ผนวก ข

การกำหนดค่าที่ใช้ในการสร้างกฎด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัยโดยใช้ Aleph ในงานวิจัย

ในงานวิจัยชิ้นนี้ได้ใช้เป็นเซตตัวอย่างเรียนรู้ 3 เซตคือ เซตข้อมูลตัวอย่างการวิเคราะห์ความสามารถก่อกลายพันธุ์ของโมเลกุล (Mutagenesis) ข้อมูลการก่อมะเร็ง (Carcinogenesis) และการหาจำนวนองค์ประกอบจำกัดของโครงข่าย (Finite-Element Mesh) ซึ่งในส่วนของข้อมูลตัวอย่างนั้นสามารถหาได้จาก http://www.doc.ic.ac.uk/~shm/applications.html และ http://web.comlab.ox.ac.uk/oucl/research/areas/machlearn/applications.html

อย่างไรก็ดีในเซตข้อมูลตัวอย่างบางตัวเป็นเซตข้อมูลที่ใช้สำหรับเครื่องมือสร้างกฎด้วยวิธีการโปรแกรมตรรกะเชิงอุปนัยตัวอื่น จึงมีความจำเป็นต้องมีการปรับรูปแบบของความรู้ภูมิหลังให้เหมาะสมกับ Aleph ด้วย ซึ่งจะทำการแสดงค่าตัวแปรบางตัวซึ่งใช้ในการเรียนรู้และสร้างกฎสำหรับงานวิจัยชิ้นนี้ อย่างไรก็ดีรายละเอียดและคำอธิบายในการกำหนดค่าในการเรียนรู้ของ Aleph ทั้งหมดสามารถหาอ่านได้จากคู่มือการใช้ Aleph

ข้อมูลตัวอย่างการวิเคราะห์ความสามารถก่อกลายพันธุ์ของโมเลกุล (Mutagenesis)

คำสั่ง (COMMANDS)

:- set(noise,0). // กำหนดให้กฎที่สร้างได้ต้องไม่ครอบคลุมตัวอย่างลบ

:- set(search,rls). // กำหนดวิธีการค้นโดยใช้วิธีการ rls

:- set(rls_type,gsat). // กำหนดวิธีการสุ่มค้นหาโดยใช้วิธีการ gsat ซึ่งสามารถหาอ่านรายละเอียดได้ในคู่มือ

:- set(clauselength,8). //กำหนดให้ความยาวของกฎสูงสุดไม่เกิน 8 อนุประโยค

:- set(minpos,3). // กำหนดให้กฎที่สร้างต้องมีตัวอย่างบวกอย่างน้อย 3 ตัว

โหมด(MODE)

:- modeh(1,active(+drug)).

```
:- modeb(1,lumo(+drug,-energy)).
:- modeb(1,logp(+drug,-hydrophob)).
:- modeb(3,bond(+drug,-atomid,-atomid,#int)).
:- modeb(3,bond(+drug,+atomid,-atomid,#int)).
:- modeb(3,atm(+drug,-atomid,#element,#int,-charge)).
:- modeb(1,gteq(+charge,#float)).
:- modeb(1,gteq(+energy,#float)).
:- modeb(1,gteq(+hydrophob,#float)).
:- modeb(1,lteq(+charge,#float)).
:- modeb(1,lteq(+energy,#float)).
:- modeb(1,lteq(+hydrophob,#float)).
:- modeb(1,(+charge)=(#charge)).
:- modeb(1,(+energy)=(#energy)).
:- modeb(1,(+hydrophob)=(#hydrophob)).
:- modeb(*,benzene(+drug,-ring)).
:- modeb(*,carbon_5_aromatic_ring(+drug,-ring)).
:- modeb(*,carbon_6_ring(+drug,-ring)).
:- modeb(*,hetero_aromatic_6_ring(+drug,-ring)).
:- modeb(*,hetero_aromatic_5_ring(+drug,-ring)).
:- modeb(*,ring_size_6(+drug,-ring)).
:- modeb(*,ring_size_5(+drug,-ring)).
:- modeb(*,nitro(+drug,-ring)).
:- modeb(*,methyl(+drug,-ring)).
:- modeb(*,anthracene(+drug,-ringlist)).
:- modeb(*,phenanthrene(+drug,-ringlist)).
:- modeb(*,ball3(+drug,-ringlist)).

:- modeb(*,member(-ring,+ringlist)).
:- modeb(1,member(+ring,+ringlist)).
```

```
:- modeb(1,connected(+ring,+ring)).
```

ดีเทอร์มิเนชั่น (DETERMINATIONS)

```
:- determination(active/1,atm/5).
:- determination(active/1,bond/4).
:- determination(active/1,gteq/2).
:- determination(active/1,lteq/2).
:- determination(active/1,'='/2).
:- determination(active/1,lumo/2).
:- determination(active/1,logp/2).
:- determination(active/1,benzene/2).
:- determination(active/1,carbon_5_aromatic_ring/2).
:- determination(active/1,carbon_6_ring/2).
:- determination(active/1,hetero_aromatic_6_ring/2).
:- determination(active/1,hetero_aromatic_5_ring/2).
:- determination(active/1,ring_size_6/2).
:- determination(active/1,ring_size_5/2).
:- determination(active/1,nitro/2).
:- determination(active/1,methyl/2).
:- determination(active/1,anthracene/2).
:- determination(active/1,phenanthrene/2).
:- determination(active/1,ball3/2).
:- determination(active/1,member/2).
:- determination(active/1,connected/2).
```

ข้อมูลตัวอย่างการก่อมะเร็ง (Carcinogenesis)

คำสั่ง (COMMANDS)

:- set(noise,1). // กำหนดให้กฎที่สร้างได้ครอบคลุมตัวอย่างลบได้ 1ตัว

:- set(search,rls). // กำหนดวิธีการค้นโดยใช้วิธีการ rls

:- set(rls_type,gsat). // กำหนดวิธีการสุ่มค้นหาโดยใช้วิธีการ gsat ซึ่งสามารถหาอ่านรายละเอียดได้ในคู่มือ

:- set(clauselength,8). //กำหนดให้ความยาวของกฎสูงสุดไม่เกิน 8 อนุประโยค

:- set(minpos,3). // กำหนดให้กฎที่สร้างต้องมีตัวอย่างบวกอย่างน้อย 3 ตัว

โหมด(MODE)

:- modeh(1,active(+drug)).

:- modeb(5,ames(+drug)).

:- modeb(5,mutagenic(+drug)).

:- modeb(5,has_property(+drug,#property,#propval)).

:- modeb(5,ashby_alert(#alert,+drug,-ring)).

:- modeb(5,ind(+drug,#alert,-nalerts)).

:- modeb(5,atm(+drug,-atomid,#element,#integer,-charge)).

:- modeb(5,symbond(+drug,+atomid,-atomid,#integer)).

:- modeb(1,gteq(+charge,#real)).

:- modeb(1,lteq(+charge,#real)).

:- modeb(1,((+charge) = (#charge))).

:- modeb(1,gteq(+nalerts,#integer)).

:- modeb(1,lteq(+nalerts,#integer)).

:- modeb(1,((+nalerts) = (#nalerts))).

:- modeb(5,nitro(+drug,-ring)).

:- modeb(5,sulfo(+drug,-ring)).

:- modeb(5,sulfo(+drug,-ring)).

:- modeb(5,methyl(+drug,-ring)).

:- modeb(5,methoxy(+drug,-ring)).

:- modeb(5,amine(+drug,-ring)).

:- modeb(5,amine(+drug,-ring)).

:- modeb(5,aldehyde(+drug,-ring)).

:- modeb(5,ketone(+drug,-ring)).

:- modeb(5,ether(+drug,-ring)).

:- modeb(5,sulfide(+drug,-ring)).

:- modeb(5,alcohol(+drug,-ring)).

:- modeb(5,phenol(+drug,-ring)).

:- modeb(5,carboxylic_acid(+drug,-ring)).

:- modeb(5,ester(+drug,-ring)).

:- modeb(5,amide(+drug,-ring)).

:- modeb(5,deoxy_amide(+drug,-ring)).

:- modeb(5,imine(+drug,-ring)).

:- modeb(5,alkyl_halide(+drug,-ring)).

:- modeb(5,ar_halide(+drug,-ring)).

:- modeb(5,benzene(+drug,-ring)).

:- modeb(5,hetero_ar_6_ring(+drug,-ring)).

:- modeb(5,non_ar_6c_ring(+drug,-ring)).

:- modeb(5,non_ar_hetero_6_ring(+drug,-ring)).

:- modeb(5,six_ring(+drug,-ring)).

:- modeb(5,carbon_5_ar_ring(+drug,-ring)).

:- modeb(5,hetero_ar_5_ring(+drug,-ring)).

:- modeb(5,non_ar_5c_ring(+drug,-ring)).

:- modeb(5,non_ar_hetero_5_ring(+drug,-ring)).

:- modeb(5,five_ring(+drug,-ring)).

:- modeb(1,connected(+ring,+ring)).

ดีเทอร์มิเนชั่น (DETERMINATIONS)

:- determination(active/1,ames/1).

:- determination(active/1,has_property/3).

:- determination(active/1,ashby_alert/3).

:- determination(active/1,mutagenic/1).

:- determination(active/1,ind/3).

:- determination(active/1,atm/5).

:- determination(active/1,symbond/4).

:- determination(active/1,nitro/2).

:- determination(active/1,nitro/2).

:- determination(active/1,sulfo/2).

:- determination(active/1,methyl/2).

:- determination(active/1,methoxy/2).

:- determination(active/1,amine/2).

:- determination(active/1,aldehyde/2).

:- determination(active/1,ketone/2).

:- determination(active/1,ether/2).

:- determination(active/1,sulfide/2).

:- determination(active/1,alcohol/2).

:- determination(active/1,phenol/2).

:- determination(active/1,carboxylic_acid/2).

:- determination(active/1,ester/2).

:- determination(active/1,amide/2).

:- determination(active/1,deoxy_amide/2).

:- determination(active/1,alkyl_halide/2).

:- determination(active/1,ar_halide/2).

:- determination(active/1,benzene/2).

:- determination(active/1,hetero_ar_6_ring/2).

:- determination(active/1,non_ar_6c_ring/2).

:- determination(active/1,non_ar_hetero_6_ring/2).

:- determination(active/1,six_ring/2).

:- determination(active/1,carbon_5_ar_ring/2).

:- determination(active/1,hetero_ar_5_ring/2).

:- determination(active/1,non_ar_5c_ring/2).

:- determination(active/1,non_ar_hetero_5_ring/2).

:- determination(active/1,five_ring/2).

:- determination(active/1,connected/2).

:- determination(active/1,lteq/2).

:- determination(active/1,gteq/2).

:- determination(active/1,'='/2).

ข้อมูลตัวอย่างการหาจำนวนองค์ประกอบจำกัดของโครงข่าย (Finite-Element Mesh)

คำสั่ง (COMMANDS)

:- set(noise,0). // กำหนดให้กฎที่สร้างได้ต้องไม่ครอบคลุมตัวอย่างลบ

:- set(search,rls). // กำหนดวิธีการค้นโดยใช้วิธีการ rls

:- set(rls_type,gsat). // กำหนดวิธีการสุ่มค้นหาโดยใช้วิธีการ gsat ซึ่งสามารถหาอ่านรายละเอียดได้ในคู่มือ

:- set(clauselength,8). //กำหนดให้ความยาวของกฎสูงสุดไม่เกิน 8 อนุประโยค

:- set(minpos,3). // กำหนดให้กฎที่สร้างต้องมีตัวอย่างบวกอย่างน้อย 3 ตัว

โหมด (MODE)

:- modeb(1,long(+atom)).

:- modeb(1,usual(+atom)).

:- modeb(1,short(+atom)).

:- modeb(1,circuit(+atom)).

:- modeb(1,half_circuit(+atom)).

:- modeb(1,quarter_circuit(+atom)).

:- modeb(1,short_for_hole(+atom)).

:- modeb(1,long_for_hole(+atom)).

:- modeb(1,circuit_hole(+atom)).

:- modeb(1,half_circuit_hole(+atom)).

:- modeb(1,not_important(+atom)).

:- modeb(1,free(+atom)).

:- modeb(1,one_side_fixed(+atom)).

:- modeb(1,two_side_fixed(+atom)).

:- modeb(1,fixed(+atom)).

:- modeb(1,not_loaded(+atom)).

:- modeb(1,one_side_loaded(+atom)).

:- modeb(1,two_side_loaded(+atom)).

:- modeb(1,cont_loaded(+atom)).

:- modeb(1,neighbour_xy_r(+atom,-atom)).

:- modeb(1,neighbour_yz_r(+atom,-atom)).

:- modeb(1,neighbour_zx_r(+atom,-atom)).

:- modeb(1,neighbour_xy_l(-atom,+atom)).

:- modeb(1,neighbour_yz_l(-atom,+atom)).

:- modeb(1,neighbour_zx_l(-atom,+atom)).

:- modeb(1,opposite_r(+atom,-atom)).

:- modeb(1,opposite_l(-atom,+atom)).

:- modeb(1,equal_r(+atom,-atom)).

:- modeb(1,equal_l(-atom,+atom)).

:- modeh(1,mesh(+atom)).

ดีเทอร์มิเนชั่น (DETERM)

:- determination(mesh/1,long/1).

:- determination(mesh/1,usual/1).

:- determination(mesh/1,short/1).

:- determination(mesh/1,circuit/1).

:- determination(mesh/1,half_circuit/1).

:- determination(mesh/1,quarter_circuit/1).

:- determination(mesh/1,short_for_hole/1).

```
:- determination(mesh/1,long_for_hole/1).
:- determination(mesh/1,circuit_hole/1).
:- determination(mesh/1,half_circuit_hole/1).
:- determination(mesh/1,not_important/1).
:- determination(mesh/1,free/1).
:- determination(mesh/1,one_side_fixed/1).
:- determination(mesh/1,two_side_fixed/1).
:- determination(mesh/1,fixed/1).
:- determination(mesh/1,not_loaded/1).
:- determination(mesh/1,one_side_loaded/1).
:- determination(mesh/1,two_side_loaded/1).
:- determination(mesh/1,cont_loaded/1).
:- determination(mesh/1,neighbour_xy_r/2).
:- determination(mesh/1,neighbour_yz_r/2).
:- determination(mesh/1,neighbour_zx_r/2).
:- determination(mesh/1,neighbour_xy_l/2).
:- determination(mesh/1,neighbour_yz_l/2).
:- determination(mesh/1,neighbour_zx_l/2).
:- determination(mesh/1,opposite_r/2).
:- determination(mesh/1,opposite_l/2).
:- determination(mesh/1,equal_r/2).
:- determination(mesh/1,equal_l/2).
:- determination(mesh/1,mesh/1).
```