Why should I turn left?
Towards active explainability for spoken dialogue systems.

Vladislav Maraev
University of Gothenburg
vladislav.maraev@gu.se

Ellen Breitholtz
University of Gothenburg
ellen.breitholtz@gu.se

Christine Howes
University of Gothenburg
christine.howes@gu.se

Jean-Philippe Bernardy
University of Gothenburg
jean-philippe.bernardy@gu.se

Abstract

In this paper we argue that to make dialogue systems able to actively explain their decisions they can make use of enthymematic reasoning. We motivate why this is an appropriate strategy and integrate it within our own proof-theoretic dialogue manager framework based on linear logic. In particular, this enables a dialogue system to provide reasonable answers to why-questions that query information previously given by the system.

1 Introduction

Thus far, explainability in dialogue systems has been implemented to the extent of *passive* explainability, where the developer of a dialogue system evaluates a system's decisions based on analysing the system's internal state (either ML-based or rule-based). Systems themselves lack the capacity to explain their decisions and conclusions in the way humans can understand. We call such an ability *active* explainability.

Natural language dialogue is an efficient way of creating trust between users and systems, since it allows users to get answers to any doubts and questions they might have. "How do you know?" and "Why?" questions are especially important, and are answered by explaining the reasoning behind a claim or suggestion. Such explanations may be given at different levels of detail, and users may occasionally want more details than given by the system. Providing an explanation which is optimally adapted to the individual user's level of expertise and general background knowledge is a very difficult task. This can be made considerably easier if the user is allowed to ask for additional information, challenge the system with counter-arguments, and establish a sense of common ground with the system regarding relevant background information. This means there needs to be an *interaction* between user and system. The system can build trust with the user by answering questions about the claims and decisions made.

The paper is organised as follows. In Section 2 we further motivate the importance of reasoning capabilities for conversational reasoning. Section 3 discusses our approach to reasoning in natural human dialogue, and explores the notions of *topos* and *enthymeme* that can be employed in spoken dialogue systems. In Section 4 we briefly present our proof-thoretic framework for dialogue management, which we further extend with a rudimentary support of active explainability (Section 5). We conclude in Section 6, discussing further directions of relevant research.

2 Background

Current artificial conversational agents are severely limited in their ability to reason. This reduces their usefulness to relatively simple tasks that do not involve much (or any) reasoning, which means that a huge number of tasks are out of reach for them. In the following genuine example interaction (queried 14th January 2020) the system fails to provide a simple inference from known facts:

(1) User: How old is President Trump?
Siri: President Trump is 73 years old.
User: How old is President Macron?
Siri: President Macron is 42 years old.
User: Is President Trump older than President Macron?
Siri: Here is what I found on the web *(displays irrelevant web links about Brigitte Macron)*

By contrast, humans engaged in real dialogue can make more complicated inferences such as example 2 below. Lee conveys that he is well enough to play football but not well enough to go to school because football takes place outdoors. His father

Proceedings of the Conference on Reasoning and Interaction (ReInAct), pages 58–64
October 4-6, 2021.

Dave infers that if Lee is well enough to play football then he is well enough to go to school.

(2) Dave: . . . you're gonna be home from football until four, you gonna have your dinner, want a bath.
Lee: Yeah, but I might not go to school tomorrow.
Dave: Why?
Lee: Cos of my cough.
Dave: How can you play football and not go to school then?
Lee: Cos I was going out in the fresh air, I'm alright when I'm out in the fresh air.
Dave: So why aren't you going to school then?
Lee: I'm in the classroom all day dad. [BNC KBE 10554-10561]

While the state of the art in conversational agents does not yet reflect that of computational models of inference, the reasoning capabilities displayed by Dave and Lee in example 2 are beyond the current reach of both fields. This study aims at advancing the current state of research towards enabling artificial conversational agents to achieve such reasoning capabilities.

3 Reasoning in dialogue

3.1 Enthymemes and topoi

While much research has been dedicated to enabling computers to reason from premises to conclusions, *how people interactively reason in natural dialogue* is still poorly understood (despite some relevant work in psychology on the structure of argumentative dialogue; Rips, 1998). One reason for this is that reasoning in dialogue often involves non-logical common-sense inferences. We refer to such inferences as *enthymemes*. An enthymeme is an argument which relies on information and principles of reasoning accessible to the interlocutors. In his *Rhetoric* Aristotle relates the enthymeme to logic by calling an enthymeme a "rhetorical syllogism". However, an enthymeme differs from a syllogism in that its conclusion does not follow by necessity. Instead the enthymeme owes a lot to context and background knowledge, and is therefore negotiable and cancellable.

The dialogue in (2) involves a lot of enthymematic reasoning. For example, Lee's argumentation is based on an assumption that having a cough is a reason not to go to school, and Dave's argument that if Lee can play football, he can go to school, is warranted by the principle that if you cannot participate in one strenuous activity, you cannot participate in another. This kind of underpinning principles of reasoning have been discussed at length in the literature on rhetoric and argumentation (Toulmin, 2003, a.o.), going back to Aristotle, who called them *topoi* (sg. *topos*). Ducrot (1988) introduced the concept of topos as part of a theory of linguistic meaning and cohesion with discourse units perceived as being connected by topoi. On this view the set of topoi accessible to an individual is a set of resources at the disposal of a dialogue participant for producing and interpreting arguments. However, this set does not constitute a monolithic logical system, as found in traditional default logics, where a rule holds if there is no other rule that contradicts it.

This presents a problem for conversational AIs since in the resources of an agent there can be contradicting topoi, or topoi that lead to contradicting conclusions (Breitholtz, 2014). In addition to this, which topoi apply in a particular situation, and which topos takes precedence over another is relative to the context, including the agent itself. This is demonstrated in (2), where Dave evokes a topos that contradicts Lee's reason for not going to school, by using the enthymeme 'if someone is well enough to play football then they are well enough to go to school'. Thus, the pragmatic meaning conveyed by an enthymeme in relation to a listener may differ depending on which topos the listener accesses in the interpretation process. This ability to follow various strains of reasoning — including inconsistent ones -– seems to be a prerequisite for the complex type of interactive language understanding and problem solving that humans master so well but conversational AIs do not.

Additionally, NLI datasets do not reflect the importance of interaction in reasoning: there are currently no datasets for inference in dialogue corresponding to those for monological text. Breitholtz (2014) combines insights from conversation analysis and rhetoric with dialogue semantics formalised in Type Theory with Records (TTR; Cooper, 2005). Like TTR linear logic has the advantage of letting us model both utterance events and utterance types. This is crucial for analysing meta-communicative aspects of interaction.

3.2 Dialogue agents

As discussed above, current conversational agents are severely lacking in inferential capabilities. Nevertheless, it is easy to see the benefit of explainability and interactive reasoning even in simple everyday tasks such as in-vehicle navigation.

(3) Sys: Turn right at the next junction
Usr: Why?
Sys: There is an accident ahead causing delays. I suggest going through the city instead.
Usr: How long is the delay?
Sys: About 15 minutes.
Usr: Hmm...
Sys: Based on similar previous routes and current city traffic, you would gain about 5 minutes.
Usr: Then I think I prefer to not take a right, I'm not in a hurry and it's not worth the trouble.
Sys: Got it.

In example (3) above, the system provides a motivation of the recommendation to make a right turn. After requesting additional information (some inferred *ad-hoc* by the system from a database of examples), the user makes an informed decision to not follow the system's suggestion. To accomplish participation in this kind of interaction, the system needs to (1) be able to answer "Why?" (and "How do you know?") questions to explain its suggestions and claims (explainability); (2) to draw conclusions from a database of information in a way that helps the user to draw relevant conclusions and make relevant decisions (interactive reasoning).

To enable explainable and interactive reasoning in conversational AI, systems need access to a vast store of topoi, encoding everyday principles of inference. In real-life settings, manually collecting these topoi is challenging, and such principles are also culturally conditioned and change over time. It would therefore be beneficial if systems could learn topoi from interaction with users, and in particular from observing enthymematic arguments. Systems therefore need to be endowed with dialogue strategies for eliciting such arguments from users, such as some clarification requests (CRs; Breitholtz, 2011).

(4) 1. **U:** Will you meet me in the drama studio?
2. **C:** The drama studio?
3. **U:** Yes, I've got an audition. (Ginzburg, 2012, p149)

In (4), C's CR elicits information that functions as a premise in the enthymematic argument "Let's meet in the drama studio, because I have an audition (so that's where I'll be)". If C can match this argument with topoi stored in her cognitive resources, she will make sense of what U is trying to communicate. This link between questions and enthymemes is also apparent in the case of why-questions (see (2)) (Jackson and Jacobs, 1980; Schlöder et al., 2016), which can be used to elicit a reason for the speaker saying something. Below is an (imagined) example where a system elicits and learns an enthymeme by asking "why?":

(5) Sys: Today you have a meeting at 8
Usr: Please cancel all my meetings today, I have to stay home
Sys: Why?
Usr: I have a cough

From the above (and similar interactions) the system learns "if one is sick, one should not attend work or school", and use this in later interactions.

(6) Sys: Do you want to hear your meetings for today?
Usr: I'm not feeling well, I think I have a fever
Sys: Okay, I take it you will stay home today. Should I cancel all your meetings?
Usr: Yes please

To accomplish the kinds of interactions we envisage, the dialogue agent needs be connected to one or more inference engines, allowing the system to (among other things) retrieve claims and suggested actions from a body of knowledge, together with (partial or full) motivations and justifications (sometimes in the form of proofs). Furthermore, as mentioned, they also need dialogue strategies enabling them to utilise inference engines in answering user questions (especially "Why?" and "How do you know?"-style questions), and preferably also mechanisms for learning relevant knowledge for use in future reasoning.

4 Dialogue manager architecture

We believe that it is crucial to use formal tools which are most appropriate for the task: one should

be able to express the rules of various genres of dialogue in a concise way, free, to any possible extent, of irrelevant technical details. In the view of Dixon et al. (2009) this is best done by representing the information-state of the agents as updatable sets of propositions. Very often, dialogue-managument rules update subsets (propositions) of the information state independently from the rest. A suitable and flexible way to represent such updates are as function types in linear logic. The domain of the function is the subset of propositions to update, and the co-domain is the (new) set of propositions which it replaces.

By using well-known techniques which correspond well with the intuition of information-state based dialogue management, we are able to provide a fully working prototype of the components of our framework:

1. a proof-search engine based on linear logic, modified to support inputs from external systems (representing inputs and outputs of the agent)

2. a set of rules which function as a core framework for dialogue management (in the style of KoS (Ginzburg, 2012))

3. several examples which use the above to construct potential applications of the system.

4.1 Linear rules and proof search

Typically, and in particular in the archetypal logic programming language prolog (Bratko, 2001), axioms and rules are expressed within the general framework of first order logic. However, several authors (Dixon et al., 2009; Martens, 2015) have proposed using linear logic (Girard, 1995) instead. For our purpose, the crucial feature of linear logic is that hypotheses may be used *only once*.

In general, the linear arrow corresponds to *destructive state updates*. Thus, the hypotheses available for proof search correspond to the *state* of the system. In our application they will correspond to the *information state* of the dialogue participant.

In linear logic, normally firing a linear rule corresponds to triggering an *action* of an agent, and a complete proof corresponds to a *scenario*, i.e. a sequence of actions, possibly involving action from several agents. However, the information state (typically in the literature and in this paper as well), corresponds to the state of a *single* agent. Thus, a scenario is conceived as a sequence of actions and updates of the information state of a single agent a, even though such actions can be attributed to any other dialogue participant b. (That is, they are a's representation of actions of b.) Scenarios can be realised as a sequence of actual actions and updates. That is, an action can result in sending a message to the outside world (in the form of speech, movement, etc.). Conversely, events happening in the outside world can result in extra-logical updates of the information state (through a model of the perceptory subsystem).

In our work we are employing an information-state update (ISU) approach, following several authors, including Larsson (2002) and Ginzburg (2012). We treat the information state as a multiset of *linear hypotheses* that can be queried. Because they are linear, these hypotheses can also be removed from the state. In particular, we have a fixed set of rules (they remain available even after being used). Each such rule manipulates a part of the information state (captured by its premises) and leaves everything else in the state alone.

4.2 Questions and answers

The essential components of the representation of a question are a type A, and a predicate P over A. Using a typed intuitionistic logic, we write:

$$A : Type \qquad\qquad P : A \rightarrow Prop$$

The intent of the question is to find out about a value x of type A which makes $P\ x$ true, or at least entertained by the other participant.

We make use of metavariables to represent what is being asked, as the unknown in a proposition. Within the state of the agent, if the value of the requested answer is represented as a metavariable x, then the question can be represented as: $Q\ A\ x\ (P\ x)$. That is, the pending question (Q denotes a question constructor) is a triple of a type, a metavariable x, and a proposition where x occurs. We stress that $P\ x$ is *not* part of the information state of the agent yet, rather the fact that the above question is *under discussion* is a fact. For example, after asking "Where does John live?", we have a fact (we use the double colon for information-state facts and we assume that agent's questions under discussion are stacked in the $Cons$-list):

$$\_ :: QUD\ (Cons\ (Q\ Location\ x\ (Live\ John\ x))\ Nil)$$

Resolving a question can be done by communicating an answer. An answer to a question

$(A : Type; P : A \rightarrow Prop)$ can be of either of the two following forms: i) A **ShortAnswer**, which is a pair of an element $X : A$ and its type A, represented as $ShortAnswer\ A\ X$ or ii) An **Assertion** which is a proposition $R : Prop$, represented as $Assert\ R$. For a more detailed description we refer the reader to Maraev et al. (2020).

4.3 Dialogue management

Our DM models the information-state of only one participant. Regardless, this participant can record its own beliefs about the state of other participants. In general, the core of DM is comprised of a set of linear-logic rules which depend on the domain of application. However, many rules will be domain-independent (such as generic processing of answers). Here we will provide a few examples of the rules which are implemented in our system, and we refer our reader to Maraev et al. (2020) for more detailed description.

The following rule accounts for pushing the content (a question) of any received Ask move on top of the stack of questions under discussion (QUD).

$$
\begin{aligned}
&pushQUD : (x\ y : DP) \rightarrow \\
&\quad (q : Question) \rightarrow \\
&\quad (qs : List\ Question) \rightarrow \\
&\quad Pending\ (Ask\ q\ x\ y) \multimap \\
&\quad QUD\ qs \multimap QUD\ (Cons\ q\ qs)
\end{aligned}
$$

After the question has been integrated, if the system has a fact p which answers the question in its database it can produce an answer.[1]

$$
\begin{aligned}
&produceAnswer : \\
&\quad (a : Type) \rightarrow (x : a) \rightarrow (p : Prop) \rightarrow \\
&\quad (qs : List\ Question) \rightarrow \\
&\quad QUD\ (Cons\ (Q\ USER\ a\ x\ p)\ qs) \multimap \\
&\quad p \multimap \\
&\quad [\ \_ :: Agenda\ (ShortAnswer \\
&\qquad a\ x\ SYSTEM\ USER); \\
&\quad\ \_ :: QUD\ qs; \\
&\quad\ \_ :: Answered\ (Q\ USER\ a\ x\ p)]
\end{aligned}
$$

Note: taking a linear argument and producing it again is a common pattern, which can be spelled out $A \multimap [\_ :: A; \_ :: P]$. It is so common that from here on we use the syntactic sugar $A \multimap P$ for it.

[1] or, possibly, a domain-specific clarification request depending on whether the fact is unique and concrete, see the work of Maraev et al. (2020) for further details

5 Extending the framework with enthymematic reasoning

In this section we will describe a rudimentary support for enthymematic reasoning within the framework described above. It appears to be easier to discuss the extension of the system with a simple example, adapted from Breitholtz (2020).

(7)

1	U:	How can I get home?
2	S:	Via the bypass.
3	U:	Why the bypass?
4	S:	Because the route is the shortest.

For dealing with lines 1–2 of example (7), let's assume that system has an access to the following facts from the knowledge base which represent three possible routes to home, via three different roads.

$$
\begin{aligned}
&\_ :: Route\ Bypass\ Home; \\
&\_ :: Route\ ParkLane\ Home; \\
&\_ :: Route\ BridgeRoute\ Home;
\end{aligned}
$$

Assuming that the question under discussion is $(Q\ USER\ Road\ x\ (Route\ x\ Home))$, and the choice of hypothesis is pseudo-random, we can see that x unifies with any of three facts, therefore using the $produceAnswer$ rule, system can produce a short answer $(ShortAnswer\ Road\ Bypass\ SYSTEM\ USER)$, which can be realised as "Via the bypass".

Now let us turn to the argumentative part of the dialogue. We would need to use a domain specific representation of a question by adding an additional predicate: $(Q\ USER\ Road\ x\ (Pick\ (Route\ x\ Home)))$. The knowledge base can also be extended with some additional facts about the qualities of the routes.

$$
\begin{aligned}
&\_ :: Shortest\ (Route\ Bypass\ Home); \\
&\_ :: Cheapest\ (Route\ ParkLane\ Home); \\
&\_ :: Prettiest\ (Route\ BridgeRoute\ Home);
\end{aligned}
$$

To represent enthymematic reasoning (lines 3–4 of (7)), we will introduce the reasoning pattern represented by following rule:

$$
\begin{aligned}
&toposShortest : (x : Road) \rightarrow (y : To) \rightarrow \\
&\quad (qs : List\ Question) \rightarrow \\
&\quad QUD\ (Cons\ (Q\ USER\ Road\ x \\
&\qquad (Pick\ (Route\ x\ y)))\ qs) \multimap \\
&\quad Route\ x\ y \multimap
\end{aligned}
$$

$$
\begin{array}{l}
Shortest\ (Route\ x\ y) \rightarrowtriangle \\
\quad [\_ :: Pick\ (Route\ x\ y); \\
\quad\ \_ :: Topos\ (Shortest\ (Route\ x\ y))\,;\,]\,;
\end{array}
$$

This can be read as follows: "In the context of a question under discussion, involving picking a route, pick the shortest one, and remember why it was picked". The latter is represented in the last line and alludes to the salient *topos* used for this choice. Note, that here we leave destination underspecified, and further underspecifications are possible: not only shortest routes might be preferred but also shortest times or sentences.

Following Breitholtz (2020) we treat why-questions as questions asking for a topos, which becomes the question under discussion $(Q\ \ USER\ \ Reason\ \ t\ \ (Topos\ \ t))$ where t is a metavariable representing the reason for choosing the bypass. With a local topos produced by $toposShortest$ rule at hand we can apply the standard $produceAnswer$ rule, which would elicit a short answer:

$$
\begin{array}{l}
ShortAnswer\ Reason \\
\quad (Shortest\ (Route\ Bypass\ Home)) \\
\quad\ SYSTEM\ \ USER
\end{array}
$$

It can be realised as an utterance "Because it is the shortest", concluding our example (7).

Our system supports several competing topoi, for instance we can analogously add $toposPrettiest$ and $toposCheapest$ rules. Assuming random selection of an applicable rule, the system will be able to offer a justification for whichever of the routes it chooses, based on the underlying topos.

6 Discussion

Currently, if the system has access to multiple topoi, the choice of which topos the system should prioritise when making a suggestion is made at random. Even if the suggestion is not the one preferred by the user, being given a reason based on an acceptable topos will enable the user to evaluate the suggestion and decide whether or not to go along with it. However, a useful extension would be to add a probabilistic component. For example, this would enable a system to learn that a particular user ranks certain topoi higher when making various decisions, thus adapting the topoi on which to base instructions according to that user's profile. One of the ways to achieve this would be to represent user profiles as vector of real values for each preference and have a vector real-valued information about the qualities of the item (e.g., characteristics of each route). Then, we compute a score for each item as the sum of item's qualities, weighted by user's preferences for each quality and select the one with the best score. For the selection the topos can be recorded (in the simplest case as a fact like "the route is shorter which corresponds to your preference"). Later, during the course of the dialogue, the topoi can be challenged and user preferences can be updated accordingly using Bayesian updates. We leave the detailed description of this process for future work.

In future investigations, it should be possible to assume that the publicised enthymemes and their corresponding topoi become a part of the information-state, allowing further discussion and, possibly, re-ranking of topoi. It should also be possible to learn topoi directly from the user, for instance, in the form of requests such as "if the caller is a VIP, remind me to call back within 1 hour".

In the future we plan to assess the trust placed in a system by users under different conditions, in order to find out if the reliability of a system that uses topoi is rated higher by users than one that does not (for instance, using subjective evaluation methods, like SASSI, Hone and Graham, 2000), and how much adapting the ranking of topoi to a specific user's profile contributes to building trust in the system on the part of the user.

Acknowledgments

This work was supported by the Swedish Research Council (VR) grant 2014-39 for the establishment of the Centre for Linguistic Theory and Studies in Probability (CLASP). Howes and Breitholtz were additionally supported by the VR grant 2016-0116, Incremental Reasoning in Dialogue. We would like to thank Bill Noble and our anonymous reviewers for their insightful comments.

References

Ivan Bratko. 2001. *Prolog programming for Artificial Intelligence*. Pearson education.

Ellen Breitholtz. 2011. Enthymemes under Discussion: Towards a micro-rhetorical approach to dialogue modelling. In *Proceedings of SPR-11 ILCLI International Workshop on Semantics, Pragmatics, and Rhetoric*.

Ellen Breitholtz. 2014. Reasoning with topoi–towards a rhetorical approach to non-monotonicity. In *Proceedings of the 50th Anniversary Convention of the AISB*.

Ellen Breitholtz. 2020. *Enthymemes and Topoi in Dialogue: The Use of Common Sense Reasoning in Conversation*. Brill, Leiden, The Netherlands.

Robin Cooper. 2005. Records and record types in semantic theory. *Journal of Logic and Computation*, 15(2):99–112.

Lucas Dixon, Alan Smaill, and Tracy Tsang. 2009. Plans, actions and dialogues using linear logic. *Journal of Logic, Language and Information*, 18(2):251–289.

Oswald Ducrot. 1988. Topoï et formes topiques. *Bulletin d'Etudes de Linguistique Francaise*, 22(1):1–14.

Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*. Oxford University Press, Oxford.

Jean-Yves Girard. 1995. *Linear Logic: Its syntax and semantics*, London Mathematical Society Lecture Note Series, page 1–42. Cambridge University Press.

Kate S Hone and Robert Graham. 2000. Towards a tool for the subjective assessment of speech system interfaces (sassi).

Sally Jackson and Scott Jacobs. 1980. Structure of conversational argument: Pragmatic bases for the enthymeme. *Quarterly Journal of Speech*, 66(3):251–265.

Staffan Larsson. 2002. *Issue-based dialogue management*. Ph.D. thesis, University of Gothenburg.

Vladislav Maraev, Jean-Philippe Bernardy, and Jonathan Ginzburg. 2020. Dialogue management with linear logic: the role of metavariables in questions and clarifications. *Traitement Automatique des Langues (TAL)*, 61(3):43–67.

Chris Martens. 2015. *Programming Interactive Worlds with Linear Logic*. Ph.D. thesis, Carnegie Mellon University Pittsburgh, PA.

Lance J. Rips. 1998. Reasoning and conversation. *Psychological review*, 105(3):411.

Julian Schlöder, Ellen Breitholtz, and Raquel Fernández. 2016. Why? In *Proceedings of JerSem*, pages 5–14.

Stephen E. Toulmin. 2003. *The Uses of Argument*. Cambridge University Press.