Conflict Search Graph for Common Ground Consistency checks in Dialogue Systems

Maria Di Maro
University of Naples
"Federico II"
maria.dimaro2@unina.it

Antonio Origlia
University of Naples
"Federico II"
antonio.origlia@unina.it

Francesco Cutugno
University of Naples
"Federico II"
cutugno@unina.it

Abstract

In this work, we account for the formalisation of a Conflict Search Graph as a module managing domain knowledge, dialogue state tracking and information consistency in dialogue systems. Insights on its ability to recognise Common Ground Inconsistencies and make them explicit via specific linguistic feedback are also reported.

1 Introduction

The wide success and the current spread of conversational agents are shedding a new light not only on conversation analysis but also on computational pragmatics. In fact, beside the study of dialogue systems architectures, training techniques and materials, many other aspects are important when dealing with conversational agents. Among them one is not to take for granted: *Understanding* (i.e., in terms of words identification, word meaning, and *speaker meaning/intention*). To make Understanding an easier task, messages are, usually, encoded upon the so called *common ground*.

As pointed out by scholars such as Clark (1996), to pursue the aim of succeeding in the joint activity of conversation, the interlocutors need to *ground* what is being communicated. *Grounding* refers to the act of establishing that what we intend to say (or what has been said) can be well understood (or has been well understood) (Clark and Brennan, 1991). To establish a *common ground* (CG), different strategies, such as linguistic or para-linguistic feedback (Traum, 1999), are adopted. From the linguistic point of view, dialogue efficiency can rely on the analysis of communicative feedback, whose relevance was pointed up by Allwood et al. (1992) and which continues to be considered as an important characteristics in dialogue modelling (Buschmeier and Kopp, 2018).

In this report, we consider the specific case of deliberation dialogues, as defined in Walton (1984); Walton and Krabbe (1995) and we investigate how corrective feedback, in this type of dialogue, can be generated upon problems in the Common Ground, namely when inconsistencies in the Common Ground occur (§ 2). Specifically, we propose the use of graph databases as an integrated solution to dialogue state tracking, knowledge representation, and conflict detection as a fundamental building block for dialogue systems with argumentation capabilities (§ 3).

2 Common Ground Inconsistencies

With *Common Ground Inconsistencies* we refer to the incompatibility between the listener belief and the new evidence provided by the speaker. Given a domain D, we define a set of sequential actions A as a number of different a_i. Each a_i is associated with a set of states S_i composed of verifiable pre-conditions s_pre and post-conditions s_post. D is inconsistent when an action a_i exists, associated with its S_i, where either s_pre and/or s_post are incompatible with respect to the S set belonging to another a_j, as they cannot co-exist. When this conflict takes place, an inconsistency occurs. This conflict can depend on i) a s_pre which is incompatible with the rules of the Communal Common Ground (CCG)[1] (i.e., *cut the milk*), ii) the incompatibility of s_pre of the current a with s_post resulting from a preceding a, saved in the set of shared knowledge - the Personal Common Ground (PCG)[2]. Clarification requests can be in this case adopted as a corrective feedback.

[1] The amount of information shared with people that belong to the same community (Clark, 2015)

[2] The amount of information collected over time through communicative exchanges with an interlocutor(Clark, 2015)

Proceedings of the 25th Workshop on the Semantics and Pragmatics of Dialogue, September 20–22, 2021, Potsdam / The Internet.

3 Conflict Search Graph

The Conflict Search Graph allows to represent dialogue history and connect it with domain knowledge, so that CG stability checks and dialogue state tracking can be represented in the form of graph queries. From a formal point of view, dialogue states are defined by extending the concept of D as a sequence of actions. The aim of this module is to have a structured resource where the domain knowledge (part of the CCG) is stored, and whose conflict search module can be used to signal which input does not respect the rules of the CCG and cannot become part of the PCG. In fact, the graph is not just used to represent the domain and its rules: it also supports the automatic process of recognising Common Ground Inconsistencies. In other words, it is used to store the dialogue history so that inconsistencies caused by post-conditions applied by previous actions guide the identification of the potential source of the current inconsistency. *Preconditions* of an action describe the configurations of the CG that are compatible with action instancing. On the other hand, *post-conditions* are the graph updates applied after an action has been accepted in the PCG. When a post-condition resulting from a previous action clashes with a pre-condition of the current action an inconsistency occurs and a responsible action can be identified. Whereas the pre-conditions make aware of the possible presence of a conflict, the post-conditions help identify the conflicting action. The consistency checking process guides the adoption of linguistic feedback, such as Clarification Requests.

This module is represented as a (Neo4j-based (Webber, 2012)) graph[3] $D = \langle V, E \rangle$ where V is a set of vertices and E is a set of edges among the vertices in V. Edges are defined as functions between v_1 and v_2 where $v_1, v_2 \in V$. The edge is assumed to be oriented from v_1 to v_2. A *stable* CG is defined as a graph G where a set of stability checks, also based on frames pre-conditions, are all verified. A new candidate action to be included in the CG can be defined as a tuple $X = \langle a_n \rangle, < \bar{N}, \bar{E} >$ containing a new action a_n, a set of named entities $\bar{N}$ and a set of new edges $\bar{E}$. At any given time t, G_t represents the common ground configuration at t. Updating G by accepting X means creating a new graph $G' =< V', E' >$ where $V' = V \cup a_n \cup \bar{N}$ and $E' = E \cup \bar{E}$. G', can be accepted as an updated version of G only if G' is stable, so that:

$$G_{t+1} = G' \; if \; stable(g') \; else \; G$$

To verify that the Conflict Search Graph structure could actually detect inconsistencies to be, consequently, properly signalled, dedicated tests were carried out. For the test, 20 cooking recipes were used, in the form of lists of actions. For instance, the action *melt butter in a pan* was represented in a frame-based structure (Baker et al., 1998), as follows:

$$Apply\_heat \quad Food : butter; Container : pan$$

In each recipe, an erroneous action was inserted. While the command itself is acceptable by the time it is presented, it prevents the acceptability of a command appearing at least five steps later in the recipe, thus raising a conflict. The conflict was found **85%** of the times. In fact, for 3 recipes out of 20 the expected conflict action did not correspond to the one selected by the system. Nevertheless, the system outcomes cannot be considered as proper mistakes, as the system choices have reasonable explanations. For the *Pancakes* recipe, the expected conflict corresponded to *melt butter in a pan*, where *butter* was entirely used because no quantity was specified. The conflict is, therefore, triggered when the action *put the butter in the pan* is received in input, as the butter is no longer available. Nonetheless, the conflict was found at *add milk and butter to the yolks*. In fact, the unavailability of the ingredient can also be caused by the action of adding *butter* to other ingredients. In fact, when the system must select only one conflicting action, the most recent one is chosen. These results proved that the system could analyse pre-conditions rules correctly in a real context of use, even providing alternative views about the potential problems than the ones expected at design time.

4 Conclusions

Promising preliminary results collected in a simulated interaction scenario showed the potentiality of the Conflict Search graph for finding Common Ground Inconsistencies in dialogue. Starting from here, our purpose is to extend the experimentation of such a module in a real interactive scenario and to generalise the application of the graph. Other types of conflicts will also be investigated.

[3] The graph was built using data coming from Wikidata and FrameNet to represent the knowledge domain, part of the CCG; the PCG is, on the other hand, represented by the list of communicated actions. These are incrementally stored in the graph after running consistency checking queries.

References

Jens Allwood, Joakim Nivre, and Elisabeth Ahlsén. 1992. On the semantics and pragmatics of linguistic feedback. *Journal of semantics*, 9(1):1–26.

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90.

Hendrik Buschmeier and Stefan Kopp. 2018. Communicative listener feedback in human-agent interaction: Artificial speakers need to be attentive and adaptive. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, pages 1213–1221, Richland, SC. International Foundation for Autonomous Agents and Multiagent Systems.

Eve V. Clark. 2015. Common ground. In *The Handbook of Language Emergence*, page 328–353. Wiley, Chichester, UK.

Herbert H. Clark. 1996. *Using Language*. Cambridge University Press, Cambridge, UK.

Herbert H. Clark and Susan E. Brennan. 1991. Grounding in communication. In *Perspectives on Socially Shared Cognition*, pages 222–233, Washington, DC, USA. American Psychological Association.

David R Traum. 1999. Computational models of grounding in collaborative systems. In *Psychological Models of Communication in Collaborative Systems-Papers from the AAAI Fall Symposium*, pages 124–131.

Douglas N Walton. 1984. Logical dialogue-games and fallacies.

Douglas N Walton and Erik CW Krabbe. 1995. *Commitment in dialogue: Basic concepts of interpersonal reasoning*. SUNY press.

Jim Webber. 2012. A programmatic introduction to neo4j. In *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity*, pages 217–218.