On the differences between human agents and logic-based software agents discourse understanding*

Adrian Groza[1[0000−0003−0143−5631]]

Technical University of Cluj-Napoca, Cluj-Napoca, Romania
http://users.utcluj.ro/∼agroza
Adrian.Groza@cs.utcluj.ro

Abstract. We are interested in the differences between how a human agent and a logic-based software agent interpret a text in natural language. When reading a narrative, the human agent has a single interpretation model. That is the preferred model among the models consistent with the available information. The model is gradually adjusted as the story proceeds. Differently, a logic-based software agent works with a finite set of many models, in the same time. Of most interest is that the number of these models is huge, even for simple narratives. We compare here the reduction strategies of humans and software agents to keep the discourse more intelligible and tractable. One the one hand, the human agent extensively uses common knowledge, contextual reasoning and closes the world as much as possible. On the other hand, the logical agent adds domain knowledge (such as ontologies) and applied reduction strategies (such as identifying isomorphisms). The differences are analyse with puzzles in First order logic, Description logic and Dynamic epistemic logic.

Keywords: machine comprehension · discourse understanding · interpretation models · logical agents · logical puzzles

1 Interpretation models in First Order Logic

Let the classic love story between Abelard and Heloise, with the text *Abelard and Heloise are in love*. The human agent interpretation is that there are two individuals Abelard (a) and Heloise (h) that love each other (see Fig. 1). Instead, for the logical agent, the number of these models is huge, even for such simple narratives. The variety of interpretation models depends on at least two factors: i) the errors and ambiguities introduced during natural language processing (NLP) or ii) the way in which the interpretation are built based on the resulted formalisation of the text.

First, assume during natural language processing, the statement is interpreted as *Abelard is in love and Heloise is in love*. The formalisation in First Order Logic is:

* This research is part-funded by the ExNanoMat-21PFE grant.

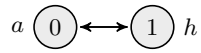


Fig. 1. The unique interpretation model of the human agent.

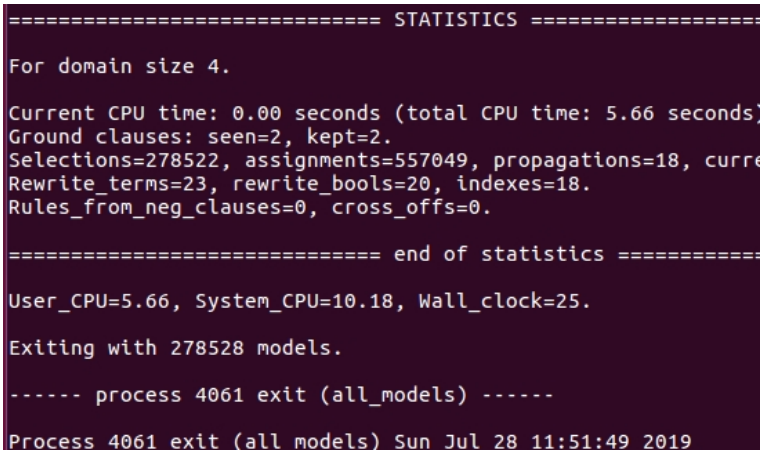
$$A_1\colon \exists x,\ love(abelard, x)$$
$$A_2\colon \exists x,\ love(heloise, x)$$

This formalisation is explained by the fact that NLP is based on statistical analysis. Based on statistics, our sentence will follow the same pattern as: *Abelard and Heloise are in park* (that is $inPark(abelard) \wedge inPark(heloise)$ or *Abelard and Heloise are in happy* ($happy(abelard) \wedge happy(heloise)$).

Second, we are interested how many models does a FOL-based model finder compute for axioms A_1 and A_2? To answer this question we played with the MACE4 [10]. First, we closed the domain to 4 individuals (see Listing 1.1). Figure 2 illustrates the output of MACE4: there are 278,528 models.

Listing 1.1. Finding models with domain closed to 4 individuals.

```
assign(max_models, -1).
assign(domain_size, 4).
formulas(assumptions).
   exists x love(abelard,x).
   exists x love(heloise,x).
end_of_list.
```



```
============================ STATISTICS ==================

For domain size 4.

Current CPU time: 0.00 seconds (total CPU time: 5.66 seconds)
Ground clauses: seen=2, kept=2.
Selections=278522, assignments=557049, propagations=18, curre
Rewrite_terms=23, rewrite_bools=20, indexes=18.
Rules_from_neg_clauses=0, cross_offs=0.

============================ end of statistics ===========

User_CPU=5.66, System_CPU=10.18, Wall_clock=25.

Exiting with 278528 models.

------ process 4061 exit (all_models) ------

Process 4061 exit (all_models) Sun Jul 28 11:51:49 2019
```

Fig. 2. MACE4 finds 278,528 interpretation models (domain is closed to four individuals only).

As these variety of models of unexpected (recall that the domain was restricted to four individuals), we took a look at the generated models (see Fig. 3). Here, a stands for *abelard*, h for *heloise*, while c_1 and c_2 are the Skolem constants generated for the existential quantifiers in A_1 and A_2. As the domain is closed to four individuals we work only with the set of integers $\{0, 1, 2, 3\}$. The first model (first row, left) is consistent with the human interpretation: *abelard* and *heloise* do love each other. Note also that all four individuals are distinct: $a \to 0$, $b \to 1$, $c_1 \to 2$, $c_2 \to 3$. In the second model (first row, center), abelard loves an individual c_1, while heloise loves a distinct individual c_2. In the third model (first row, right), both abelard and heloise love the same individual c_1. Moreover, no one sad that the *love* relation is not reflexive. One such model is the fourth one (second row, left), where both abelard and heloise love each other. The variety of the models is also increased by different possible love relations involving c_1 and c_2. For instance in the fourth model, c_1 loves heloise and c_2 loves c_1. Similarly, no one sad that someone can love only one person at the same time. Therefore, the fifth model (second row, center) is possible. Here, abelard loves both heloise and c_1. The largest influence is given by the fact that the logical agent can interpret that some individuals are not distinct. In the sixth model (second row, right) abelard and heloise are interepreted as the same individual ($a \to 0$, $h \to 0$) referred by two distinct names.

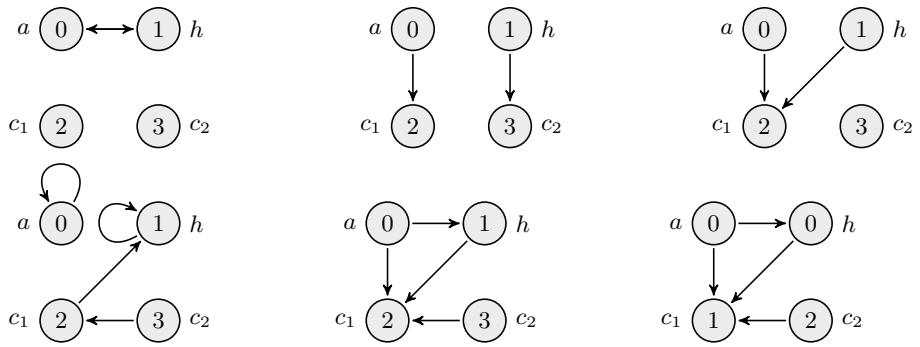


Fig. 3. Sample of interpretation models for the software agent.

The above analysis explains how the logical agent has indeed 278,528 interpretation models for the simple sentence *Abelard and Heloise are in love*. Moreover, all these models are equally plausible for the software agent. Given this gap (278,528 models vs. one model of the human agent), the natural question is *How the two agents would understand each other?* Our approach is to reduce the number of interpretation models for the software agent.

2 Reducing the interpretation models of the logical agent

To reduce the number of interpretation models, the knowledge base of the logical agent should be extended with several constraints.

First, the unique name assumption (UNA) can be added. In the MACE4 case, this is explicitly added with

$$A_3 : abelard \neq heloise$$

Models like the sixth one in Fig. 3 are removed. Note that we cannot apply this assumption on the Skolem constants that are generated during the FOl theory is processed. Under UNA, there are still 163,840 remaining models.

Second, we can assume that the love relation is not narcissistic. That is

$$A_4 : \forall x,\ \neg love(x, x)$$

With this constraint models like the fouth one in Fig. 3 are removed, leading to 5,120 remaining models.

Third, we add the somehow strong constraint that someone can love only one person at a time. That is

$$A_5 : love(x, y) \wedge love(x, z) \rightarrow y = z$$

Models like the fifth one in Fig. 3 are removed. The remaining models are 80. Unfortunately, love is not a symmetric relation. Hence, we cannot add the axiom $\forall x, y\ love(x, y) \leftrightarrow love(y, x)$.

Forth, we can exploit the fact that some of these models are isomorphic. After applying the MACE4's algorithm to remove isomorphic models [10], we keep 74 non-isomorphic models.

Fifth, recall that there are 2 Skolem constants after converting axioms A_1 and A_2. If we are not interested in the love relations of individuals represented by these constants, we can ignore them. This would result in 17 models obtained with the extended knowledge base from Fig 2.

Some observations follow.

First, the order in which we apply the reductions is computationally relevant. For instance, it would be prohibitively to search for isomorphic models in the initial two steps, when there are 278,528 or 163,840 models. Mace4 reduces the initial 278,528 models to 186,976 non-isomorphic models in a User CPU time of $\sim$2 hours. Hence, the above strategy was to add domain knowledge to the initial narrative discourse, and then to search for the isomorphic structures.

Second, which domain knowledge to add is subject to interpretation. For instance, axiom A_5 might be too strong. There are various contexts, in which someone can love more than one individual in the same time. There are some contexts in which the human agent would have as the interpretation model, the second model in Fig. 3). We argue that the decision what domain knowledge to activate should rely on some contextual reasoning step.

Third, the interpretation models vary as the story evolves. Let the following statement in the story:

Abelard and Heloise are in love. They are getting married.

This statement has the same pattern as:

Abelard and Heloise are in park. They are playing chess.

Assume the translation from natural language to FOL is the one in Listing 1.2. Note that we have already included here the domain knowledge to reduce the number of models: domain is closed to 4 individuals (line 2); UNA is applied on the named individuals (line 6); love is not narcissistic (line 7); love is a functional relation (line 8); the anaphora is correctly identified by the translator - that is the pronoun *they* is correctly replaced by Abelard and Heloise (lines 9 and 10); one person cannot married to him/herself (line 11); each person can be married with maximum one person at the same time (line 12). Given the above restrictions, MACE4 computes 5,242,880 models.

Listing 1.2. Increasing number of models as the story evolves. There are 5.242.880 models for the theory below.

```
1  assign(max_models, -1).
2  assign(domain_size, 4).
3  formulas(assumptions).
4     exists x love(abelard,x).
5     exists x love(heloise,x).
6     abelard != heloise.
7    all x -love(x,x).
8    love(x,y) & love (x,z) -> y = z.
9    exists married(abelard,x).
10   exists married(heloise,x).
11   all x -married(x,x).
12   married(x,y) & married (x,z) -> y = z.
13 end_of_list.
```

If one wants to assure to the logical agent the same view of the story as the human agent, the burden seems to be on the NLP to FOL translator. A theory that is closer to the human interpretation is the one in Listing 1.3. Here the main advantage is that existential quantifiers do not appear and thus the domain can be closed to 2 individuals only.

Listing 1.3. FOL theory closed for the human model.

```
1  assign(max_models, -1).
2  assign(domain_size, 2).
3  formulas(assumptions).
4     love(abelard,heloise).
5     love(heloise,abelard).
6     abelard != heloise.
```

```
7     all x -love(x,x).
8     love(x,y) & love (x,z) -> y = z.
9     married(abelard,heloise).
10    married(heloise,abelard).
11    all x -married(x,x).
12    married(x,y) & married (x,z) -> y = z.
13 end_of_list.
```

Here we focused on restricting interpretation models given a statistical-based translation from NLP to FOL. For some reasoning tasks the aim is indeed to have a single interpretation model of a narrative. One example is when specifying commands or tasks to a robot. Let the command: *Bring two espresso coffees to Abelard and Heloise.* In order to assure the correct interpretation, the agent should compute a single model for this command. We are interested next, in which situations when reducing models is not required.

3 When more models are better?

For some reasoning tasks (e.g. solving lateral thinking puzzles [3]) keeping all possible models might be desirable. Let the following puzzle:

> *Two American Indians were sitting on a log - a big Indian and a little Indian. The little Indian was the son of the big Indian, but the big Indian was not the father of the little Indian. How do you explain this?*

Most of the people are able to quickly figure out the solution. However, the online forums indicate that there are human agent having difficulties to identify an interpretation model.

The logical agent does not have difficulties to compute the one interpretation model. Let the formalisation in Description Logic (DL) from Listing **??**. We picked DL as it is easier to import domain knowledge as ontologies are available on the Web. The family ontology is particularly useful here[1]. Relevant here is that the concept $Father$ is disjoint to $Mother$. Also, in line 9, the relation $hasSon$ is included in the more general relation $hasChild$. That is, if two individuals are related through the $hasSon$ relation, the logical agent infers that they are also related through the $hasChild$ relation. Additionaly to this terminological box ((lines 1 to 9)), the information from puzzle is formalised in the assertional box in lines 10-13. Here, $littleIndian$ and $bigIndian$ are instances of the concept $AmericanIndian$ (lines 10-11). The $bigIndian$ has son the $littleIndian$ (line 12), while $bigIndian$ is not an instance of the $Father$ concept. Given the above knowledge to reasoner in DL (such as Racer [6]), the system is able to infer that $bigIndian$ is an instance of the $Mother$ concept.

[1] We assume the the reader is familiar with the Description Logic syntax. Otherwise, the reader is referred to [1].

1 $Indian \sqsubseteq Person$
2 $Woman \sqsubseteq Person$
3 $Man \sqsubseteq Person$
4 $Parent \sqsubseteq \exists hasChild.Person$
5 $Father \equiv Man \sqcap Parent$
6 $Mather \equiv Woman \sqcap Parent$
7 $Parent \sqsubseteq Father \sqcap Mother$
8 $Father \sqsubseteq \neg Mather$
9 $hasSon \sqsubseteq hasChild$
10 $hasDaughter \sqsubseteq hasChild$
11 $littleIndian : AmericanIndian$
12 $bigIndian : AmericanIndian$
13 $(bigIndian, littleIndian) : hasSon$
14 $bigIndian : \neg Father$

This is one example in which the human agent might not have an interpretation model, while the logical agent has one. A more difficult lateral thinking puzzle for the human agent is the following one:

Two girls are born to the same mother, on the same day, in the same month and year and yet they're not twins. How can this be?

The problem is that the human agent closes the world too much. What is relevant here to add to the family ontology are the axioms for the $Twin$ or $Triplet$ concept. Based on assertion that the girls are not Twin, the Racer reasoner deduces that they are triplets.

The problem here is that the human agent closed the world two much. The human agent fails to consider models in which a third individual exists. Differently, the software agent reasons here under the Open World Assumption.

Other examples in which the software agent is more aware of current world come from the epistemic puzzles. Let the following one:

There are 3 logicians at a table in a pub, The waitress asks them: "Does everyone want beer?" The first logician answers "I don't know". The second logician answers "I don't know". The third logician answer "Yes".

A logical agent, based on the possible worlds semantics of the Kripke structures [12] does not have problems to gradually reduce the interpretation models to the correct one.

Initially, there are 8 possible models, depending on which logician wants beer or not. Let b_i true if logician i wants beer, and false otherwise. Each world is characterised by three propositional variables: b_1, b_2 and b_3:

$$\{\langle 0,0,0\rangle, \langle 0,0,1\rangle, \langle 0,1,0\rangle, \langle 0,1,1\rangle, \langle 1,0,0\rangle, \langle 1,0,1\rangle, \langle 1,1,0\rangle, \langle 1,1,1\rangle\}$$

After the public answer of the first agent, all the models in which he does not want beer ($b_1 = 0$) are eliminated. That is because if he would not want beer the answer would have been "No". The words in which b_1 is false are eliminated for remaining four possible cases:

$$\{\langle 1,0,0\rangle, \langle 1,0,1\rangle, \langle 1,1,0\rangle, \langle 1,1,1\rangle\}$$

Similarly, after the second answer, all the agents will know that the second logician also wants beer. Otherwise, his answer would have been "No". Hence, the models with $b_2 = 0$ are eliminated:

$$\{\langle 1,1,0\rangle, \langle 1,1,1\rangle\}$$

Now the third agent can figure in which world there are. As he wants beer, the only possible model is $\{\langle 1,1,1\rangle\}$. This model is conveyed to waitress[2]

4 Discussion and related work

Both the human reader and the software agent aim to keep the story more intelligible and tractable. But they apply different reduction strategies. On one hand, humans understand stories by inferring the mental states (e.g. motivations, goals) of the characters, by applying parabolic projections of known stories into the target narrative [7], by extensively using commonsense reasoning [11] and fuzzy reasoning [9], or by closing the world as much as possible. On the other hand, logic-based software agents reduce the models by formalising discourse representation theories [8], by adding domain knowledge, or by identifying isomorphisms.

We also exemplified here how the number of interpretation models vary as the story evolves. Sentences introducing new objects and relations do increase the number of models. Sentences introducing constraints on the existing objects and relations contribute to the removal of some models. Adding domain knowledge also contributes to model removal. One research question is how to generate stories that end with a single interpretation model for the software agent. Another issue regards the amount of domain knowledge and commonsense knowledge that should be added, and which reduction strategy is better when the aim is to keep the number of models computationally feasible.

We focused here on the model explosion in logical frameworks and not on the translation from natural language into some logical formalism. Tools like Fred [4] aim to automatically translate natural language to description logic. Still, there are very limited. Given our sentence *Abelard and Heloise in love*, Fred translation[3] identifies *love* as an individual, not a relation: $in(abelard, love) \wedge in(heloise, love)$ (see Fig. 4). Note that Fred faccilitate contextual reasoning by

[2] For one implementation of this puzzle, the interested reader is referred to SMCDEL symbolic model checker for Dynamic Epistemic Logic (https://github.com/jrclogic/SMCDEL) [2].

[3] http://wit.istc.cnr.it/stlab-tools/fred/demo/

correctly identify the characters from DBpedia: Peter Abelard and the abbess Heloise. Another tool that aim to translate natural language to FOL is NLTK. NLTK has an interface with MACE4 and Prover9[4] to perform logical inference and model building of the translated FOL knowledge [5].

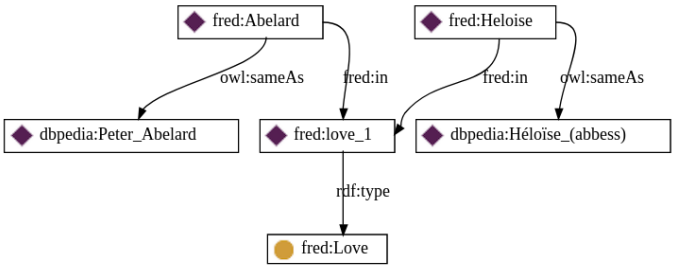


Fig. 4. Automatic translation from natural language into description logic.

We noticed that text models built with machine learning applied on big data, would benefit from some crash diet. In this line, we try to extract as much as we can from each statement, instead of statistically analysing the entire corpus. That is, the model of the story is built bottom-up and not top-down as machine learning does.

5 Conclusion

We compared the reduction strategies of humans and software agents to keep the discourse more intelligible and tractable. One the one hand, the human agent extensively uses common knowledge, contextual reasoning and closes the world as much as possible. On the other hand, the logical agent adds domain knowledge (such as ontologies) and applied reduction strategies (such as identifying isomorphisms).

For most of the reasoning tasks, the human agent keeps only one interpretation model. In this case, the aim to reduce the interpretation models of the software agent as much as possible in order to facilitate communication between human and software agent. In case of puzzles, some human agents fails to have an interpretation. Differently, the logical agent is able to compute a consistent model with the given knowledge, if such a model exists.

References

1. Baader, F., Calvanese, D., McGuinness, D., Patel-Schneider, P., Nardi, D.: The description logic handbook: Theory, implementation and applications. Cambridge university press (2003)

[4] http://www.nltk.org/howto/inference.html

2. van Benthem, J., van Eijck, J., Gattinger, M., Su, K.: Symbolic model checking for dynamic epistemic logic—s5 and beyond. Journal of Logic and Computation **28**(2), 367–402 (2017)
3. De Bono, E., Zimbalist, E.: Lateral thinking. Viking (2010)
4. Gangemi, A., Presutti, V., Recupero, D.R., Nuzzolese, A.G., Draicchio, F., Mongiovì, M.: Semantic Web Machine Reading with FRED. Semantic Web **8**(6), 873–893 (2017)
5. Garrette, D., Klein, E.: An extensible toolkit for computational semantics. In: proceedings of the eighth international conference on computational semantics. pp. 116–127. Association for Computational Linguistics (2009)
6. Haarslev, V., Hidde, K., Möller, R., Wessel, M.: The racerpro knowledge representation and reasoning system. Semantic Web **3**(3), 267–277 (2012)
7. Herman, D.: Story logic: Problems and possibilities of narrative. U of Nebraska Press (2004)
8. Kamp, H., Reyle, U.: From discourse to logic: Introduction to model theoretic semantics of natural language, formal logic and discourse representation theory, vol. 42. Springer Science & Business Media (2013)
9. Letia, I.A., Groza, A.: Modelling imprecise arguments in description logic. Advances in Electrical and Computer Engineering **9**(3), 94–99 (2009)
10. McCune, W.: Mace4 reference manual and guide. arXiv preprint cs/0310055 (2003)
11. Mueller, E.T.: Commonsense reasoning: an event calculus based approach. Morgan Kaufmann (2014)
12. Van Ditmarsch, H., van Der Hoek, W., Kooi, B.: Dynamic epistemic logic, vol. 337. Springer Science & Business Media (2007)