

Abduction に於ける欠如仮説節生成手法

6 G-1

阿部 明典

NTT コミュニケーション科学研究所

1 はじめに

一般に仮説推論 [2] 等の abduction では知識ベースに仮説が欠如していると推論を行なうことが出来ない。従って、欠如している仮説の生成が必要となる。それに対して、CMS [3] は欠如している節を生成可能なシステムである。しかし、観測側から abduction することにより仮説を求めているので、知識ベースに存在する節で構成される推論木の最後にくつつく節しか生成出来ない。つまり、観測の説明に用いることの出来るリーフからゴールに向かうことの出来る節集合が知識ベースにあってもそれは使われず、短絡的な節を生成してしまう。例えば、3 節の例では、CMS を用いた場合、得られる minimal support の否定(仮説の候補)は {"east.", "ride.", "car\fuel.", "horse.", "village.", "fuel\town."} となってしまう。つまり、"desert" は全く反映されていない結果が得られる。それを防ぐために、deductive にリーフから節を連結し、abductive に観測から連結した節との間を繋ぐことで仮説を生成することが考えられるが、節が欠如している場合、どの節が繋がるかの情報が皆無であるし、over-abduction、over-deduction の可能性もある。従って、本論文に示す手法は推論木の途中に欠如している節の候補を abductive、deductive に推論木を辿ることで探し、更に、知識ベースに含まれる仮説となるべき節の類似節を参照することで節を生成するものである。従って、リーフから観測に向かうことが出来る節もしくは、節の連結が存在する場合、それに deductive に繋がる節を新たに生成して観測を説明することが出来る。

2 仮説節の生成

本節では abduction、deductin により節を生成するアルゴリズムを示す。基本的には abduction、deduction 双方により絞り込みを行い、その結果を参考にして知識ベースの節から写像することにより、欠如節の生成を行なう。従って、以下の3つのフェーズに分けて行なう。

◇ Abduction phase

このフェーズでは abductive に仮説を探す。このフェーズで仮説が見つければ、普通の仮説推論と同じである。

1. ユーザから質問(観測)が与えられると、知識ベース内の節で観測が説明可能か調べる。観測が説明出来た場合は、観測が説明出来たとして終了する。できなかった場合は、以下のステップへ。
2. 観測から推論木を abductive に辿り、知識ベース内の節により支持が得られない、もし

くは、次に連結すべき節が見つからない直前まで節を abductive に連結する。

3. 得られた推論木の最後につく(リーフ側)節、推論木の上にある atom のうち、知識ベース内の節による支持のない節に含まれる atom を記録する。

◇ Deduction phase

このフェーズでは deductive に節を連結し、それにより、前フェーズで仮説による支持のなかった節を繋ぐべき節を求める。

4. 知識ベース内の atom 節全てから観測方向に deductive に節を連結し、先に節を連結出来ないか、もしくは、知識ベース内の節の支持が得られない直前まで節を連結する。
5. 得られた推論木の最後につく(観測側)節、推論木の上にある atom のうち、知識ベース内の節による支持のある節に含まれる節に含まれる atom のうち、リーフ位置にない atom を記録する。

◇ Hypotheses generation phase

このフェーズでは前2つのフェーズで得られた結果をもとに、仮説節を求める。

6. 前 Step で記録された atom の類似 atom を知識ベースに含まれる節に含まれる atom より求める。

類似 atom が見つからない場合は、類似節を探せる可能性はないので、Abduction phase にて仮説の支持がなかった節において、それがあれば支持されるという仮説を示し、これが正当化されれば観測は説明されると示して終了する。この場合は、CMS と同じである。類似 atom が見つかった場合は、その類似 atom が同じ節の上もしくは、deductive に連結する節に含まれるかどうか調べる。それらが同じ節の上もしくは、deductive に連結する節の上にはない場合は Step 7 へ。それらが同じ節の上もしくは、deductive に連結する節の上にある場合は Step 10 へ。

7. 類似 atom が節の head 部にある時は、その節が知識ベースの節により支持されれば観測が説明されたことになる。その節が知識ベースの節により支持された場合は、Step 8 へ。支持されない場合は、支持すべき仮説が正当化されれば観測は説明可能であると示して終了。

類似 atom が節の body 部にある時は、Step 9 へ。

8. Step 7 で見つかった節に対応する節(類似語を含む他の節)を変換し、それを仮説として観測が説明できると示して終了。
9. Step 7 で見つかった節を変換し、変換された節が知識ベースの中にある節により支持された場合は、それを仮説として観測が説明できると示して終了。支持されない場合は、支持するのに必要な節が正当化されれば、変換された節と支持するのに必要な節を仮説として観測を説明可能であると示して終了。
10. Step 6 で見つかった節を変換し、変換された節が知識ベースの中にある節により支持された場合は、それを仮説として観測が説明できると示して終了。支持されない場合は、支持するのに必要な節が正当化されれば、変換された節と支持するのに必要な節を仮説として観測を説明可能であると示して終了。

Step 8, 9, 10 における“変換”とは、例えば、 a と a' が類似しているとした場合、“ $a : b.$ ”と“ $d : a', c.$ ”があった時に、“ $d : a', c.$ ”を“ $d : a, c.$ ”に変換することなどが考えられる。こうすることにより、“ $a : b.$ ”が推論に使用可能となり、より好ましい仮説の生成を行なうことが出来る。

3 推論例

<i>beautiful</i> :- <i>palace.</i>	<i>palace</i> :- <i>east.</i>	
<i>east</i> :- <i>ride.</i>	<i>west</i> :- <i>ride.</i>	<i>saddle.</i>
<i>ride</i> :- <i>horse, saddle.</i>	<i>lovely</i> :- <i>donkey.</i>	
<i>donkey</i> :- <i>desert.</i>	<i>ride</i> :- <i>car, fuel.</i>	
<i>car</i> :- <i>city.</i>	<i>horse</i> :- <i>village.</i>	<i>desert.</i>

Figure 1: 知識ベースの例

この節では、前節で述べたアルゴリズムを簡単な例で具体的に示す。

知識ベースとして Figure 1 に示されるものがあるとする。ユーザが“*palace*”と質問したとする。質問から abductive に推論木を辿っても知識ベースに含まれる節ではこの質問を説明できないので、仮説の支持が得られなくなる所、もしくは、次に連結すべき節が見つからない所まで、知識ベース内の節を abductive に連結する。この例の場合、

{“*palace* :- *east*”, “*ride* :- *saddle, horse.*”, “*saddle.*”, “*horse* :- *village.*”, “*ride* :- *fuel, car.*”, “*car* :- *town.*”} が abductive な推論木の上に乗る。そして、以上の推論木の中で最後につく節、{“*horse* :- *village.*”, “*car* :- *town.*”} を記録する。

次に、推論木のリーフから仮説の支持が得られないか、先に節を連結出来ない所まで deductive に知識ベース内の節を繋げる。この場合、{“*desert.*”, “*saddle.*”, “*donkey* :- *desert.*”, “*lovely* :- *donkey.*”, “*donkey* :- *saddle, horse.*”} が deductive な推論木の上に乗る。そして、これらの節のうち、最後につく節、

{“*lovely* :- *donkey.*”, “*ride* :- *saddle, horse.*”} を記録する。更に、abductive に辿った推論木の上にある atom のうち、知識ベース内の仮説による支持のない節に含まれる節に含まれる atom {*ride, saddle, horse, east, village, car, town, fuel*} を集め、記録し、deductive に辿った推論木の上にある atom のうち、知識ベース内の仮説による支持がある節に含まれる節に含まれる atom のうち、リーフ位置にない atom {*lovely, donkey*} を集め、記録する。以上、記録された atom のうち、知識ベース内の節に含まれる atom と類似したものがあるかどうか調べると、*horse* \approx *donkey* が見つかり、“*ride* :- *saddle, horse.*”と“*donkey* :- *saddle.*”がお互いの類似節の候補になる。この場合は、“*desert.*”が知識ベースにあり、“*donkey*”は“*donkey* :- *desert.*”の head 部にあるので、“*desert* :- *donkey.*”, “*desert.*”をリーフに繋がる仮説と見ることが出来、それに対応する“*ride* :- *saddle, horse.*”がそれに繋がるべき節の類似節であるとみなせる。一方、尚、“*ride* :- *saddle, horse.*”は類似 atom を body 部に含む節でもあるが、その節は知識ベースの中の節により支持されない。従って、“*ride* :- *saddle, horse.*”はそのままでは観測を説明出来る節である可能性はないとみなせる。従って、“*desert* :- *donkey.*”, “*desert.*”に繋がるはずの“*ride* :- *saddle, horse.*”を“*desert* :- *donkey.*”, “*desert.*”に繋がるように類似節に変換して、“*ride* :- *saddle, donkey.*”とし、“*ride* :- *saddle, donkey.*”, を仮説として観測 *palace* を説明できると示す。

CMS では“*desert*”にいる“*donkey*”を考慮にいれた仮説を生成出来なかったが、提案する手法では、知識ベースにそれなりの節が含まれていれば、“*desert*”にいる“*donkey*”を考慮にいれた仮説を生成可能である。

4 まとめ

本論文では推論木の途中にあるべき欠如節を生成することを目的とした仮説生成手法を示した。これは、AAR[1]に於ける仮説生成をターゲットにしたものであり、AAR に利用した CMS の拡張となる。従って、前節の例で示したように CMS の枠組では得られない節が本手法では得られる。例えば、1 節で示したように、CMS の枠組では“*desert*”を反映した仮説は得られないが、提案する手法では得られる。但し、本手法は節の方向も (head と body) も利用するので、CMS と違って節の形態は Prolog-type の節に限ったので、少し自由度は減るかも知れない。

References

- [1] 阿部 明典: *Abductive Analogical Reasoning*, 人工知能学会全国大会, S6-04, pp (123)-(126) (1996)
- [2] 國藤 進: 仮説推論, 人工知能学会誌, Vol.2-1, pp. 22-29 (1986)
- [3] Reiter R. and de Kleer J.: *Foundation of assumption-based truth maintenance systems: preliminary report*, Proc. of AAAI87, pp. 183-188 (1987)