Data Science Academy

www.datascienceacademy.com.br

Introdução à Inteligência Artificial

Extraindo Regras Gerais a Partir de Exemplos

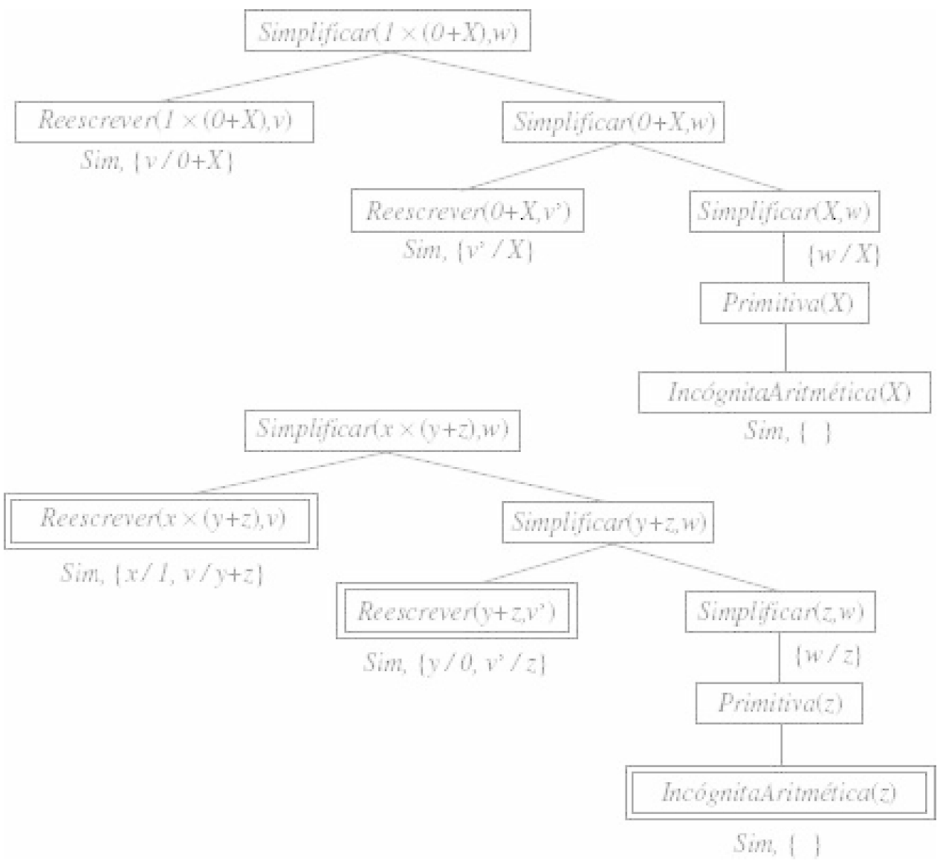
A técnica de memoização tem sido usada há longo tempo em ciência da computação para acelerar programas salvando os resultados da computação. A ideia básica das funções de memo é acumular um banco de dados de pares de entrada/saída; quando a função é chamada, primeiro ela examina o banco de dados para ver se pode evitar resolver o problema desde o início. A aprendizagem baseada na explanação tira bom proveito dessa etapa, criando regras gerais que cobrem uma classe inteira de casos. Este recurso é chamado de aprendizagem baseada na explanação (ABE).

A ideia básica por trás da aprendizagem baseada na explanação é primeiro construir uma explanação da observação usando o conhecimento a priori e depois estabelecer uma definição da classe de casos para os quais a mesma estrutura de explanação pode ser usada. Essa definição fornece a base para uma regra que cobre todos os casos na classe. A “explanação” pode ser uma prova lógica, mas, de modo mais geral, pode ser qualquer processo de raciocínio ou de resolução de problemas cujos passos sejam bem definidos. A chave é ser capaz de identificar as condições necessárias para que os mesmos passos se apliquem a outro caso.

Usaremos como nosso sistema de raciocínio o provador simples de teoremas de encadeamento para trás. A árvore de prova para Derivada(X_2, X) = 2X é muito grande para ser usada como exemplo e, portanto, usaremos um problema mais simples para ilustrar o método de generalização. Suponha que nosso problema seja simplificar 1 × (0 + X). A base de conhecimento inclui as regras a seguir:

$$
\begin{aligned}
&Reescrever(u, v) \land Simplificar(v, w) \Rightarrow Simplificar(u, w) \\
&Primitiva(u) \Rightarrow Simplificar(u, u)\ . \\
&IncógnitaAritmética(u) \Rightarrow Primitiva(u)\ . \\
&Número(u) \Rightarrow Primitiva(u)\ . \\
&Reescrever(1 \times u, u)\ . \\
&Reescrever(0 + u, u). \\
&\vdots
\end{aligned}
$$

A prova de que a resposta é X é mostrada na metade superior da figura abaixo. Na realidade, o método de ABE constrói duas árvores de prova simultaneamente. A segunda árvore de prova utiliza um objetivo variabilizado, no qual as constantes do objetivo original são substituídas por variáveis. À medida que a prova original prossegue, a prova variabilizada prossegue no mesmo passo, usando exatamente as mesmas aplicações de regra. Isso poderia fazer algumas das variáveis se tornarem instanciadas. Por exemplo, para usar a regra Reescrever(1 × u, u), a variável x no subobjetivo Reescrever(x × (y + z), v) deve estar vinculada a 1. De modo semelhante, y deve estar vinculada a 0 no subobjetivo Reescrever(y + z, v’), a fim de utilizar a regra Reescrever(0 + u, u). Uma vez que temos a árvore de prova generalizada, tomamos as folhas (com as vinculações necessárias) e formamos uma regra geral para o predicado objetivo:



Em geral, as condições podem ser descartadas da regra final se elas não impõem nenhuma restrição sobre as variáveis do lado direito da regra, porque a regra resultante ainda será verdadeira e será mais eficiente. Note que não podemos descartar a condição IncógnitaArtimética(z) porque nem todos os valores possíveis de z são incógnitas aritméticas. Valores que não são incógnitas aritméticas talvez exijam formas diferentes de simplificação: por exemplo, se z fosse 2 × 3, a simplificação correta de 1 × (0 + (2 × 3)) seria 6 e não 2 × 3. Para recapitular, o processo básico de ABE funciona assim:

1. Dado um exemplo, construa uma prova de que o predicado objetivo se aplica ao exemplo usando o conhecimento prático disponível.

2. Em paralelo, construa uma árvore de prova generalizada para o objetivo variabilizado, utilizando os mesmos passos de inferência da prova original.

3. Construa uma nova regra cujo lado esquerdo consista nas folhas da árvore de prova e cujo lado direito seja o objetivo variabilizado (depois da aplicação das vinculações necessárias a partir da prova generalizada).

4. Descarte quaisquer condições do lado esquerdo que sejam verdadeiras independentemente dos valores das variáveis no objetivo.

A árvore de prova generalizada da árovore de decisão acima realmente gera mais de uma regra generalizada. Por exemplo, se encerrarmos, ou podarmos, o crescimento da ramificação da direita na árvore de prova quando ela alcançar o passo Primitiva, obteremos a regra:

$$Primitiva(z) \Rightarrow Simplificar(1 \times (0 + z), z)$$

Embora mais geral, essa regra é tão válida quanto a regra que utiliza IncógnitaAritmética, porque cobre os casos em que z é um número. Podemos extrair uma regra ainda mais geral efetuando a poda depois do passo Simplificar(y + z, w) gerando a regra:

$$Simplificar(y + z, w) \Rightarrow Simplificar(1 \times (y + z), w)$$

Em geral, uma regra pode ser extraída de qualquer subárvore parcial da árvore de prova generalizada. Agora, temos um problema: qual dessas regras escolheremos? A escolha de qual regra gerar se reduz à questão da eficiência. Existem três fatores envolvidos na análise de ganhos de eficiência a partir de ABE:

1. A adição de grande número de regras pode diminuir a velocidade do processo de raciocínio porque o mecanismo de inferência ainda tem de verificar essas regras, mesmo em casos nos quais elas não produzem uma solução. Em outras palavras, ela aumenta o fator de ramificação no espaço de busca.

2. Para compensar a redução da velocidade de raciocínio, as regras derivadas devem oferecer aumentos significativos em velocidade para os casos que elas abrangem. Esses aumentos surgem principalmente porque as regras derivadas evitam becos sem saída que de outro forma seriam seguidos, mas também porque elas encurtam a prova propriamente dita.

3. As regras derivadas devem ser tão gerais quanto possível, de forma que elas se apliquem ao maior conjunto possível de casos.

Referências:

Livro: Inteligência Artificial
Autor: Peter Norvig