded with previous utterances or external content. Thus, whether speech or not, NLU regards the stream as some input and produces some output. Since no dialog model is employed, resulting user interfaces currently do not handle much more than single queries.

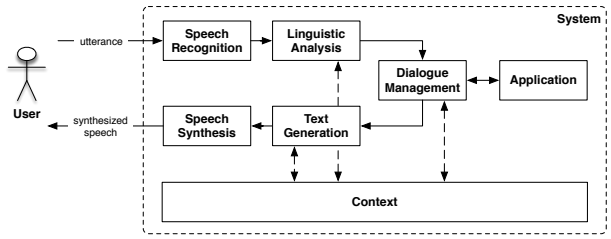

Figure 5. General architecture of a spoken dialog system including context management

Currently, efforts towards spoken interaction coming from this domain are still not fully aware of what has been done in DM research in the past decades, and vice versa. Both parties are coming from different regions in the chain of spoken dialog systems. For instance, `api.ai` recently announced that their system now supports slot filling[1]. The biggest challenges are seen in determining the user's intent and semantic slot filling [18]. The user may use these to refine a query until he ends with a single result. Current spoken dialog systems are already beyond that and are able provide good voice user interface design. For instance, grounding strategies as they are introduced e.g., by Larsson in his Information State Update approach [14], are not exploited. Another important aspect are dialog acts [3]. Interaction with smart objects must go beyond the question-answer paradigm and rely on, e.g., reject, accept, request-suggest, give-reason, confirm, clarify. And finally, uncertainty in the recognition result [26] is not considered at all. NLU focused systems rely on their ability that the user can replace any value at any time. Therefore, he will have to understand the received result and correct it as needed. The developed strategies for error correction and error prevention, as they have been researched in the DM community for years [4], like explicit or implicit confirmation, remain unexploited.

As it comes to maintaining the conversational state, both NLU and DM will need to access it. NLU will need it, e.g., to correctly determine linguistic phenomena like ellipsis or anaphoric references. DM will need it to allow for a more natural dialog flow to produce the right output following dialog theoretical aspects. Context must be accessible and manipulatable from both components, as shown in Figure 5. This aspect was already addressed, e.g. by Oviat [19] who added a *Context Management* component to the processing chain. Coming from multimodal fusion she demands for a canonical meaning representation.

SUMMARY AND OUTLOOK

In this paper, we had a look at the approaches of the community of DM and the community of NLU to voice-based interaction. We described both views onto it, that emphasize different components in the processing pipeline. Subsequently, we explored synergy effects of both views.

For a more convincing user experience both communities will be in the need of adopting techniques from the other community. The capabilities of today's NLU are already convincing. There are lacks in how to engage the user into a real conversation. These techniques have been well developed in the domain of dialog management. Adoption of dialog theory will allow for a more natural interaction.

We believe, that is time that both communities start talking to each other to better incorporate results of "the other component" to arrive at a convincing user experience. Maybe, POMPDP [31] dialog systems are a good candidate to be employed as they are also based on machine learning techniques that provided a breakthrough in NLU and are the most advanced dialog strategy. Maluuba[2], a Candaian NLU centered company already started rolling out such systems.

However, future systems may differ from what has been described above. *Cognitive Computing* is about to change the way how voice-based interactive systems will be developed in the future. We follow the definition given in [12].

Definition 4 ***Cognitive Computing*** *refers to systems that learn at scale, reason with purpose and interact with humans naturally. Rather than being explicitly programmed, they learn and reason from their interactions with us and from their experiences with their environment.*

This has implications for voice-based interaction: (i) It would be desirable if voice-based system would learn and get better while being used, instead of being statically defined or trained. This can apply to speech recognition, NLU and text generation components, with online learning from implicit or explicit user feedback. Some headway is also being made in the machine learning community in the form of proactive learning [7], as user feedback can be subjective and must be judged according to its information value. (ii) Making voice-based interaction more natural would also entail that responses are not programmed, but produced by a generative model. (iii) The ability to transfer and use knowledge from known domains and tasks to previously unknown and new tasks is also a building block of cognitive computing systems. Dialog systems could also benefit from the transfer learning paradigm [20], as it offers solutions for data scarcity in a particular domain. An example would be a tourist information dialog system that transfers what has been learned in a restaurant recommendation dialog system.

The Recurrent Neural Network (RNN) framework is a candidate that could make (i)-(iii) possible, with some recent and promising first results [11, 30].

REFERENCES

1. Fast and easy language understanding for dialog systems with Microsoft Language Understanding Intelligent Service (LUIS). In *Special Interest Group on Discourse and Dialogue.* (2015), 159–161.
2. Amores, J. G., Pérez, G., and Manchón, P. MIMUS: a multimodal and multilingual dialogue system for the

[1] `https://api.ai/blog/2015/11/09/SlotFilling/`

[2] `http://maluuba.com`

home domain. In *ACL*, Association for Computational Linguistics (2007), 1–4.

3. Austin, J. L. *How to do things with words*, vol. 367. Oxford University Press, 1975.
4. Bohus, D., and Rudnicky, A. Sorry, I Didn't Catch That! - An Investigation of Non-understanding Errors and Recovery Strategies. In *SIGdial Workshop on Discourse and Dialogue* (2005).
5. Cambria, E., and White, B. Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine* (2014), 48–57.
6. Coutaz, J. PAC: An object-oriented model for dialog design. In *Human Computer Interaction (INTERACT)* (1987), 431–436.
7. Donmez, P., and Carbonell, J. G. Proactive learning: cost-sensitive active learning with multiple imperfect oracles. In *ACM conference on Information and knowledge management* (2008), 619–628.
8. Etzioni, O., Banko, M., and Cafarella, M. J. Machine Reading. *AAAI 6* (2006), 1517–1519.
9. Ferrucci, D., Brown, E., Chu-Carroll, J., Fan, J., Gondek, D., Kalyanpur, A. a., Lally, A., Murdock, J. W., Nyberg, E., Prager, J., Schlaefer, N., and Welty, C. Building Watson: An Overview of the DeepQA Project. *AI Magazine 31*, 3 (2010), 59–79.
10. Hauswald, J., Laurenzano, M. A., Zhang, Y., Li, C., Rovinski, A., Khurana, A., Dreslinski, R. G., Mudge, T., Petrucci, V., Tang, L., and Mars, J. Sirius : An Open End-to-End Voice and Vision Personal Assistant and Its Implications for Future Warehouse Scale Computers. In *Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)* (2015).
11. Henderson, M., Thomson, B., and Young, S. Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation. In *Spoken Language Technology Workshop (SLT)*, IEEE (2014), 360–365.
12. Kelly, J. E. Computing, cognition and the future of knowing. Whitepaper, IBM Reseach, 2015.
13. Kunzmann, S. Applied speech processing technologies-our journey. *European Language Resources Association Newsletter (ELRA)* (2000), 6–8.
14. Larsson, S. *Issue-based Dialogue Management*. PhD Thesis, University of Gothenburg, 2002.
15. Larsson, S., and Traum, D. R. Information state and dialogue management in the TRINDI dialogue move engine toolkit. *Natural Language Engineering 6*, 3&4 (2000), 323–340.
16. Levin, E., Pieraccini, R., and Eckert, W. Using Markov Decision Process for Learning Dialogue Strategies. In *Conference on Acoustics, Speech and Signal Processing.*, vol. 1 (1998), 201–204.
17. Maybury, M. T., and Wahlster, W., Eds. *Readings in intelligent user interfaces*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1998.
18. Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing 23*, 3 (2015), 530–539.
19. Oviatt, S. Multimodal Interfaces. In *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, J. A. J. Sears and A., Eds. Lawrence Erlbaum Assoc., Mahwah, NJ, 2012, 413–432.
20. Pan, S. J., and Yang, Q. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering 22*, 10 (2010), 1345–1359.
21. Pieraccini, R., and Huerta, J. Where do we go from here? Research and commercial spoken dialog systems. In *SIGdial Workshop on Discourse and Dialogue* (2005), 2–3.
22. Radomski, S. *Formal Verification of Multimodal Dialogs in Pervasive Environments*. PhD Thesis, Technische Universität Darmstadt, 2015.
23. Rudnicky, A., and Xu, W. An agenda-based dialog management architecture for spoken language systems. In *IEEE Automatic Speech Recognition and Understanding Workshop* (1999).
24. Schnelle-Walka, D., and Radomski, S. A Pattern Language for Dialog Management. In *VikingPLoP* (2012), 1–8.
25. Schnelle-Walka, D., and Radomski, S. Probabilistic Dialog Management. In *VikingPLoP* (2013), 1–12.
26. Shneiderman, B. The limits of speech recognition. *Communications of the ACM 43*, 9 (2000), 63–65.
27. Traum, D. Conversational Agency: The Trains-93 Dialogue Manager. *Workshop on Language Technology: Dialogue Management in Natural Language Systems* (1996), 1–11.
28. Turunen, M., Hakulinen, J., Räihä, K.-J., Salonen, E.-P., Kainulainen, A., and Prusi, P. Jaspis An architecture and applications for speech-based accessibility systems. *IBM Systems Journal 44*, 3 (2005), 485–504.
29. Turunen, M., Sonntag, D., Engelbrecht, K.-P., Olsson, T., Schnelle-Walka, D., and Lucero, A. Interaction and Humans in Internet of Things. In *Human-Computer Interaction (INTERACT)*, J. Abascal, S. Barbosa, M. Fetter, T. Gross, P. Palanque, and M. Winckler, Eds., Springer Berlin / Heidelberg (2015), 633–636.
30. Wen, T.-H., Gasic, M., Mrksic, N., Su, P.-H., Vandyke, D., and Young, S. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. *Empirical Methods on Natural Language Processing (EMNLP)* (2015).
31. Young, S. Using POMPDs for Dialog Management. In *Spoken Language Technology Workshop, 2006. IEEE* (2006), 8 –13.