Kalpa Publications in Computing
Volume 21, 2025, Pages 8–14
Proceedings of the 14th and 15th International Workshops on the Implementation of Logics

A Chat with Bard*

Geoff Sutcliffe[1], Jack McKeown[1], and Alexander Steen[2]

[1] University of Miami, USA
geoff@cs.miami.edu,jam771@miami.edu
[2] University of Greifswald, Germany
alexander.steen@uni-greifswald.de

Abstract

Given all the excitement around ChatGPT, Bard, Bing, etc., we decided to test Bard's logical reasoning powers. Bard was given the English description of the TPTP problem PUZ001+1 - "Who Killed Aunt Agatha?", modified to really ask that question. This short paper documents the story that unfolded, and how tools from the TPTP World were used to examine the output from Bard.

1 The Initial Question and Answer

Given all the excitement around ChatGPT, Bard, Bing, etc., we decided to test Bard's logical reasoning powers. Bard was given the English description of the TPTP [11] problem PUZ001+1 - "Who Killed Aunt Agatha?"[1], modified to really ask that question. Here's what was given as the prompt ...

> *Someone who lives in Dreadbury Mansion killed Aunt Agatha. Agatha, the butler, and Charles live in Dreadbury Mansion, and are the only people who live therein. A killer always hates his victim, and is never richer than his victim. Charles hates no one that Aunt Agatha hates. Agatha hates everyone except the butler. The butler hates everyone not richer than Aunt Agatha. The butler hates everyone Aunt Agatha hates. No one hates everyone. Agatha is not the butler. Agatha is not Charles. Charles is not the butler. Who killed Aunt Agatha?*

... and here's what Bard replied ...

> *Let's analyze the information given step by step:*
>
> 1. *Agatha, the butler, and Charles live in Dreadbury Mansion, and are the only people who live therein.*
> 2. *A killer always hates his victim, and is never richer than his victim.*

*Presented at the 14th International Workshop on the Implementation of Logics [12].
[1] https://www.tptp.org/cgi-bin/SeeTPTP?Category=Problems&Domain=PUZ&File=PUZ001+1.p

> 3. *Charles hates no one that Aunt Agatha hates.*
> 4. *Agatha hates everyone except the butler.*
> 5. *The butler hates everyone not richer than Aunt Agatha.*
> 6. *The butler hates everyone Aunt Agatha hates.*
> 7. *No one hates everyone.*
> 8. *Agatha is not the butler.*
> 9. *Agatha is not Charles.*
> 10. *Charles is not the butler.*
>
> *From point 4, we know Agatha hates Charles. Since Charles hates no one that Aunt Agatha hates (point 3), Charles does not hate Aunt Agatha. Thus, Charles cannot be the killer since a killer hates his victim. From points 5 and 6, we know the butler hates everyone not richer than Aunt Agatha and everyone that Aunt Agatha hates. Since Agatha hates everyone except the butler (point 4), the butler hates Charles. According to point 7, no one hates everyone, so the butler doesn't hate Agatha. The only person left is Aunt Agatha. Since we know that Charles and the butler cannot be the killers, and Agatha is not Charles and Charles is not the butler, it implies that Agatha must have killed herself.*

So, the answer was right, but it's not hard to see that the reasoning is wrong.

1.1 Encoding Bard's Reply

Bard's reasoning was encoded in the TPTP format for derivations [13], as follows ...

```
%----Someone who lives in Dreadbury Mansion killed Aunt Agatha.
fof(a0,axiom,
    ? [X] : ( lives(X) & killed(X,agatha) ) ).

%----1. Agatha, the butler, and Charles live in Dreadbury Mansion,
fof(a1_1,axiom,
    ( lives(agatha) & lives(butler) & lives(charles) ) ).

%----and are the only people who live therein.
fof(a1_2,axiom,
    ! [X] : ( lives(X) => ( X = agatha | X = butler | X = charles ) ) ).

%----2. A killer always hates his victim,
fof(a2_1,axiom,
    ! [X,Y] : ( killed(X,Y) => hates(X,Y) ) ).

%----and is never richer than his victim.
fof(a2_2,axiom, ! [X,Y] : ( killed(X,Y) => ~ richer(X,Y) ) ).

%----3. Charles hates no one that Aunt Agatha hates.
fof(a3,axiom, ! [X] : ( hates(agatha,X) => ~ hates(charles,X) ) ).

%----4. Agatha hates everyone except the butler.
fof(a4,axiom, ! [X] : ( X != butler => hates(agatha,X) ) ).

%----5. The butler hates everyone not richer than Aunt Agatha.
fof(a5,axiom, ! [X] : ( ~ richer(X,agatha) => hates(butler,X) ) ).

%----6. The butler hates everyone Aunt Agatha hates.
fof(a6,axiom, ! [X] : ( hates(agatha,X) => hates(butler,X) ) ).
```

```
%----7. No one hates everyone.
fof(a7,axiom, ! [X] : ? [Y] : ~ hates(X,Y) ).

%----8. Agatha is not the butler.
fof(a8,axiom, agatha != butler ).

%----9. Agatha is not Charles.
fof(a9,axiom, agatha != charles ).

%----10. Charles is not the butler.
fof(a10,axiom, charles != butler ).

%----From point 4, we know Agatha hates Charles.
fof(i1,plain,
    hates(agatha,charles),
    inference(bard,[status(thm)],[a4,a10]) ).

%----Since Charles hates no one that Aunt Agatha hates (point 3), Charles
%----does not hate Aunt Agatha.
fof(i2,plain,
    ~ hates(charles,agatha),
    inference(bard,[status(thm)],[a3,i1]) ).

%----Thus, Charles cannot be the killer since a killer hates his victim.
fof(i3,plain,
    ~ killed(charles,agatha),
    inference(bard,[status(thm)],[a2_1,i2]) ).

%----From points 5 and 6, we know the butler hates everyone not richer than
%----Aunt Agatha and everyone that Aunt Agatha hates. Since Agatha hates
%----everyone except the butler (point 4), the butler hates Charles.
fof(i4,plain,
    hates(butler,charles),
    inference(bard,[status(thm)],[a5,a6,a4,a10]) ).

%----According to point 7, no one hates everyone, so the butler doesn't hate
%----Agatha.
fof(i5,plain,
    ~ hates(butler,agatha),
    inference(bard,[status(thm)],[a7,i4]) ).

%----The only person left is Aunt Agatha. Since we know that Charles and the
%----butler cannot be the killers
fof(i6,plain,
    ~ killed(butler,agatha),
    inference(bard,[status(thm)],[i5,a2_1]) ).

%----Agatha is not Charles and Charles is not the butler, it implies
%----that Agatha must have killed herself.
fof(i7,plain,
    killed(agatha,agatha),
    inference(bard,[status(thm)],[i3,i6,a9,a10,a0,a1_1,a1_2]) ).
```

Some minor adaptations of Bard's output were justified:

- For `i1`, `a10` is used but not mentioned in Bard's text.
- As is explained in Section 1.2, `i2` is unsound. The correct inference is `~hates(charles,charles)`.
- For `i4`, `a10` is used but not mentioned in Bard's text, while `a5` is mentioned in Bard's text but not needed for the inference.

- Related to `i5`, `~hates(butler,butler)` can be derived from `a4`, `a6`, and `a7`.
- For `i7`, `a0`, `a1_1`, and `a1_2` are not mentioned in Bard's text, but are implicitly necessary.

There is an interesting human inductive bias in `a7`, which contributes to the wrong conclusion of `i5` (see Section 1.2), that "no one hates everyone" is interpreted by humans (and maybe Bard) as "no one hates everyone else". The axiom could be modified to reflect that ...

```
fof(a7,axiom, ! [X] : ? [Y] : ( X != Y & ~ hates(X,Y)) ).
```

... but that makes the axioms contradictory.

1.2 Analysis with TPTP World Tools

As a first step the derivation from Section 1.1 was displayed using the IDV derivation viewing tool [16], as shown in Figure 1.

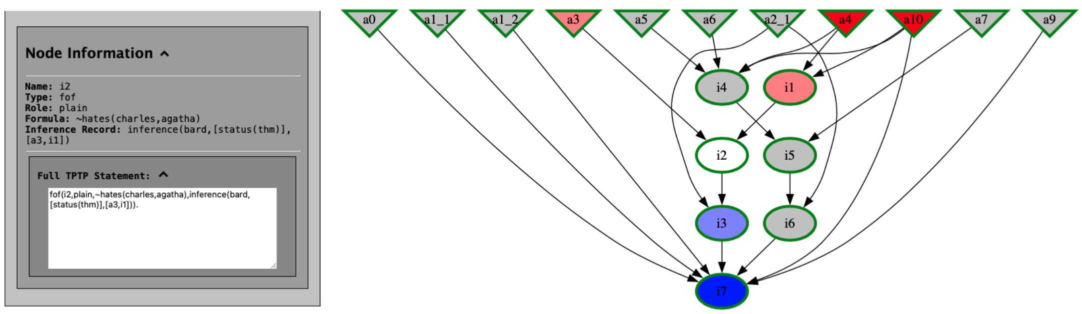

Figure 1: Visualization of the derivation

That didn't make the reasoning mistakes obvious, so the GDV derivation verification tool [10] was used to isolate the unsound inferences ...

```
SUCCESS: Derivation has unique formula names
SUCCESS: All derived formulae have parents and inference information
SUCCESS: Derivation is acyclic
SUCCESS: Assumptions are propagated
SUCCESS: Assumptions are discharged
 RESULT: axioms.sat_model.dis - Paradox---4.0 says Satisfiable - CPU = 0.00
SUCCESS: Leaf axioms are satisfiable
 RESULT: i1.thm.dis - Otter---3.3 says Theorem - CPU = 1.80
SUCCESS: i1 is a thm of a4 a10
 RESULT: addi1.thm.dis - Otter---3.3 says Theorem - CPU = 1.85
SUCCESS: addi1 is a thm of a3 i1
 RESULT: i2.thm.dis - Otter---3.3 says GaveUp - CPU = 1.87
 RESULT: i2.thm.dis - Paradox---4.0 says CounterSatisfiable - CPU = 0.00
FAILURE: i2 is not a thm of a3 i1
 RESULT: i3.thm.dis - Otter---3.3 says Theorem - CPU = 1.82
SUCCESS: i3 is a thm of a2_1 i2
 RESULT: i4.thm.dis - Otter---3.3 says Theorem - CPU = 1.85
SUCCESS: i4 is a thm of a10 a5 a6 a4
 RESULT: i5.thm.dis - Otter---3.3 says GaveUp - CPU = 1.88
 RESULT: i5.thm.dis - Paradox---4.0 says CounterSatisfiable - CPU = 0.00
FAILURE: i5 is not a thm of a7 i4
 RESULT: i6.thm.dis - Otter---3.3 says Theorem - CPU = 1.92
SUCCESS: i6 is a thm of i5 a2_1
 RESULT: i7.thm.dis - Otter---3.3 says Theorem - CPU = 1.86
```

```
SUCCESS: i7 is a thm of i3 i6 a9 a10 a0 a1_1 a1_2
CPUTIME: 18.44
FAILURE: Not verifiedSZS status NotVerified
```

The conclusions *Charles does not hate Aunt Agatha* (`i2`) and *the butler doesn't hate Agatha* (`i5`) are wrong (`FAILURE`).

2 Giving Bard More Chances

Bard was given a chance to redeem itself, by giving the prompt ...

> *What is the logical mistake in your reasoning?*

... which produced ...

> *From point 4, we know Agatha hates Charles. Since Charles hates no one that Aunt Agatha hates (point 3), Charles does not hate Aunt Agatha. Thus, Charles cannot be the killer since a killer hates his victim. From points 5 and 6, we know the butler hates everyone not richer than Aunt Agatha and everyone that Aunt Agatha hates. Since Agatha hates everyone except the butler (point 4), the butler hates Charles. According to point 7, no one hates everyone, so the butler doesn't hate Agatha. Thus, the butler cannot be the killer since a killer hates his victim. Now, considering Agatha's perspective, we know from point 4 that Agatha hates everyone except the butler. Since no one hates everyone (point 7), Agatha doesn't hate herself. The logical mistake in my previous reasoning was assuming that Agatha could be the killer despite the fact that she didn't hate herself. Given the new analysis, there is a contradiction in the information provided, as there is no valid solution to the problem.*

It would be simple enough to encode Bard's fallacious excuse in TPTP format, view the derivation in IDV, and use GDV to find the mistakes, but that's a job to be automated in the future.

The Aunt Agatha problem is well known, with many solutions that Bard could have read on the web. As a test for any reasoning capability in Bard, the problem was obfuscated as follows (*Splot* is *Agatha*, *Splat* is *Charles*, *Split* is the *butler*, *move diagonally in the plain* is *lives in Dreadbury Mansion*, *shmutzes* is *hates*, *smaller* is *richer*, and *bamboozles* is *kills*) ...

> *Splot, Split, and Splat move diagonally in the plain, and are the only things that move in such a way. Something that moves diagonally in the plain bamboozled Splot. A thing that bamboozles always shmutzes its target, and is never smaller than its target. Splat shmutzes nothing that Splot shmutzes. Splot shmutzes everything except Split. Split shmutzes everything not smaller than Splot. Split shmutzes everything Splot shmutzes. Nothing shmutzes everything. Splot is not Split, Split is not Splat, Splat is not Splot. Who bamboozled Splot?*

... and here's what Bard replied ...

> *From the given information, we can conclude that:*
>
> - *Split is not smaller than Splot.*

- *Split shmutzes Splot.*
- *Therefore, Split bamboozled Splot.*

Unsurprisingly, Bard got it wrong. GDV revealed that the first conclusion that *Split is not smaller than Splot* is unsound, and that is required for the final conclusion that *Split bamboozled Splot*. (The second conclusion that *Split shmutzes Splot* is sound.) This is further evidence for what erudite folks (think they) know . . . tools like Bard do not (yet) include formal reasoning capability.

3 Conclusion

There are well understood reasons why generative AI should not be expected to give sound answers to reasoning tasks like the one described and analysed in this paper. Other people might not know that danger, and might trust such AI tools to give sound answers. There's an opportunity here for Automated Reasoning . . . in the last decade many researchers have been developing ways to use machine learning to guide the actions (axiom selection, given clause selection, lemma retention, etc.) of Automated Theorem Provers, e.g., [6, 7, 5, 2, 8]. The toy experiment described above exemplifies what some people in the community, e.g., [17, 1, 15] have been saying for quite a while . . . (i) symbolic reasoning systems should be usefully integrated in complex reasoning systems; (ii) symbolic reasoning systems should be used to verify and point out errors in results produced by subsymbolic systems. One approach to (ii) for generated text output would be to produce the output in a Controlled Natural Language [9], e.g., Attempto Controlled English (ACE) [4], and convert that to logic to verify the reasoning in the output [3, 14]. Or as a 10 year old might cry out . . .

"Machine Learning Drools! Logic Rules!"

References

[1] C. Benzmüller and B. Lomfeld. Reasonable Machines: A Research Manifesto. In U. Schmid, F. Klügl, and D. Wolter, editors, *Proceedings of the 43rd German Conference on Artificial Intelligence*, number 12325 in Lecture Notes in Computer Science, pages 251–258, 2020.

[2] M. Crouse, B. Abdelaziz, I. Makni, S. Whitehead, C. Cornelio, P. Kapanipathi, K. Srinivas, V. Thost, M. Witbrock, and Fokoue A. A Deep Reinforcement Learning Approach to First-Order Logic Theorem Proving. In K. Leyton-Brown and Mausam, editors, *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, volume 35(7), pages 6279–6287. AAAI Press, 2021.

[3] N. Dellis. Using Controlled Natural Language for World Knowledge Reasoning. Master's thesis, University of Miami, Miami, USA, 2010.

[4] N. Fuchs, K. Kaljurand, and T. Kuhn. Attempto Controlled English for Knowledge Representation. In C. Baroglio, P. Bonatti, J. Maluszynski, M. Marchiori, A. Polleres, and S. Schaffert, editors, *Proceedings of Reasoning Web: 4th International Summer School*, number 5224 in Lecture Notes in Computer Science, pages 104–124. Springer-Verlag, 2008.

[5] J. Jakubuv, K. Chvalovský, M. Olsák, B. Piotrowski, M. Suda, and J. Urban. ENIGMA Anonymous: Symbol-Independent Inference Guiding Machine (System Description). In N. Peltier and V. Sofronie-Stokkermans, editors, *Proceedings of the 10th International Joint Conference on Automated Reasoning*, number 12167 in Lecture Notes in Artificial Intelligence, pages 448–463, 2020.

[6] C. Kaliszyk, J. Urban, and J. Vyskocil. Machine Learner for Automated Reasoning 0.4 and 0.5. In S. Schulz, L. de Moura, and B. Konev, editors, *Proceedings of the 4th Workshop on Practical*

Aspects of Automated Reasoning, number 31 in EPiC Series in Computing, pages 60–66. EasyChair Publications, 2015.

[7] S. Loos, G. Irving, C. Szegedy, and C. Kaliszyk. Deep Network Guided Proof Search. In T. Eiter and D. Sands, editors, *Proceedings of the 21st International Conference on Logic for Programming, Artificial Intelligence, and Reasoning*, number 46 in EPiC Series in Computing, pages 14–30. EasyChair Publications, 2017.

[8] J. McKeown and G. Sutcliffe. Reinforcement Learning for Guiding the E Theorem Prover. In A. Ae Chun and M. Franklin, editors, *Proceedings of the 36th International FLAIRS Conference*, 2023. DOI: 10.32473/flairs.36.133334.

[9] R. Schwitter. Controlled Natural Languages for Knowledge Representation. In C-R. Huang and D. Jurafsky, editors, *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1113–1121. Coling 2010 Organizing Committee, 2010.

[10] G. Sutcliffe. Semantic Derivation Verification: Techniques and Implementation. *International Journal on Artificial Intelligence Tools*, 15(6):1053–1070, 2006.

[11] G. Sutcliffe. The TPTP Problem Library and Associated Infrastructure. From CNF to TH0, TPTP v6.4.0. *Journal of Automated Reasoning*, 59(4):483–502, 2017.

[12] G. Sutcliffe, J. McKeown, and A. Steen. A Chat with Bard. In S. Schulz, K. Korovin, and M. Rawson, editors, *Proceedings of the 14th International Workshop on the Implementation of Logics*, number 10329 in EasyChair Preprints. EasyChair Publications, 2023.

[13] G. Sutcliffe, S. Schulz, K. Claessen, and A. Van Gelder. Using the TPTP Language for Writing Derivations and Finite Interpretations. In U. Furbach and N. Shankar, editors, *Proceedings of the 3rd International Joint Conference on Automated Reasoning*, number 4130 in Lecture Notes in Artificial Intelligence, pages 67–81. Springer, 2006.

[14] G. Sutcliffe, M. Suda, A. Teyssandier, N. Dellis, and G. de Melo. Progress Towards Effective Automated Reasoning with World Knowledge. In C. Murray and H. Guesgen, editors, *Proceedings of the 23rd International FLAIRS Conference*, pages 110–115. AAAI Press, 2010.

[15] O. Sychev. Combining Neural Networks and Symbolic Inference in a Hybrid Cognitive Architecture. *Procedia Computer Science*, 190:728734, 2021.

[16] S. Trac, Y. Puzis, and G. Sutcliffe. An Interactive Derivation Viewer. In S. Autexier and C. Benzmüller, editors, *Proceedings of the 7th Workshop on User Interfaces for Theorem Provers*, volume 174 of *Electronic Notes in Theoretical Computer Science*, pages 109–123, 2007.

[17] Y. Yang and L. Song. Learn to Explain Efficiently via Neural Logic Inductive Learning. In S. Mohamed, D. Song, K. Cho, and M. White, editors, *Proceedings of the 8th International Conference on Learning Representations*, 2020.