

مدل سازی روابط و نسبت‌های خانوادگی در پرولوگ

استاد: جناب آقای دکتر فیضی درخشی

نویسنده: پارسا یوسفی نژاد محمدی

شماره دانشجویی: ۱۴۰۰۰۵۷۱۱۰۴۱

مقدمه

پرولوگ به عنوان یک زبان برنامه نویسی منطقی، قابلیت‌های منحصر به فردی در پیاده‌سازی و استخراج الگوهای داده‌های منطقی دارد، از این رو می‌توان از آن در پیاده‌سازی، ارزیابی و بررسی الگوهای رایج روابط خانوادگی و ارتباطات مرسوم بین اعضای یک خانواده و تحلیل آن‌ها پرداخت؛ در این زبان می‌توان به سادگی با ایجاد قوانین و حقایق گوناگون به کوئری‌های مرتبط با روابط خانوادگی پاسخ‌های دقیقی ارائه کنیم؛ به همین دلایل پرولوگ گزینه بسیار مناسبی جهت پیاده‌سازی چنین پروژه‌هایی است.

هدف پروژه

در این پروژه قصد داریم تا اکثر روابط مرسوم و رایج بین خانواده را در زبان پرولوگ تنها با تعریف کردن چند واقعیت ساده، مثل حقایق father, mother, female, male سپس بتوانیم با دانش طبیعی که در این مورد داریم، سعی به برقراری روابط منطق بولی بین واقعیت‌هایمان کنیم و برای هر یک از این روابط، قوانینی را که درواقع همان روابط خانوادگی پیچیده‌تر هستند را پیاده‌سازی کنیم. بدین منظور در ابتدای کار، یک درخت خانواده اصیل ایرانی (فرضی) را می‌سازیم تا در ادامه پروژه بتوانیم تمام روابط ممکن بین اعضای این خانواده خیالی را مدل کنیم و سپس به سادگی درخواست‌ها و کوئری‌های خودمان را درباره روابط هر یک از اعضا بپرسیم و پاسخ‌های درست و دقیق را فوراً دریافت کنیم.

تشریح پروژه

در ابتدای کار شروع به ساختن این خانواده فرضی می‌کنیم، بدلیل اینکه می‌خواهیم تمام روابط مرسوم را پیاده‌سازی کنیم، نیاز داریم بنحوی این ارتباطات را با ایجاد فرزندها و والدھای درست مدل کنیم، به همین دلیل این خانواده به صورت عمده دارای این تعداد از افراد فرزند و والد دختر و پسر بوده تا امکان پیاده‌سازی تمام روابط خواسته شده میسر شود، در تصویر زیر شجره‌نامه‌ای از خانواده درست شده، تا امکان مشاهده روابط به شکل عینی و ملموس امکان پذیر کند.

پارسا یوسفی نژاد محمدی



همانطور که در تصویر بالا می‌بینیم، این خانواده شامل ۱۰ مرد و ۱۱ خانم است همچنین رابطه پدری (father) برای ۱۴ جفت از افراد برقرار می‌باشد و رابطه مادری نیز (mother) برای ۱۴ جفت برقرار است، این ۴ رابطه تنها دانسته‌ها و واقعیت‌های موجود در فرض مساله برنامه است، به همین دلیل این حقایق را به صورت واقعیت تعریف می‌کنیم:

- 1** male(sorena).
- 2** male(arvin).
- 3** male(mahyar).
- 4** male(radvin).
- 5** male(shayan).
- 6** male(farhad).
- 7** male(milad).
- 8** male(mehran).
- 9** male(parham).
- 10** male(kian).

- 1** female(baran).
- 2** female(delaram).
- 3** female(bahar).
- 4** female(raha).
- 5** female(delasa).
- 6** female(nafas).
- 7** female(yasamin).
- 8** female(negar).
- 9** female(shahrzad).
- 10** female(giso).
- 11** female(sara).

- 1** mother(baran,delaram).
- 2** mother(delaram,mahyar).
- 3** mother(delaram,radvin).
- 4** mother(delaram,delasa).
- 5** mother(delaram,nafas).
- 6** mother(delasa,negar).
- 7** mother(delasa,milad).
- 8** mother(delasa,yasamin).
- 9** mother(raha,shahrzad).
- 10** mother(raha,mehran).
- 11** mother(bahar,giso).
- 12** mother(bahar,parham).
- 13** mother(nafas,kian).
- 14** mother(nafas,sara).

- 1** father(sorena,delaram).
- 2** father(arvin,mahyar).
- 3** father(arvin,radvin).
- 4** father(arvin,delasa).
- 5** father(arvin,nafas).
- 6** father(shayan,negar).
- 7** father(shayan,milad).
- 8** father(shayan,yasamin).
- 9** father(radvin,shahrzad).
- 10** father(radvin,mehran).
- 11** father(mahyar,giso).
- 12** father(mahyar,parham).
- 13** father(farhad,kian).
- 14** father(farhad,sara).

راهنمای واقعیت‌ها:

male(X).

این واقعیت بدین معناست که رابطه مرد بودن بر روی فرد X برقرار است.

female(X).

این واقعیت بدین معناست که رابطه خانم بودن بر روی فرد X برقرار است.

mother(X,Y).

این واقعیت بدین معناست که X فرد مادر فرد Y است.

father(X,Y).

این واقعیت بدین معناست که فرد X پدر فرد Y است.

پس از اتمام تعریف تمام fact‌ها، شروع به ایجاد قوانینی برای روابط گوناگون خانوادگی، تنها به کمک همین فکت‌ها و گاهی استفاده از برخی قوانین دیگر می‌کنیم. حال نوبت تفسیر و پیاده‌سازی این قوانین به کمک پایگاه دانش‌مان است.

پیاده‌سازی قوانین نسبت‌های فامیلی

• **parent(X, Y)**

این قانون یعنی اینکه فرد X والد فرد Y است، برای برقرار بودن چنین عبارتی لازم است:

parent(X, Y) :-

mother(X, Y);

father(X, Y).

در آن واحد، X مادر Y باشد و یا (;) X پدر Y باشد، در این صورت این عبارت **true** خواهد شد.

• **grandmother(X, Y)**

برای اینکه فرد X، مادربزرگ فرد Y باشد، باید چنین شروطی همزمان رعایت شوند:

grandmother(X, Y) :-

female(X),

mother(X, Z),

parent(Z, Y).

X برای مادربزرگ بودن، حتما باید خانم باشد و (,) اینکه مادر فرد Z ای باشد که آن فرد والد فرد Y باشد، در صورت پیدا شدن چنین فردی (Z) عبارت **true** خواهد شد.

• **grandfather(X, Y)**

همچنین لازمه اینکه فرد X، پدربزرگ Y باشد، این چنین است:

grandfather(X, Y) :-

male(X),

father(X, Z),

parent(Z, Y).

پدربزرگ باید حتما مرد بوده و همچنین پدر فردی مثل Z باشد بطوری که همان Z والد فردی چون Y باشد، در این صورت می‌شود گفت که X پدربزرگی برای Y است.

• **sister(X,Y)**

sister(X,Y) :-

female(X),
mother(W,X),
mother(W,Y),
X \= Y.

فرد X خواهر فرد Y است، اگر و تنها اگر (:-) :

X خانم باشد و برای افراد X و Y، مادر مشترکی به نام W پیدا شود و اینکه X و Y یکسان نباشند.

• **brother(X,Y)**

brother(X,Y) :-

male(X),
mother(W,X),
mother(W,Y),
X \= Y.

برادر Y است، تنها در صورتی که:

X باید مرد بوده و همچنین مادر این دو فرد (یا پدر) یکی و برابر W بوده و هم اینکه این دو فرد یکی نباشند.

• **khale(X,Y)**

khale(X,Y) :-

female(X),
mother(W,Y),
sister(X,W).

خاله فرد Y است، اگر و تنها اگر:

X خانم باشد، و Y فرزند مادر W بوده بطوری که W با خانم X خواهر باشند، در این صورت، این خاله Y فرد X خواهد بود.

• **amo(X,Y)**

amo(X,Y) :-

male(X),
father(W,Y),
brother(X,W).

فرد X عمو فرد Y است، تنها به شرطی که:

X باید مرد بوده و اینکه پدر Y با X برادر باشد، اگر چنین پدری (W) پیدا شود، پرسش ما true خواهد شد.

برای مابقی نسبت‌های فامیلی، بدلیل زیاد بودن روابط و واضح و یکی بودن روند تشخیص وجود چنین ارتباطاتی، تنها به آوردن کد شروط آن‌ها اکتفا می‌کنیم:

dayi(X,Y) :-

**male(X),
mother(W,Y),
brother(X,W).**

ame(X,Y) :-

**female(X),
father(W,Y),
sister(X,W).**

pesarkhale(X,Y) :-

**male(X),
khale(W,Y),
mother(W,X).**

dokhtarkhale(X,Y) :-

**female(X),
khale(W,Y),
mother(W,X).**

dokhtarame(X,Y) :-

**female(X),
ame(W,Y),
mother(W,X).**

dokhtardayi(X,Y) :-

**female(X),
dayi(W,Y),
father(W,X).**

pesarame(X,Y) :-

**male(X),
ame(W,Y),
mother(W,X).**

pesardayi(X,Y) :-

**male(X),
dayi(W,Y),
father(W,X).**

Parse Yousefi Nejad

شیوه‌های کوئری دادن

• بررسی صحت یک رابطه و قانون

در این نوع کوئری‌ها، ما تمام پارامترهای واقعیت و یا قانون را داریم، و از کامپایلر پرولوگ می‌خواهیم که صحت و درستی چنین گزاره‌ای را بررسی کند و پرولوگ با توجه به پایگاه دانش و قوانینی که در اختیار دارد، دست به تایید و یا رد کردن صحت پرسش ما می‌کند، نمونه‌هایی از چنین پرسش‌هایی در ادامه آمده است، همچنین خود نیز با توجه به شجره نامه افراد خانواده، مستقیماً می‌توانیم به چنین سوالاتی پاسخ دهیم.

?- dokhtarame(kian,mehran).
false.

?- mother(nafas,giso).
false.

?- dayi(radvin,negar).
true.

?- parent(delaram,kian).
false.

?- father(arvin,mahyar).
true.

?- dokhtarkhale(sara,milad).
true .

?- ame(delasa,parham).
true.

?- dokhtarame(kian,mehran).
false.

• درخواست یافتن فرد [دسته افراد] مجهول طرفین یک قانون

در این دسته از کوئری‌ها، یک فرد از طرفین قانون درخواست شده، مجهول می‌باشد و ما از پرولوگ میخواهیم تا فرد یا دسته افرادی که در این قوانین و یا فکت‌ها موجب true. شدن کوئری می‌شوند را پیدا کرده و آن‌ها را برگرداند، برای این منظور فردی را که مجهول واقع شده است را با متغیر دلخواهی چون X، در کوئری‌مان پارامتردهی می‌کنیم و پرولوگ سعی در یافتن افرادی می‌کند که در رابطه صدق کنند اما اگر نتواند که فردی را برای رابطه پیدا کند، را به معنای عدم وجود هرگونه رابطه خانوادگی با فرد مذکور باز می‌گردد.

```
?- sister(yasamin,X).  
X = negar ;  
X = milad ;  
false.
```

```
?- pesarame(X,giso).  
X = milad ;  
X = kian ;  
false.
```

```
?- ame(nafas,X).  
X = shahrzad ;  
X = mehran ;  
X = giso ;  
X = parham ;  
false.
```

```
?- dokhtardayi(X,milad).  
X = shahrzad ;  
X = giso ;  
false.
```

```
?- amo(X,shahrzad).  
X = mahyar ;  
false.
```

```
?- brother(giso,X).  
false.
```

```
?- grandfather(X,baran).  
false.
```

توجه: در انتهای هر عبارت بدین معنا است که پرولوگ دیگر نتوانسته است که فرد دیگر را که در این رابطه صدق کنند را پیدا کند به همین منظور شاهد وجود چنین عبارتی در انتهای هر کوئری هستیم.

• درخواست ارائه کل افراد دارای یک رابطه خاص در مساله

در این شیوه درخواست، از پرولوگ خواسته می‌شود تا تمام دسته افرادی را که در رابطه درخواست شده صدق می‌کنند را برگرداند، بدین منظور تنها کاری که نیاز است هنگام کوئری دادن به کامپایلر swi-prolog انجام دهیم، آن است که تمام پارامترهای کوئری را متغیرهایی با نام‌های گوناگون مقداردهی کنیم، تا پرولوگ در صورت امکان تمام دسته افرادی را که در رابطه مدنظر صدق می‌کنند را در خروجی چاپ کند، توجه کنید

که در آخر نیز مانند کوئری‌های قبلی، شاهد عبارت **false**. بدلیل عدم امکان پیدا کردن فرد دیگری، خواهیم بود که کاملاً منطقی می‌باشد.

```
?- female(X).
X = baran ;
X = delaram ;
X = bahar ;
X = raha ;
X = delasa ;
X = nafas ;
X = yasamin ;
X = negar ;
X = shahrzad ;
X = giso ;
X = sara.
```

```
?- amo(X,Y).
X = mahyar,
Y = shahrzad ;
X = mahyar,
Y = mehran ;
X = radvin,
Y = giso ;
X = radvin,
Y = parham ;
false.
```

```
?- ame(X,Y).
X = delasa,
Y = shahrzad ;
X = delasa,
Y = mehran ;
X = delasa,
Y = giso ;
X = delasa,
Y = parham ;
X = nafas,
Y = shahrzad ;
X = nafas,
Y = mehran ;
X = nafas,
Y = giso ;
X = nafas,
Y = parham ;
false.
```

```
?- pesarame(X,Y).
X = milad,
Y = shahrzad ;
X = milad,
Y = mehran ;
X = milad,
Y = giso ;
X = milad,
Y = parham ;
X = kian,
Y = shahrzad ;
X = kian,
Y = mehran ;
X = kian,
Y = giso ;
X = kian,
Y = parham ;
false.
```

PARSA YOUSEFI NEJAD

جمع‌بندی پروژه

در این پروژه پرولوگ توانستیم روابط فامیلی و خانوادگی واقعی افراد یک خانواده فرضی را با کمک قابلیت‌ها و ویژگی‌های خاص زبان پرولوگ به سادگی محل کنیم؛ در بخش اول، واقعیت‌های مساله را با کمک شجره‌نامه خانواده طراحی کردیم، و در بخش دوم، شروع به ایجاد قوانین و فکت‌های متناسب با هر نسبت فامیلی کردیم و هر کدام را با جزئیات بررسی کردیم و در انتها با توجه به توانایی‌های پرولوگ در برقراری ارتباطات منطقی بین هر واقعیت و قانون‌ها توانستیم انواع اقسام کوئری‌های گوناگون را بپرسیم و متناسب با آن‌ها جواب مورد انتظار را دریافت کنیم، به این ترتیب در این پروژه یک سیستم کامل را برای بررسی روابط خانوادگی را با زبان پرولوگ طراحی و ایجاد کردیم.