Automatic Modification of Communication Style in Dialogue Management

Louisa Pragst[1], **Juliana Miehle**[1], **Stefan Ultes**[2], **Wolfgang Minker**[1]
[1] Institute of Communications Engineering, Ulm University, Germany
{firstname.lastname}@uni-ulm.de
[2] Engineering Department, University of Cambridge, UK
su259@cam.ac.uk

Abstract

In task-oriented dialogues, there is often only one right answer the system can give. However, a lack of variation can seem repetitive and unnatural. Humans change the way they express something, e.g. by being more or less concise. We aim to approximate this ability by automatically varying the level of verbosity and directness of a given system action. In this work, we illustrate how verbosity and directness may be utilised in adaptive dialogue management and present different approaches to automatically generate varying levels of verbosity and directness for given system actions. Thereby, new and unforeseen system actions can be created dynamically.

1 Introduction

In a dialogue system, the Dialogue Management (DM) is responsible for selecting the system's next action, thereby shaping the flow of the conversation between human and computer. To this end, all possible system actions that can be executed are defined in advance. The definition of system actions requires diligence. Each system action needs to be foreseen and included. If an action is not foreseen in advance the resulting dialogue system is unable to react adequately. Adaptive DM puts even higher requirements on the definition of system actions as suitable possibilities for adaptation have to be provided in addition to the elementary dialogue flow.

One possibility for adaptation is changing the way information is communicated. Pragst (2015) shows that the level of verbosity and directness (in the following summarised as Communication Style (CS)) has a situation-dependent influence on the user's assessment of the dialogue. By automatically generating CS variants of system actions, the potential for adaptivity is increased without additional work during the definition of system actions. Furthermore, new and unforeseen system actions can be created by the system without human interference.

In this paper, we introduce approaches for the automatic generation of CS variants of system actions. We present CSs suitable for adaptive DM and discuss their applicability. Furthermore, we show how new system actions can be generated automatically by changing the CS of existing system actions.

The remainder of the paper is structured as follows: in Section 2, related work is introduced and put in relation to the work at hand. Subsequently, we introduce the CSs that are targeted for the automatic generation, namely verbosity and directness. Section 4 provides examples for the application of CS in adaptive DM. In Section 5, we introduce the architecture of our system and in Section 6, we give an overview over the utilised corpus. Our approach to the automatic generation of variations of system actions is presented in Section 7. Finally, we draw a conclusion and propose future work in Section 8.

2 Related Work

Adaptive DM has been in the focus of researchers for a long time. Hence, a number of adaptive DM architectures exists, e.g. (Gnjatović and Rösner, 2008; Ultes and Minker, 2014; Rieser and Lemon, 2011). Adaptation of the dialogue strategy to culture (Aylett and Paiva, 2012; Mascarenhas et al., 2013) as well as emotion (André et al., 2004; Gnja-

Proceedings of the INLG 2016 Workshop on Computational Creativity and Natural Language Generation, pages 36–40,
Edinburgh, September 2016. ©2016 Association for Computational Linguistics

tović and Rösner, 2008; Pittermann and Pittermann, 2007), among many others, has been implemented. While such architectures provide the means to execute an adaptive dialogue strategy, they rely on predefined system actions to provide a variety of system actions for adaptation.

In the area of language generation, paraphrasing is a major field of research and a lot of approaches for automating paraphrasing have been discussed, e.g. (Kozlowski et al., 2003; Langkilde and Knight, 1998). While this research addresses the generation of variations in the language output with the same semantic content, we focus on variations of the semantic content of a system action.

There have been efforts to model dialogue flows automatically, e.g (Beveridge and Fox, 2006; Niraula et al., 2014; Zhai and Williams, 2014; Kadlec et al., 2015). Their focus lies on the automatic extraction of complete dialogue flows and is strictly task-oriented. While in some of those examples system actions are extracted automatically, those system actions are reproductions. Also, no variants are provided as those would be unnecessary to solve a task. In contrast, we aim to enable our system to generate new ways to express the same semantics and thereby create possibilities for adaptation.

3 Introducing the Communication Styles

The CSs we take into account for realising adaptivity in DM stem from the Communication Sciences. In this paper, we address the level of verbosity and directness as they are feasible for adaptation to the user's situation (Pragst, 2015). We present their origins and a short description in the following.

Verbosity as CS is derived from Kaplan's description of cultural thought patterns (Kaplan, 1966) which refer to the way arguments are presented in written text. Amongst others, the thought patterns can be distinguished by the amount of information that is provided. The fact that different amounts of information are preferred by different cultures is also reported by Feghali (1997). Unnecessary information can be distracting from the main message to people accustomed to other thought patterns.

Regarding directness, Feghali (1997) observes that in some cultures it is favoured and expected to directly express your intent, while others prefer a more indirect CS whereby the listener has to deduce the intent from the context. In such cultures, directness can even be perceived as aggressive. An example for different levels of directness is saying either 'Take an aspirin.' or 'People often use aspirin when they have a headache.'

4 Application in Dialogue

Considering their origin as CS variants of different cultures, verbosity and directness are suitable choices for adaptation to the user's culture. Here, we present further application scenarios.

4.1 User Situation

It can be useful to include unrequested information in a system action. Useful background knowledge may be provided and an indirect CS can help to keep the user focused on the conversation as their constant attention is required to grasp the context.

However, in critical situations, e.g. if the user is driving, distracting them should be avoided. This may be achieved by being concise. Similarly, direct, unambiguous system actions do not have to be interpreted and thereby decrease the cognitive load.

The level of directness and verbosity was shown to be feasible as means of adaptation to the user situation by Pragst (2015).

4.2 User Emotion

Several studies indicate that adaptation to the user emotion can improve the user experience (e.g. (Bertrand et al., 2011)). CS may be a suitable mean to implement such adaptation.

A sad user may benefit from a system communicating in their culture's style as it provides a sense of familiarity. An angry user may prefer concise and direct system actions, because they are unwilling to listen to lengthy statements. This of course is also influenced by the user's culture. If direct statements are perceived as aggressive, they will likely disgruntle the user even more.

Considering the given examples, we are confident that CS is a suitable way to adapt the system behaviour to the user emotion.

4.3 System Emotion

Endowing emotion on a dialogue system can render the system more human-like and relatable. In addition to expressing the system emotion by modalities such as facial expression, it can be used in adaptive DM in order to provide system behaviour that is consistent with the system emotion. For example, anticipation might lead to concise, direct statements to avoid a delay of the anticipated event. As described in Section 3, a deviation in rhetorical style from the cultural norm often carries emotional meaning: directness can be perceived as aggression in indirect cultures. Hence, we are convinced that the presented CSs can be used to express emotion.

5 System Architecture

Our approaches to the automatic generation of CS are developed as part of the KRISTINA Project (Wanner et al., 2016; Meditskos et al., 2016). At the core of the aspired system, a DM component decides on the next system action. It is supported by a Knowledge Base (KB) that provides the core semantic information to a user request as RDF statement. Furthermore, the KB can be queried for further information. The DM decides whether to act independently, e.g. by requesting a confirmation of the user, or to use the statements provided by the KB in a system action. In that case, it also selects and generates a suitable CS. A language generation component transforms the semantic information it gets from the DM into sentences. This ensures the flexibility needed to generate the system output for newly devised system actions.

6 Data

In the scope of the KRISTINA project, extensive corpus recordings are being performed. The corpus encompasses over ten hours of recordings and 400 dialogues at the time of publication and the recordings are still ongoing. We train our model with conversations between German, Turkish, Polish, Spanish and Arab people. Topics include personal habits and preferences as well as medical questions, amongst others.

Current annotations cover semantic content and emotion (Sukno et al., 2016). They are being utilised and enhanced to support the automatic generation of system actions. To this end, the verbosity of a system action is derived from the number of its semantic topics. Annotators determine the directness of a system action by assessing its intent. Where necessary, they provide the underlying semantic content.

7 Generation of Communication Styles

Variants of system action with the same semantic content but different CS must be available to the DM in order to enable adaptation. While it is possible to define a list of all system actions with all CS variations, it represents an immense amount of work. In this section, we introduce approaches to automatically generate variations of system actions with regard to their CS.

7.1 Level of Verbosity

The level of verbosity reflects how much information is given in addition to the core semantics. Answering a question with low verbosity may consist of a simple ‘No’. A slight increase of verbosity might result in ‘No, that is wrong’, while a high level of verbosity can result in a lengthy answer such as ‘No, that is wrong. This is the actual fact. Here is some further background information’.

The challenge of automatically generating verbosity is to determine relevant further information. The same KB that provides the statements of the core semantic content is used to this end. Starting with the resources of the core statements, the KB is queried for further statements that contain them. Such statements are included in the system action as additional information. By recursively repeating this process on the newly gathered statements, an arbitrary amount of additional information can be acquired. The level of verbosity chosen by the DM determines when to stop the collection of new information. This is modified by the relevance of the information: relevant information is pursued longer, while irrelevant information is disregarded.

The relevance of the acquired information is derived from the dialogues collected in the KRISTINA corpus and adjusted with each conversation the system participates in. The overall frequency of a class or property in all conversations as well as their frequency in combination with the classes and properties of the core statements is used as indication of importance. Furthermore, the relevance of an information decreases if it is present in the recent dialogue history.

By taking into consideration the user input, the core semantic content, the targeted level of verbosity and the dialogue history, which change with each dialogue turn, as well as the content of the KB and the importance rating, which also change over time, the additional content of a system action is selected based on a unique combination of input information. This results in the production diverse, ever-changing and unforeseen new system actions.

7.2 Level of Directness

The level of directness relates to the degree to which the core information is concealed. A direct way to help the user with a problem would be 'Do this.', while the indirect approach could be 'People often do this when they have that problem'. Instead of a direct question to gather missing information from the user, the system can also utilise 'I still need this information' or 'This information is missing'. Explicit and implicit confirmation strategies are a common example of different levels of directness in DM.

It is more complex to change the level of directness than that of verbosity. The semantic content of a system action needs to be changed while preserving its implications to achieve indirectness.

To generate alternative, more indirect system actions, we use predefined templates for each category of system action. For example, a request for missing information is replaced by a statement that the information is missing. The templates are chosen dynamically and filled with relevant content derived from the original system action. This approach relies on predefining all possible alternatives. We pursue two further approaches that result in a more autonomous system: supervised learning and reward functions.

The generation of indirect versions of system actions can be trained by supervised learning, using the level of directness and the core semantics, as determined by the annotators, as input data and the original semantics as target. This requires suitably annotated data. Each dialogue contribution has to be annotated with not only the actual, but also the underlying semantics. The annotators must be able to reliably determine the hidden (and possibly ambiguous) direct meaning. With our annotation of the KRISTINA corpus we make an effort to obtain suitable data. Using this approach, indirect system actions can be automatically extracted from a corpus. While this improves the flexibility compared to hand-crafted templates, it does not enable the system to create new system actions autonomously. This can be achieved with reward functions.

When training with reward functions, the system tests arbitrary statements as substitute for the core statements. Those are extracted from the KB. A reward is given if the alternative version still achieves the desired goal. This way, appropriate substitutes are detected and are more likely to be generated in the future. A number of alternatives may be predefined to provide a starting point and avoid unguided exploration. For this approach, a well-defined goal for each system action is required. Furthermore, the system needs to be able to determine automatically if the goal is achieved. For questions, the goal is to get information from the user. It is achieved if the user provides the desired information. For statements, one common goal is to inform the user. The success of a statement is evaluated by requesting the user to repeat the provided information. Using reward functions is the approach with the highest amount of independence as it allows for the generation of unforeseen system actions.

8 Conclusion and Future Work

In this paper, we presented approaches to the automatic generation of new system actions that are based on changing the verbosity and directness of existing system actions. We provided exemplary applications to illustrate the potential for adaptation of those newly created system actions and proposed approaches to implement their autonomous generation.

The level of verbosity can be altered by adding semantic content retrieved from a KB to the system action. Templates can be used to provide alternative semantics for indirect system actions; machine learning approaches offer more flexibility.

In future work, we will evaluate our approaches for automatic generation of CS by confirming the soundness of the newly created system actions in the context of the dialogue. Furthermore, user studies will be conducted test the aptitude of CS to enable adaptation to the user and system emotion. Additional user studies will determine which CS is suited best for which emotion, with special consideration of the influence of the culture.

Acknowledgements

This paper is part of a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 645012.

References

Elisabeth André, Matthias Rehm, Wolfgang Minker, and Dirk Bühler. 2004. Endowing spoken language dialogue systems with emotional intelligence. In *Affective Dialogue Systems*, pages 178–187. Springer.

Ruth Aylett and Ana Paiva. 2012. Computational modelling of culture and affect. *Emotion Review*, 4(3):253–263.

Gregor Bertrand, Florian Nothdurft, Wolfgang Minker, Harald Traue, and Steffen Walter. 2011. Adapting dialogue to user emotion-a wizard-of-oz study for adaptation strategies. *Proc. of IWSDS*, pages 285–294.

Martin Beveridge and John Fox. 2006. Automatic generation of spoken dialogue from medical plans and ontologies. *Journal of biomedical informatics*, 39(5):482–499.

Ellen Feghali. 1997. Arab cultural communication patterns. *International Journal of Intercultural Relations*, 21(3):345–378.

Milan Gnjatović and Dietmar Rösner. 2008. Adaptive dialogue management in the NIMITEK prototype system. In *Perception in Multimodal Dialogue Systems*, pages 14–25. Springer.

Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. 2015. Improved deep learning baselines for ubuntu corpus dialogs. *arXiv preprint arXiv:1510.03753*.

Robert B Kaplan. 1966. Cultural thought patterns in inter-cultural education. *Language learning*, 16(1-2):1–20.

Raymond Kozlowski, Kathleen F McCoy, and K Vijay-Shanker. 2003. Generation of single-sentence paraphrases from predicate/argument structure using lexico-grammatical resources. In *Proceedings of the second international workshop on Paraphrasing-Volume 16*, pages 1–8. Association for Computational Linguistics.

Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 704–710. Association for Computational Linguistics.

Samuel Mascarenhas, Rui Prada, Ana Paiva, and Gert Jan Hofstede. 2013. Social importance dynamics: a model for culturally-adaptive agents. In *Intelligent virtual agents*, pages 325–338. Springer.

Georgios Meditskos, Stamatia Dasiopoulou, Louisa Pragst, Stefan Ultes, Stefanos Vrochidis, Ioannis Kompatsiaris, and Leo Wanner. 2016. Towards an ontology-driven adaptive dialogue framework. In *Proceedings of the 1st International Workshop on Multimedia Analysis and Retrieval for Multimodal Interaction*, pages 15–20. ACM.

Nobal B Niraula, Amanda Stent, Hyuckchul Jung, Giuseppe Di Fabbrizio, I Dan Melamed, and Vasile Rus. 2014. Forms2dialog: Automatic dialog generation for web tasks. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 608–613. IEEE.

Johannes Pittermann and Angela Pittermann. 2007. A data-oriented approach to integrate emotions in adaptive dialogue management. In *Proceedings of the 12th international conference on Intelligent user interfaces*, pages 270–273. ACM.

Louisa Pragst. 2015. Multimodal adaptive dialogue management in OwlSpeak.

Verena Rieser and Oliver Lemon. 2011. *Reinforcement learning for adaptive dialogue systems: a data-driven methodology for dialogue management and natural language generation*. Springer Science & Business Media.

Federico Sukno, Mónica Domínguez, Adria Ruiz, Dominik Schiller, Florian Lingenfelser, Louisa Pragst, Eleni Kamateri, and Stefanos Vrochidis. 2016. A multimodal annotation schema for non-verbal affective analysis in the health-care domain. In *Proceedings of the 1st International Workshop on Multimedia Analysis and Retrieval for Multimodal Interaction*, pages 9–14. ACM.

Stefan Ultes and Wolfgang Minker. 2014. Managing adaptive spoken dialogue for intelligent environments. *Journal of Ambient Intelligence and Smart Environments*, 6(5):523–539.

Leo Wanner, Josep Blat, Stamatia Dasiopoulou, Mónica Domínguez, Gerard Llorach, Simon Mille, Federico Sukno, Eleni Kamateri, Ioannis Kompatsiaris, Stefanos Vrochidis, et al. 2016. Towards a multimedia knowledge-based agent with social competence and human interaction capabilities. In *Proceedings of the 1st International Workshop on Multimedia Analysis and Retrieval for Multimodal Interaction*, pages 21–26. ACM.

Ke Zhai and Jason D Williams. 2014. Discovering latent structure in task-oriented dialogues. In *ACL (1)*, pages 36–46.