

(es)
expert system

دوشنبه

۳۰

هفته ۵۳

۱۴۳۸ / جمادی الثانی / ۲۱

۳۶۶ گذشتہ / ۰ مانده

20 / March / 2017

اوسقشک

هوش مصنوعی و سیستم خبره

↓ Artificial Intelligence (AI)

۸ هوش یعنی توانایی درک و فهم مسئله جهت حل آن همانند انسان. → هوش مصنوعی سیستمی که این توان را از طریق روش سیستم هوشمند (ربات) نامیده می شود.

۱۰ معیارهای تشخیص داریم: ۱. بازه توان را تشخیص، نوآوری آن را تشخیص. هوش طبیعی انسان وارد ماشین می شود به همین دلیل به آن هوش مصنوعی می گویند.

۱۲ ۲. تا نظریه وجود دارد. ۱. انسان کنترلی دارد و ماشین هم کار می کند. ۲. بسیاری از ویژگی های انسان

مثل خواب دیدن و حس ششم ناشناخته است و نمی توان به ربات انتقال داد.

عامل agent چیزی است که با اطلاعاتی که از محیط می گیرد تصمیم می گیرد.

۱۴ + زبان برنامه نویسی پرولوگ
ارزیابی کتاب: میان ترم: (حجت سیستم خبره) ۵ نمره
۲۰ پایان
۲۰ کلاس

تکلیف ۲ نمره

۱۶ پایان ترم ۸ نمره

پروژه ۳ نمره

میان ترم ۲ ۲ نمره

۱۸

۲۰ اگر به پروژه راجع به سیستم خبره logic باشد باید حتماً با زبان predicate
prolog نوشته شود
↓
programming logic

کلیه courses : AIES

تا ختم = سیرینی موبایل X

دانش knowledge عیناً اصلی صورت است.

Knowledge Representation بازتابی دانش:

۱- OAR Triplets ستای - ستی صفت خاصه - مقدار

۲- Semantic Network شبکه معنای

۳- Frame قاب

۴- Rule یا if-then قاعده یا اگر - آنگاه

۵- logic منطق - منطق گزاره‌ای propositional logic

منطق مسندی predicate logic

۱۴ روش Frame و rule بیشترین استفاده را در سیستم خبره دارند.

در N.L.P از منطق مسندی خیلی استفاده می‌شود.

لایه‌های زبان طبیعی natural language Process

سیستم خبره سیستمی است که توانایی حل مسئله اش همانند یک فرد خبره است.

صورتی معنوی: تعاریف و تاریخچه کاربردها

agent

problem solving

search

knowledge Representation

ES

تعریف و تاریخچه Rule

K-R Frame logic

knowledge Acquisition

احتیاجات کتاب دانش

ولادت حضرت فاطمه زهرا (س) (هشتم قبل از هجرت) و روز زن - تولد حضرت امام خمینی (ره) رهبر کبیر انقلاب اسلامی (۱۳۲۰ هـ.ق) - روز ملی شدن صنعت نفت ایران (۱۳۲۹ هـ.ش) (تعطیل)

inferrence engine موتور استنتاج ← روش‌های استنتاج

۸ inference engine - موتور استنتاج

۹ explanation - توجیه کننده یا تشریح کننده

۱۰ validation and verification of knowledge - اعتبار رسانی و تأیید دانش

۱۱ reasoning - استدلال و استنتاج

۱۲ learning - یادگیری

uncertainty knowledge - دانش های نامطمئن ، عدم قطعیت

۱۴ دورنگار در هوش مصنوعی وجود دارد - هوش
حاشیه می تواند کاری کند که روی محاسبات تاثیر بگذارد به طور یو یا داینامیک تغییر کند - مربوط به هوش
زبان شناسی Linguistic سودمندی utility

۱۶ مکانیک + کنترل + کامپیوتر - شبکه های عصبی

۱۸ List programming - لیست برنامه نویسی (لیست که همه چیز به صورت لیست استفاده می شود)

۱۹ دلیلی بر دقت بودن پراکنش گذاری ندارد داشته شده کم کم

سیستم خبره به عنوان یکی از کاربردهای AI می باشد

پیدا کردن رفتار - شباهت کامپیوتر - یادگیری ماشین - استنتاج - برنامه نویسی - مربوط به AI

۲۰ evolutionary algorithm - الگوریتم تکاملی

swarm algorithm - الگوریتم مورچه ها

ant colony - الگوریتم مورچه ها - گوناگون ترین مسیر را پیدا می کنند

در رفتارشناسی هم باید کلمه استخفاف را هم باید معنی آن را با توجه به جمله بگویم - ai

سی ما انتظار داریم که سیستم به صورت مستقل و مستقل از سیستمی دهد و معنی علم را بفهمد. انتظار داریم از ربات ها این است که بتوانند یاد بگیرند و خود را با محیط سازگار کنند. به طوری که اطلاعاتی که می گیرند اطلاعات درست باشد. ما پیشین باید توانایی دیدن هم داشته باشد. بحث تشخیص همه پیشین ها می آید.

۱۰. انتظار داریم ربات ها هم مثل انسان تصمیم گیری و planning انجام دهند. که سیستم به صورت مستقل با مسائل که قید و محدودیت دارند می توانند این دو کار را کنند. دو نوعی هست وجه هم از نمونه های AI است. اگر مثل انسان فکر کند - مثل انسان عمل کند - هر چیزی را فکر کند - چه کاری را انجام دهد - خوشمند.

۱۲. تست تورینگ - آیا این ماشین خوشمند هست یا نه ۱۲. توی اتاق بسته و در یک اتاق انسان است و باز پرس سوال می کند و اگر نتواند بفهمد انسان در کدام اتاق است سیستم خوشمند نیست.

۱۴. سیستم باید بتواند مثل یک انسان رفتار کند و باز پرس را قبول بزند.

علوم شناختی cognitive science

autonomous خود مختار agent ها به صورت خود مختار رفتار عقلانی و درست را نشان می دهند

۲۷

جمعه

(rational)

۱۴۳۸ / جمادی الثانی / ۱۸

هفته ۵۲

۱۶

کنترل

استفانی

17 / March / 2017

۳۶۳ گذشته / ۳ مانده

agent عامل های خوشمند

۱۸. agent هر چیزی است که می تواند محیط را لمس کند و بر اساس آن می تواند محیط را توصیف کند

هوشیار دریافت کردند که action انجام دهند و تصمیم گیری می کنند و برای انجام کار از محرک ها استفاده می کنند. و دهان دست و ... → actuator

۲۰. actuator به هر چیزی که به act کردن کمک می کند.

هر agent یک تابع دارد agent function

هر agent یک معادله می خواند و یک برنامه (program)

۷. $performance\ measure$ یا معیار ارزیابی دانش بالینیم که $agent$ را از نظر رفتاری و عملی بودن بررسی کنیم و $agents$ کار درست را انجام دهد. مثلاً هدف به دست آوردن حقارت، حقارت
 نوین به وجود آورده و حقارت اشتغال جمع کرده - این موارد معیارهای ارزیابی قابل بررسی هستند.
 ۱۰. این معیارها را گاهی ماکزیمایز و گاهی مینیمایز می‌کنیم.

۱۱. اگر بتوانیم دانش های مفید و درست به ربات چارو بدهیم $action$ آن منطقی و درست خواهد بود.
 اگر به دیوار برخورد، معیار ارزیابی به ما می‌گوید که سطح جمع گیری ربات درست نیست.

۱۲. $PEAS$ → $actuators$ $environment$
 معیار به هر چیزی که در سطح جمع گیری مؤثر است.
 معیارهای ارزیابی

یک تاکتی خود رها ر:

۱۴. معیارهای ارزیابی: رعایت قوانین رانندگی، سریع ترین و بهترین مسیر، تشخیص شرایط اضطراری،
 ایمنی، هزینه، مصرف سوخت و ...

۱۶. محیط: خیابان، ماشین، عابر، علائم راهنمایی رانندگی، آب و هوا، موقعیت و ...
 $actuator$: چراغ، بوق، چرخ، ترمز، پدال گاز و برف پاک کن و ...
 $sensors$: دوربین، حسگرهای خاص، سونار، GPS،

۱۸. مدل $PEAS$ را می‌توانیم به کمک ربات $Part\ Picking\ Agent$ و ...
 رباتی که قطعه قطعاتی را در انبارها بگذارد:

۲۰. P : سرعت جابه جایی قطعات - سلامت قطعات - قرار گرفتن قطعه در مکان درست -
 E : قطعات - موانع - شرایط محیطی - وضعیت مسیرهای حرکتی - انبار و مقصد -

A : بازوی ربات - چرخ - مفاصل -
 S : دوربین

نیم اقدار آموزش زبان (چهارده ای گفتگویی) :

۸ P : وقت در سطح صدا - کیفیت هزات - سرعت مدرسی - سرعت فزاینده - کیفیت تست ها - ارزیابی از روی

ج : دانشجو ها - سیستم (توسعه یا windows یا web) - خود زبان - آزمون (تست) - فزاینده ها - نتیجه آزمون - برآورد تست ها

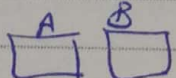
A : صفت نهایی - speaker - واحد پردازش

۱۰ 8 : میکروفون - mouse - key board

هدف این است که score دانش آموزان را maximize کند.

۱۲ Fully observable ← یعنی به طور کامل مشاهده کرده و قابل دسترس است ← تصمیمات راحت تر است.

Partially Deterministic ← یعنی بخشی از اطلاعات مخفی را می رسد. ← مثل پاسور

مثلاً اگر در ربات ها و برق  فقط فزونی در بین دانسته مانده partially است

۱۴ و اگر در هر دو آن دور بین باشد Fully است.

Deteminiotic ← قطعی ← state عری به state قبلی مربوط می شود.

stochastic ← state عری قطعی به state قبلی مرتبط نیست و اتفاق می افتد که آن ارزش بی غیر است.

Epizodic ← محیط را بتوان به بخشی های مستقل جدا از هم تقسیم کنیم که به هم ربطی ندارند.

sequential ← یعنی محیط برای تصمیم گیری به کارهای قبلی این توجه کند و تصمیم گیری کرد.

۱۸ به عنوان مثال : رباتی که در خانه قطعه فزاینده ها را پیدا می کند episode است ولی تعامل رخ

محیط sequential است.

Single r. ← محیط single است اگر agent داشته باشیم

multiagent ← محیط multiagent است اگر چند agent داشته باشیم مثل taxi driver.

static (ایستا) ← محیطی که قبل از تصمیم گیری تغییری نکند. { action و تغییر کنیم

(رویا)

dynamic ← محیطی که در آن وضعیت تغییر می‌کند.

۸ dynamic ← محیط پویا

static ← محیط ثابت

dynamic ← پویا

↓
مهره را با هم می‌زنیم و در حین بازی فیزیکی می‌کنیم.

Deterministic ← محیط قطعی

۱۰ continuous ← زمانی به صورت پیوسته

discrete ← زمانی به صورت پله‌ای گسسته

۱۱ static ← محیط ثابت

dynamic ← محیط پویا

multi agent

stochastic

۱۲ ← پویا

Simple reflex agent
State خود را مشخص می‌کند و تصمیم می‌گیرد که action انجام دهد.۱۳ $f(\text{location} = A, \text{status} = \text{dirty}) \rightarrow \text{condition}$

then suck

 $\rightarrow \text{action}$

بعد از کارش environment تغییر می‌کند.

model-base reflex agent
علاوه بر کارهایی که تصمیم به کارهای قبلی است هم توجه می‌کند و از وضعیت

های قبلی که در آن قرار گرفته، از خوراک مدل یادگیری دارد.

goal-base reflex agent
از محیط یادگیری یادگیری دارد و علاوه بر قبلی می‌تواند

۱۸ که آیا آن action به هدفش می‌رسد یا نه. در condition-action به goal توجه می‌کند.

تصمیم می‌گیرد که مثل taxi که به جای ۳ تا راه دارد و با توجه به هدفش معلوم می‌کند که کدام را برگزیند.

۲۰ utility-base reflex agent
تابع ارزیابی داریم. اگر آن action سودمند بود انتخاب می‌کنیم.

critic برای سنجش مسی قوای حدیث. Performance standard خارج از agent است. ممکنه agent

Performance standard را عایت کرده باشد. هر جا عایت نکرد sense مسی و feedback مسی و یادگیری

دیکه اون استاده رو نکرده.

مدیر باید مسائل کمی را به مافوق و مسائل

knowledge-base system

expert system ← سیستم مبتنی بر دانش

۸. یک سیستم هو شناس که در مورد موضوعی بصیرت ما را از یک فرد خبره و متخصص مهارت داشته باشد.

نیروی متخصص که کار دگدان است و es به عنوان دستیار نیروی متخصص بسیار مفید است.

در طراحی es می توانیم از دانش های چند خبره استفاده کنیم.

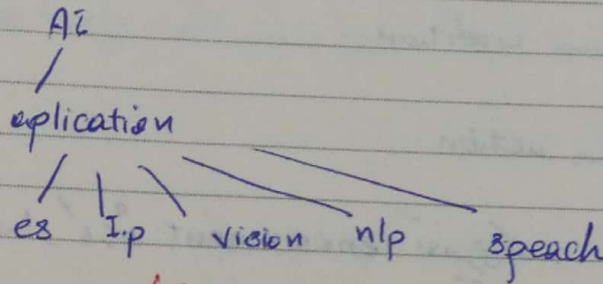
۱۰. هم است که از دانش های چه فردی داریم استفاده می کنیم در تولید سیستم خبره.

در سیستم خبره استدلال انسان مد نظر است ، لزوماً به جواب قطعی نمی رسیم ، ممکنه دانشمان ناقص

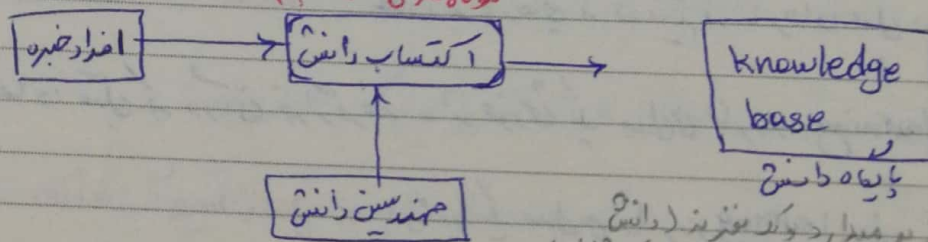
۱۲. نبوده باشد. سیستم خبره با موضوعات بصیرت سروکار دارد و با فرد خبره کار دارد. باید سریع و قابل

اعتماد باشد. نقدی کاربرهای حوش معنوی به عنوان ابزار تحقیقی استفاده می شوند. سیستم خبره

explanation نیاز دارد.



سکولاه برای es مهم ترین کار



بازبینی خواهی کرد که اگر چه بر می دارد و یک نفر (دانش) مکاشفه ای

۱۸. مسئله نیاز به حیرت انگیز دانسته باشد و افراد خبره کم باشند و اطلاعات ناقص باشد و هم می بینی

به دانش اطلاعات ناقص باشد. دانش مانی باید به اندازه ای باشد که kb با ارزش باشد

۲۰. سیستم خبره باید domain محدودی داشته باشد.

Knowledge Representation روش های ذخیره سازی دانش. روش هایی که برای کامپیوتر یا ربات قابل فهم باشد.

شفاف (transparent) روشی است که توسط انسان قابل فهم و درک باشد.

inferred knowledge در حین consult ای را می‌نویسد و case به case متفاوت اند. مثل پرسش.

priori knowledge به تغییر می‌کنند به مربوط به domain

سی در ez باید داشتن های priori را کسب کنیم. در همان دامنه.

اینها را خودی به inferred علامت می‌دهیم که ما خودی به priori

به هر دانی را در k.b نمی‌کنیم و در inferred می‌کنیم و اگر لازم باشد به k.b منتقل می‌کنیم.

Object Attribute - Value → ویژگی های هر شیء
↓
jet تعداد موتور هواپیما
به عنوان مثال در هواپیما

Semantic network (شبکه معنایی): گراف جهت داری که has، is-a و can و ... دارد.

node رابطی is-a تراشیده باشد کلاسی نیست. is-a به این وارد

روش مبتنی بر قاعده:

برای طراحی سیستم خبره همی rule ها باید if-then باشند. (if else نباشد)

if () → premise part

then () → conclusion part

اگر if clause، and داشته که هیچی ولی اگر OR بود آن را می‌توانیم به یک انسان. با این کاری می‌کنیم

کدام یک از clause صابی که در OR آمده درست بوده و دیگر قرار بوده.

prompt سوالی است که در مورد Attribute می‌پرسیم. به عنوان مثال تب داشتن Attribute

است و آیات داری prompt است.

specified value به شیء attribute فقط مقدار دارد یا صفت مقدار می‌تواند داشته باشد.

legal value loc = {A, B}

legal value status = {clean, dirty}

پمپت ← شیف؟

آیا B تمیزه؟ به عنوان ۵۰٪

۱۰. بی عنوان مال {عقد و قرضه و سبب و غیره} ← Attribute

prompt

specified ← یک رتبه یا عین رتبه؟

prompt حاوی برای صفت های تعریف می شوند که قابلیت بر سنج دارند و اینطور صفت محصول در نظر گرفته می شود

۱۲. action یک Attribute است که قابل بر سنج نیست و prompt ندارد.

intermedia conclusion ← هر clause نتیجه ای که در نظر ما قاعده ی دیگری آمده باشد. نتیجه میانی

iP-then ← روش مبتنی بر قاعده

۱۴. می توان در روش مبتنی بر قاعده و شبکه معنایی از OAV استفاده کنیم.

AV ← همان کلاس است.

۲۰

جمعه

۱۴۳۸ / جمادی الثانی / ۱۱

هفته ۵۱

استفاد

10 / March / 2017

۳۵۶ گذشتہ / ۱۰ مانده

unknown یعنی نمی دانیم جوابی را و سیستم بر همین

اساس باید تصمیم گیری کند که یک صفت اطمینان

۱۸. Not yet known یعنی هنوز نمی دانیم مقدار آن Attribute چیست ولی می تواند همین.

۱۸. clause حتماً به ← عطف یا فصلی می تواند باشند.

در قاعده هم یک certainty factor داریم که میزان اطمینان از نتیجه قاعده است.

۲۰. اگر قاعده True بود، صفت قاعده Trigger است و اگر نظر ما قاعده False باشد و صفت قاعده dis card است

اگر قاعده ی Trigger به عنوان یک Fact به زبانه بود Fire و صفت آن قاعده خواهد بود.

له عبارت های همیشه درست که اتوماتیک وضعیتش True می باشد.

قاعده ای که شرطی دارد که هنوز بررسی نشده ← active نامیده می شود.

۸

درست و غلط و وضعیت clause شرطی است و Trigger و Fire و discard وضعیت یک قاعده.

قواعدی که دارای وضعیت discard یا Fire هستند inactive و قاعده ای وضعیت Trigger.

۱۰ active است. اگر Fire شد نتیجه به عنوان Fact استفاده می شود و inactive می شود.

چیزی را Fire می کنیم که بعداً حتماً از آن نتیجه درست و ثابت استفاده کنیم.

هر موقع شرایط کم و اهداف زیاد بود ← forward chaining حرکت رو به جلو

۱۲

هر موقع شرایط زیاد و اهداف کم ← backward chaining حرکت رو به عقب

روش های استخراج نسبت استخراج را تغییر می دهد و در نتیجه دیگری ندارد.

۱۴ اول قاعده Free است و engine type نیز دانسته می شود است. بعد به Fact اضافه

می کنیم و می بینیم که Disablerule clause می شود چون clause آن false است. قاعده بعدی هم

هم در ابتدا Free است بعد حکم می کنیم می بینیم شرط اول True است. قاعده ی دوم هم

۱۶

Discard می شود. قاعده ی سوم شرطش True می شود و آن قاعده Trigger می شود. در قاعده دوم یک بار

prompt می بینیم و وضعیت bulge ها معلوم می شود و آن وضعیت را به Fact ها اضافه می کنیم حال در

قاعده سوم یک بار prompt می بینیم چون Fact را update کردیم. هنوز قاعده داریم پس active

است. وضعیت بالا ها absent است. inferred knowledge است.

پنری ← قبل از مسأله در سیستم در k.b. موجود دارند.

inferred knowledge

bulges position = absent
plane = CSA

working storage

دانش هایی که در WS هستند بسته به کاربرد متفاوت اند. این اطلاعات نباید در k.b. برود چون کاربرد بسیار

روز بزرگداشت سید جمال الدین اسدآبادی - سالروز تأسیس کانون های فرهنگی هنری مساجد کشور

این اطلاعات متفاوت است

۱۵

۱۴

۱۳

۱۲

۱۱

۱۰

۹

۸

۷

۶

۵

۴

۳

۲

۱

دانش های کاربردی

کاربرد آموزشی سیستم های اطلاعاتی از اندرون
اینها گرفته تا اداره ایالات متحد آمریکا به کیفیت
بهتری آن منتهی است

در این مثال کاربر به دنبال آن مدل خاص یا آن fact

۸. هاست پس fact ها کاربر به کاربر یاد می شود. اگر کاربر به prompt ما گفت "unknown" می توانیم حدس کنیم و مثلاً اگر به کاربر سوالی بپرسیم و جوابی بگیریم که کاربر به گفته حجت بود ما فرض کنیم کاربر گفته حجت و در explanation به این دلیل این در نظر گرفته می شود و باعث به تو پاسخ داده ایم. اگر تریا حرکت کنیم و به نتیجه رسیدیم پس forward chaining کوئری

اگر همین مثال را بخواهیم backward chaining بنویسیم. از goal ها شروع می کنیم.

۱۲. قاعده ای که نتیجه اش این بود Discard می شود. پس به این goal نیز می شود رسید.

| | | |
|------|---|----------------|
| goal | { | plane = C130 → |
| | | BV4V → |
| | | COA |
| | | CI21 |

۱۴. اگر تعداد هواپیماها خیلی کم درونی عقب رد مناسب نبود ولی از هر دو رو می به یک جواب می رسید. اگر تریا حرکت کنیم باید روشی رو به جلو مناسب نیست.

آن قواعدی که محتمل تر هستند باید اولی قرار بگیرند. آن هایی که احتمال به قرار گرفتن کمتر است باید تریا باشند. Deterministic باشد. در روش backward به معنی satisfy شدن یک قاعده دیگر به قاعده مربوط به آن goal را بررسی نمی کنیم. در روش forwarding چون ممکن است اولویت بندی داشته باشند قواعد و تریا های

۱۸. به روشی در قواعد باید تریا حرکت شود باید همه قواعد را بررسی کرد و بعد نتیجه گرفت. اگر stochastic باشد. فقط به روز رسانی می تواند بشود ولی با یک می شود بعد از هر کاربر

۲۲. Rule 1a ← Trigger می شود. در این قاعده با fire کنیم D=VX به عنوان D=VX

۲۰. با fact ها اضافه می شود. اگر قواعدمان به گونه ای باشند که یک CF در کنار rule وجود داشته باشد این حالت را در S می بینیم و تریا نتایج چیز گانه را می توانیم بنویسیم.

| |
|------|
| W,3 |
| D=XX |
| D=YY |

اگر رابطه Deterministic باشد ما می‌توانیم به صورت قطعی جواب بدهیم

تکلیف می‌شود اگر تناقض در استنتاج ما وجود داشته باشد rule های ما قطعی بودن Deterministic بود

پس با بردن اولین جواب آن را Fire می‌کنیم چون ممکن است جواب‌های دیگری باشد

داشتن پاسخ و یا نگاه داشتن Deterministic نباشد

مثلاً در Fact ها مقدار را داریم ولی B را نداریم پس در Fact Rule باید prompt بزنیم و

مقدار B را بگیریم Attribute

یکی از سیاست‌های تولید قاعده برای افزایش سرعت، دسته‌بندی قاعده‌ها می‌باشد. قواعدی که نتایج مشابه

یکسان اند در یک گروه قرار می‌گیرند. بعد دسته‌ها مرتب می‌کنیم Final conclusion را اول می‌زنیم.

(دسته‌ی مربوط به Final Conclusion) بعد در دسته‌های Final هر کدام که نتایج مشابه

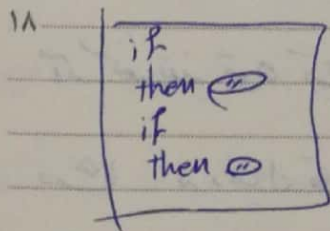
داشتند را با یکدیگر ترکیب می‌کنیم و نتیجه intermedia را دقیقاً زیرین قرار می‌دهیم.

اینکه دو قاعده را با یک قاعده کنیم همیشه خوب نیست چون طریقی که می‌تواند به نتیجه‌ی اشتباهی منجر شود

explaining نیاز داریم. اگر دسته‌بندی کردیم و به این دلیل که نتیجه‌ی دسته‌بندی مان یکی شد خوب

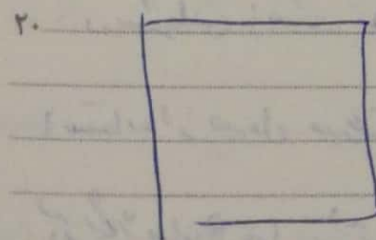
نیست. به همین دلیل است که خوب نیست OR استفاده کنیم و خوب نیست در Conclusion از

and استفاده کنیم.



Rule 1: if $A=a \wedge B=b \wedge C=c$,
then $D=x$

Rule 2: if $k=k_1$,
then $C=c_1$



پس اول می‌زنیم Rule 2 را که نتیجه‌ی این intermedia است برای کد

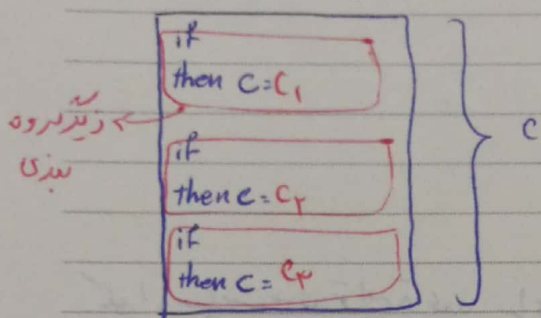
(backward) prompt می‌زنیم ببینیم $k > k_1$ است یا نه

Triangle می‌شود. Fire می‌شود. پس $C=c_1 \leftarrow k$ می‌شود. برای کد

به Rule 1 و چون satisfy می‌شود $D=x$ می‌شود و چیزی k inferred قرار می‌گیرد.

سه وقتی به $C=C_1$ رسم و جزو fact هستیم و prompt نداریم برای backward دنبال قاعده ای میگردیم که $C=C_1$ در نتیجه آن باشد. اگر برای $k=k_1$ prompt نداشته باشیم باید اول به دنبال قاعده ای میگردیم که $k=k_1$ در نتیجه آن بود.

۱۰. تا اینجا گروه ها را اولویت داریم.



$$\text{legal value } C = \{C_1, C_2, C_3\}$$

۱۲. سیاست داریم: قواعدی که محتمل ترین و احتمال برقراری

شرایطشان بیشتر است را زودتر میزنیم.

ترتیب در and هم مهم است. سی سیاست بعدی ترتیب (order) شرایط در شرایط مهم است و سرعت

نتیجه گیری را با هم برد مرتب کردن درست. سیاست دیگری این است که آن حد که میبینیم باید به نری

کنیم نه اینکه کار و نتیجه ای که میخواهیم بگیریم را باید به نری کنیم. از هر جنبه رویه را کسب می کنیم

۱۶. و آن رویه را باید به نری کنیم نه رویه خودمان. سیاست دیگری این است که از symbol ها استفاده

کنیم. به عنوان مثال نلیم اگر وزن بیشتر از ۱۰۰ بود دو تا کمر بنشینیم بلکه دانشی سنگین بودن را به $k.p$

افزافه کنیم و بکنج اگر بیشتر از ۱۰۰۰ بود سنگین است و اگر سنگین بود ۲ تا کمر بنشینیم. در این صورت

۱۸. اگر کمر بنشیند قوه تری تولید شود ما با تغییر یک خط همه چیز را به روز می کنیم.

۱۵. به دنبال قاعده ای میگردیم که $a=A$ در شرطش باشد. بعد به دنبال قاعده ای میگردیم که $A=a$ باشد

۲۰. در شرطش باشد. بعد به دنبال $A=0$ که intermedia باشد میگردیم و $A=0$ در نتیجه باشد

استفاده از جدول سرعت استخراج را با هم برد. هم همین برای گروه بندی و زیر گروه بندی که می کند.

اگر Rule ها درست هم و با فاعلی که از هم باشد خوب است. و اگر نه Rule ها در جدول

فصلی زیاری داشت معنی خوب کرده بنده نداریم.

۸

| Rule | Att. | Att. | valu |
|------|------|------|------|
| | | | |
| | | | |
| | | | |

۱۰

نتیجه ها

intermedia

در نتیجه

table به learning هم گفته اند. با توجه به مکانیزم جدولی هر Attribute و attribute در legal value

صفت می توانیم مکانیزم قانونی برای هر Attribute صفت و مشکل با نگاه داشتن صفت - ۱۲

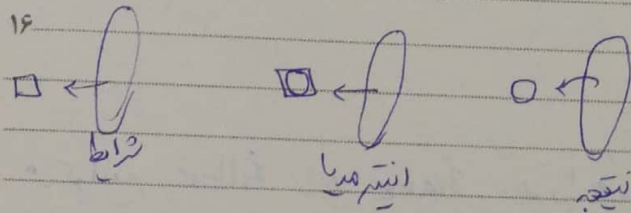
مثلاً $B=b$ مقدار داریم $\{B=b\}$ پس k کم داریم. می بینیم مقدار legal نیست یا نه داستان کم.

یکی از روش های سیستم خبره rule base است.

۱۴ با استفاده از جدول می توانیم واسطه کاری را به صورت menu در بیاریم چون می دانیم چه اتفاقاتی می افتد

در inference network، intermedia، واسطه های این. فلسفه ها هم اند. rule ها باید ذکر شوند.

لیم شبکه استنتاج



در شبکه استنتاج، or هم داریم ولی محققان or!

در نتوا بهینه شدن گفتن شبکه استنتاج حالت

۱۸ کد فای دارد و محققان راحت تر است. در شبکه استنتاج به سرعت می توانیم استنتاج را بهینه کنیم.

مثلاً وقتی B به قرار نیست، $R_1 \leftarrow$ discard می شود و با این کار همی قاعده هایی که B در شرط آن ها

است خود به خود discard می شود. $R_k \leftarrow$ Trigger می شود حال R_k را fire می کنیم و H .

به قرار می شود. اگر fire می کنیم می توانیم H را در شرط قاعده ی دیگری استفاده کنیم. \leftarrow forward

backward \leftarrow برای اینکه I به قرار باشد، R_2 باید fire شود. پس R_2 باید به قرار باشد.

۲۰ R_2 باید Trigger شود روز احسان و نیکوکاری

اولیای fire بودن را می‌کنیم بعد trigger بودن بعد درستی شرایط قاعده ← در backward

در inference network حرکت از ص به راست ← fan-out
 حرکت از راست به چپ ← fan-in
 backward

اگر یک قاعده fire شود نتیجه‌اش می‌شود در inferred knowledge. fact ماه ۱۰ ک و ۵ w.

می‌روند. w. 3. به دل inferred knowledge های درست ما هست. اگر یک قاعده discard شود دیگر
 نمی‌روند و نتیجه‌اش را در w. 3 قرار می‌دهیم.

Semantic Network یک سری از روابط OAR به صورت گرافیکی کامل شده‌های که گان یا زیر گان بودند.

اسکال این روش این است که node ها را می‌دانیم کدام گان است کدام زیر گان کدام object و

هم چنین نقش‌ها را می‌توانیم با این روش نتیجه‌گیری کنیم.

Frame یا قاب داستان‌های مرتبط را در کنار هم قرار می‌دهیم. Frame حافظه‌اند و Filler مقادیر اند.
 slot

شهادت حضرت فاطمه زهرا (س) (۱۱ هـ. ق) (تعطیل)

می‌توانیم Filler را در Frame درج کنیم که خودش یک slot است.

۱۸ می‌توانیم Filler را خالی بگذاریم. نمونه‌های مختلف از قاب می‌توان ساخت.

بسیار برای Frame می‌توانیم attribute هایی تعریف کنیم و بعد برای Frame نمونه‌هایی تعریف کنیم

که هر نمونه آن صفات را به ترتیب در یک object از Frame است در واقع.

۲۰ Filler ها با روشی های مختلف بدست می‌توانند بیایند یا اینکه by default باشند.

در شبکه معنایی هدف برقرار کردن رابطه بین گره‌هاست که در nlv استفاده می‌شود.

هدف از Frame، جمع‌آوری همه‌ی دانش‌ها در یک جایی است. در rule base قاعده‌های مختلف در جاهای مختلف

ordering همواره برای لایه نتیجه است. ordering خود روابط را درست می‌کند.

۸. یک دیدار از حسن های هم Frame ها وجود Facet است. یعنی می‌توانیم متد داشته باشیم. متد Facet کاری را انجام می‌دهد و روی property محدودیت می‌گذارد. if-needed میله در زمانی

۱۰. که به عدد آن property نیاز داری این کار را انجام بده و زمانی که مقدار آن property if-changed تغییر کند این کار را انجام بده.

روابط در قالب Generatization رابطه بین کلاس و زیر کلاس و همچنین object و instance و subclass و object.

۱۲. aggregation رابطه بین has-a و بخشی از

association رابطه بین دو object. متد رابطه یشارت می‌دهد که تو بکار ببری !!

۱۴. Knowledge Acquisition کتاب دانش مربوط به یک domain خاص

هدف: تولید K.b یعنی دانش‌هایی که به صورت priori هستند.

روش‌های کتاب دانش ← مصاحبه (interview) Rule induction ← با مراجعه به کتاب و استاد.

۱۶. برای مصاحبه فردی رایانه انتخاب کنیم که خبره است. افرادی که فقط کار را به انجام رسانده‌اند فرد

خبره نیستند. آن کسی که خبره است و تشخیص داده مد نظر است. فرد خبره رایانه قانع نگردد سیستم

۱۸. خبره به عنوان دستیار فرد به کاری رود. استخراج دانش خبره است درحقیقت اگر فرد خبره صورت تولید می‌دهد

درحقیقت کارش ممکن است از خبره‌هایی استفاده کند که به سیستم ما بیاید ولی در خارج از محیط فرد خبره دستگاه

ها و شرایط لازم برای نشان دادن دمو را باید بیاورد و ما متوجه می‌شویم در روز تقسیم ۲۰

چه خبره‌هایی برایش مهم است.

نوال: وقتی می‌خواهیم عمل مصاحبه انجام دهیم چه نکاتی باید رعایت شود.

مقدار ملاقات ها بستگی به توانایی مادر کسب دانش و توانایی فرزند دارد به احتمال دانستن دارد. مصاحبه کننده قبل از

مصاحبه باید اطلاعات در آن زمینه کسب کند و هم چنین روش بازخابی دانش را انتخاب کرده باشد و موافق باشد

مصاحبه به پیراهن کشیده نشود. ملاقات اولیه باید غیر رسمی باشد و زمان محدود باشد و جلسه گذشته نباشد.

۱۰. فرزند را باید با روشی بازخابی دانش منتقل است. اگر در هر دو ملاقات های بعدی

بعد از هر جلسه ملاقات هم چنین باید پیشرفت شود. ترفند های فرزند را حتماً باید کسب کنیم.

اگر مسئله از نوع دسته بندی باشد هر دسته نشانه هایی دارد. آن نشانه ها را باید پیدا کنیم و از فرزند بخواهیم راه های

درمان و رابطه ی بین آن نشانه و درمان را باید کسب کنیم.

اگر مسئله از نوع ساخت باشد از قسمت ها و قطعاتی تشکیل شده است. به دنبال کلاس و علائم و ارتباط بین آن

روش Rule Induction: زمانی که فرزند جزیره در دسترس نیست آنها مستعدی وجود دارد.


سیستم نشانه سازی هوایی مثل از نوع دسته بندی است چون هر حواس و ویژگی های خود را دارد.

با استفاده از مستندات

اول صفات خاص را مشخص می کنیم. بعد صفات خاصه هایی که در لغت نمی آید ما متون را می بینیم حذف می کنیم.

مثلاً آن هایی که مقدارشان در همه کسان اند یا آن هایی که مقدارشان را بدست آوردن سخت است. بعد باید از روی

۱۸ جدولی که ساختیم درخت تقسیم می سازیم. (برای هر نمونه مقدار صفات های خاصه را مشخص کردیم اینجا)

در درخت تقسیم هر  یک نود است و جواب هایش شاخه هایش است. leaf node ها صفاتی هستند

که نشانه سازی می شوند که class یا label (برچسب) اند. یکی از صفات خاصه به عنوان ریشه ی درخت انتخاب

می شوند. می توانیم درخت های تقسیم متفاوتی بسازیم برای این که در هر مرحله کدام صفات را انتخاب کنیم.

اگر از بزرگ که conclusion است به صورت backward به ریشه برسیم قاعده تولید کرده ایم و در مسیر

پیش از آن نودات if ما را تشکیل می دهند. پس درخت تقسیم در ساخت قاعده به من کمک کرده.

۸. attributes هایی که به کار می رود برای هر conclusion ساخته شود به درخت تصمیم و طبق conclusion.

۹. اگر درخت بهینه درست کنیم به سبب از دست دادن از بین می رود. با چه معیاری می باید درخت ساخت که

دانشمان از بین نرود؟ چقدر می توانیم برای decision tree همی ساخت نمونه ها می باید

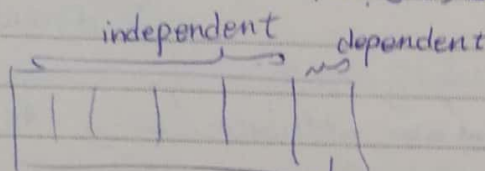
۱۰. مقدار داشته باشد. در همی سطح های خردی باید explanation چه داشته باشد.

ID3 Algorithm یک فرمول آنتروپی برای براس تعداد افراد ساخته شده. طبق صفت یا صدهای که کمترین آنتروپی

را دارد یعنی کمترین اطلاعات را دارد به کار رفتن آن attribute در تقسیم گیری داریم.

۱۲

درخت تصمیم یک روش machine learning با نظارت می باشد.



۱۴. یک طبقه بندی به ساخت جدول

۱۴

کوتاه ها

$H(C|A_k)$ آنتروپی \leftarrow می فرایم طبق صفت A_k به تمام صفت مربوط است.

۱۶

k امین صفت

۱۵. a_{k1} \leftarrow اولین صفت و اولین مقدار می دارد.

نتیجه اول

۱۷. $P(C_1|a_{k1}) = 3/4$ \leftarrow اولی صفت سوم و اولین مقدار می داشته باشد در کلاس اول

۱۸. برای همی صفات آنتروپی گرفته ایم. هر کدام آنتروپی این کمترین باشد اطلاعات ما از اینکه در تقسیم

گیری نقش مهم دارد و سبب می باشد بیشتر است.

۱۹. همیشه صفاتی که در level های بعدی هستند یکسان نمی شود. یعنی یکسان شده.

۲۰

براس این درخت تصمیم با یکاه قاعده می سازیم. با یکاه قاعده عمل می است.

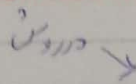
آنتروپی معیاری آماری است. معیار دیگری وجود دارد بنام Gini. الگوریتم ID3 براس درخت به آنتروپی درخت تقسیم می سازد.

در هر مرحله آنتروپی می گیریم. هر چقدر انتخاب شد جدول را براس آن تقسیم می کنیم و آن تقسیم را هم

در جدول های بعدی حذف می کنیم. اعجاز عبارات تاکیدی و مثبت را نادیده بگیرید.



forward



inferrence engine موتور استنتاج

موتور استنتاج برای استنتاج نیاز داریم . Rule / Premise table و Attribute - Queue table و w.m table

forward

بعد از اینکه دیدیم که rule علامت می توانیم اول unmark کنیم بعد discard یا اینکه عبارت unmark

کنیم ، mark کردن مسقیم discard کنیم نگهاری rule ها در Attribute - Queue Table explanation

۱۰

و بعد از آن که می کشد

وقتی که rule را انتخاب می کنیم تا شرط را بررسی کنیم آن را mark می کنیم. همزمان هم

۱۲ rule هایی که آن شرط را دارند بررسی می شوند. ممکنه با این کار یک قاعده که هنوز mark نشده شرط

برقرار نباشد و discard شود

۱۴ حسن AR table این است که attribute های یکسان پشت هم می آیند و می توانیم به ترتیب آنرا در موتور

۱۴

تست بردیم همه False است و discard می شوند آن rule ها

| rule# | # | close | Attr. | value |
|-------|---|-------|-------|-------|
| ۱ | ۱ | E.T | set | ۱ |
| ۲ | ۱ | E.T | prop | ۱ |
| ۳ | ۱ | E.T | prop | ۱ |
| ۴ | ۱ | E.T | prop | ۱ |

موتور
discard

می توانیم آن rule ای که اولویت بیشتری دارد fire شود یا نه

۱۶ اخیرا Trigger شده fire شود. چون از ordering

استفاده کنیم. یا اینکه تعداد کلزهای شرطی بیشتر باشد را fire کنیم

۱۸ در روش backward chaining به جای جدول Attribute جدول goal داریم و می mark

۱۸

unmark نیاز نیست. اگر rule ← Trigger شود همان را fire شود و می بینیم جوابی از خروج

BLAV است و می اندر حتی rule ← Trigger شود سیاست خودمان را می کشیم

۲۰ باید توجه کنیم که rule ی ما حتی backward دارد و کدام قاعده ما حتی forward

می کشی از آن و استفاده کنیم. در این حالت جدولان متفاوت می شود. موتور استنتاج

۲۳ k را به عنوان final goal و A را به عنوان fact می بینیم قبل از خروج k. اولویت در این مثال

人

دولتی رضا دایم : ۱۷ - رضا دینعلی

Ques 2 -
semantic

learning, ۷۸۲ بیرون از سیستم خبره به K.B وصل می شوند.

بررسی روش های حسابداری

عامل هر چیزی است که از جمیع اطلاعات را توسط حسگرها می‌گیرد و action انجام می‌دهد و به وسیله actuator ۱۲.

فرموله کردن هدف یعنی هدف نهایی چیست.

فصله کردن مسئله : در مسئله χ و معنی های χ و χ دارد ، χ را علی و χ را برای رسیدن به هدف

صفت موصوفی انتخاب بهترین دنبال از action حاض مغالطه حایر ایاریه به هدف.

مستطورات هزینه‌ها performance از بلاک، در agent است.

[illegible]

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

Covalent state, 11

مسیر دیناله ای از state های فضای حالت است.

action ها اینست ساخت عقای حالت می شوند

مسیر در اینجا از حالتی که ...

۱۸ ← مثلاً این عقده حالت است.

```
graph TD; A(( )) --- B(( )); A --- C(( )); B --- D(( )); B --- E(( )); C --- F(( ))
```

action ها به طور نا خود اگاه در مسیر هم محور دارند.

روشن حلیمان سیرج کردن بود با توجه به حقایق حالات سیرج کریم حاصل بودن یعنی راه حل تعیین کنند که قضا بود
راه حل بدیهه اگر وجود داشت.

blind search یا جستجوی ناگهانه فوق العاده است و فقط در صورتی که هیچ اطلاعاتی (information) اضافه نکرده باشد.

۲۴

کتاب و جوی عمقی - جست و جوی عرضی - DFS BFS

۸ مثلاً از این به بعد با درخت سروکار داریم چون در تلافی ممکن است ۱۰۰۰ داشته باشیم

۹ مثلاً در جست و جوی عمقی از A شروع می‌کنیم. به B می‌رویم. تست حرف اتمام می‌دهد، B حرف نیست. به C می‌رویم.

۱۰ بعد به D می‌رویم. بعد به K می‌رویم. بعد به F می‌رویم. در هر مرحله هم، بدون حذف می‌رویم. وقتی B را حذف می‌کنیم

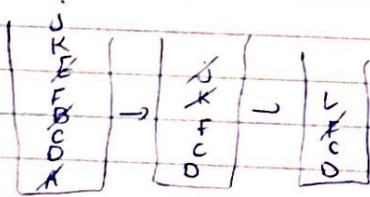
۱۱ C و D را حذف می‌کنیم یا صیف نه این‌ها معنای اسفندی را تشکیل می‌دهد. درجات مقادیر با بازه

به جواب می‌رسیم یا جست و جوی عمقی. مگر اینکه عمق درخت بی‌نهایت باشد. اگر هیچ کدام از فرزندان به جواب

۱۲ نرسید back track می‌کنیم و قبلاً را حذف می‌کنیم به همین ترتیب.

۱۳ مثلاً با جست و جوی عمقی با هزینه ۳ به L می‌رسیم ولی در جست و جوی عرضی با هزینه ۲ به I می‌رسیم. مسیر

۱۴ به این می‌رسیم هزینه کمتر در معموری است که برای L ABFL و برای I ADI باشد.



complexity می‌تواند در هر حال که در حالت نوشتیم و برداشتیم.

این روشی بهینه نیست. (DFS) چون back track زمان را

گاز می‌دهد.

۱۵ BFS هم کامل است و صحیحاً به جواب می‌رسد مگر اینکه مقدار فرزندان نامتناهی باشد.

BFS بهینه است یا نه؟ بستگی به هزینه‌ها دارد. اگر مسیرها هزینه‌شان برابر باشد صحیحاً بهینه می‌شود.

812

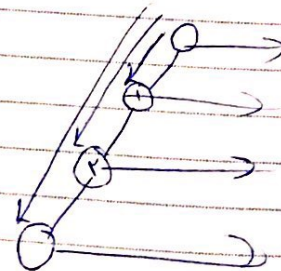
۱۸ DFS زمانی که بیشترین back track دارد.

۲۲ جست و جوی DFS و BFS را قاطعاً نکرده. پس صحیحاً به بهینه می‌رسد.

BFS



جست و جوی عمقی گنگاری



مطمئن شوید ابراز رضایت شغلی افراد به سبب ترس از توبیخ مسئولین و سرپرستان نیست.

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| ۳۰ | ۲۹ | ۲۸ | ۲۷ | ۲۶ | ۲۵ | ۲۴ | ۲۳ | ۲۲ | ۲۱ | ۲۰ | ۱۹ | ۱۸ | ۱۷ | ۱۶ |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

در بحث ۲ از ۵ به ۱ و بعد به ۲ می رویم بعد همان طه را visit می کنیم ولی در BFS حتی باید طه را به طه visit کنیم.

۱۰ اگر تعداد فرزندان در یک طه نامحدود باشد BFS گریه کند ولی حسب وجوهی بحثی گفته شد برای آن طه را در جوی ریزه در بحثی بعدی را انتخاب می کند. $DFS \leftarrow iterative\ DFS$.

از root شروع می کنند. بحث اول، اگر به جواب نرسیم و از $t+k$ بیشتر شد زمان اجرا پس با بحث دوم این کار را می کنند. یعنی در هر depth، BFS می زنیم.

۱۲ $DFS \leftarrow$ استیل $BFS \leftarrow$ صف $IDFS \leftarrow$ مخلوطی از هر دو

۱۴ وقتی صف داریم $EF \rightarrow CD$ وقتی استک داریم $CD \rightarrow EF$

پس در IDFS هم صف داریم هم استک. دو الگوریتم DFS و BFS را جدا داریم و مخلوط نمی سازیم.

یکی از راه های حل مسئله، حسب وجوهی باشد. حذف برگ های درخت اند.

۱۶ DFS و BFS و حسب وجوهی بحثی گفته شد برای حسب وجوهی ناگاهانه هستند.

صفت وجوهی ناگاهانه یعنی علاوه بر مسئله فرمول شده و ملاک های دیگری هم داریم.

BFS ما را به جواب بهترین می راند ولی DFS ممکن است نرساند.

۱۸ تابع جابجایی عملیات قابل انجام را به ما می دهد.

در BFS باید همگی گره ها را که داریم چرتن یا خودشان پدر اند یا فرزندان یکدیگرند. پس مقایسه کنیم بازه ای زیاد است. به هر گره می رسم فرزندان او را تولید می کنیم بعد به سراغ گره بعدی می رویم. همان عرض می رویم. هر چند در DFS بهترین نیست چون در بحثی بیشتر جواب را می یابیم.

روش دیگری نیز وجود دارد که برای DFS، بحثی کنیم و با همان بحث فقط DFS می زنیم. برای این روش

۲۰ $limit\ DFS$

کتابخانه سازمان را به روز کنید و اسامی کتابهایی را که اضافه می شود به صورت لیست منتشر شده در اختیار کارمندان قرار دهید.

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|
| ۱۵ | ۱۴ | ۱۳ | ۱۲ | ۱۱ | ۱۰ | ۹ | ۸ | ۷ | ۶ | ۵ | ۴ | ۳ | ۲ | ۱ |
|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|

این است که از گره ۴ به بعد درخت حذف می شود و نمایه جواب در عمق بیشتر از آن عدد باشد ولی خوبی این

روش این است که در درخت های با عمق بی نهایت گسترش کند.

در مرحله ۱ فقط A تولید می شود. در مرحله ۲ A و B و C و D تولید می شود. مرحله ۳ A و B و C و D و E و F

و G و H و I تولید می شوند و B و C تولید نمی شوند. ← IDFS

بهینه یعنی رسیدن به مسیری که کمترین هزینه را دارد یعنی مقدار $cost$ ها (از ریشه تا آن گره).

درخت و جوی دو طرفه: دو سرچ می کنیم همزمان. هر موقع صب و جوی از جلو و جوی عقب به هم

برسیم مسیر از start به goal می شود جواب مسئله.

حسب و جوی هزینه یک نواخت: چون اطلاعاتی علاوه بر فضای حالت داریم، اما حالا در نظر گرفته می شود که

مزدی که یا آن کمترین هزینه را دارد انتخاب می شود. ما هم این اطلاعات جزو مسئله کردن مسئله

در نظر گرفته می شود.

روز فناوری فضایی

حسب و جوی های ۱ تا ۵ دارند.

حسب و جوی اول بهترین

حسب و جوی محلی

درخت و جوی اول بهترین روش های حرصانه و A^* استفاده می شوند.

مثلاً $f(n)$ تابع ارزیابی است. $h(n)$ دهمین هزینه n در n گره به هدف

الگوریتم greedy از $h(n)$ استفاده می کند.

مثلاً ۳۲ گره C و D و E که هزینه ترین را بسط می دهیم. چون حرصانه نگاه می کنیم. اعداد $h(n)$

هستند. هزینه ها کمتری جزو صورت بوال اند. ✓ صحت منی گره بسط داده شده است. گره λ هزینه اش

تاریکترین به هدف = است پس هدف است.

ولادت حضرت زینب (س) (۵ هـ ق) و روز پرستار

برای ثبت ایده های درخشانی که ناکهان به ذهن می رسند، همیشه یک قلم و کاغذ به همراه داشته باشید.

۱۶ ۱۷ ۱۸ ۱۹ ۲۰ ۲۱ ۲۲ ۲۳ ۲۴ ۲۵ ۲۶ ۲۷ ۲۸ ۲۹ ۳۰

۸ BCGNX را نده میاریم. چون C فرزند B نیست پس B جزو مسیر رسیدن به هدف نیست. به سراغ هر کد

که هم رسم فرزندانش را بسط می دهیم.

چون تنها معیار رسم گیری این تخمین فاصله تا هدف است پس بهینه نیست. پس اگر حدی هدف داشته باشیم

ممکن است به جواب بهینه نرسد.

۱۰ کامل نیست چون ممکنه به کد دیگری برسیم و اگر نه صفا به جواب می رسم.

حساب و جوی A^* اعدادی $f(n)$ حساب کرده اند.

۱۲ در یادمان بازی بهتر است وقتی نودها را در استک می ریزیم به ازای $f(n)$ مرتب کنیم. هر موقع بسط می دهیم

باید مرتب کنیم. به هر حال در مرحله ی بعد باید کمترین $f(n)$ را برداریم.

۱۴ هم نگاه میکنیم کمترین را آلوده یا به دهم کمترین را برود

در سینه به خاطر همین به جواب بهینه می رسد.

تایم ها نسبتی که مورد نظر را بسط می دهد.

A^* همواره بهترین حالت را می دهد.

برای معمای ۸ روش A^* نیاز به heuristic برای تسهیل h دارد.

کاملاً کار می کند

$$18 \quad h = 3 + 1 + 2 + \dots$$

۵۳

در جست و جوی محلی مسیر رسیدن به هدف مهم نیست. حتی جواب های نزدیک به optimal هم خوب هستند برایمان.

در جست و جوی گلوبال هر نقطه ای که در فضای حالت بهتر از نقطه ای ما است به آن می رویم. در سینه مورد

۲۰ این جست و جوی گلوبال است. این روش جست و جوی در نقاط local گیر می کند ولی ممکن است به Global نرسد.

پس باید روش هایی مانند hill climbing (گرفتن در local ها) را به طرف گلوبال یا hill climbing

بهتر دوری نباشد. یا اینکه هفت و آنکه قدم بردارد از فضای مسئله این خارج می شود.

برای حل این مشکل یک راه این است که به جای اینکه نقاط دور خود را نگاه کنی، کل نقاط را نگاه کنی و جمع بگیری.

اینکه ببینیم رفتن به سمت یکدیگر حقیقتاً هزینه دارد یعنی حساب و جبری است یا نه.

تا اینجا مسئله را به درخت تبدیل کردیم، اگر بخواهیم دیدگی داشته باشیم از روش‌های حساب و جبری local استفاده می‌کنیم.

یکی از روش‌های حساب و جبری الگوریتم‌های تکاملی (ژنتیک) هستند.

مثلاً به سدی نقاط اولیه داریم، بعضی‌ها بسیار را انتخاب می‌کنیم، این ژن‌ها با هم ترکیب می‌شوند و نسل را به وجود می‌آورند. ^{جست} Mutation و ^{برش} Cross Over عملیاتی اند که این‌ها می‌توانند روی ژن‌ها تا نسل بعدی تولید شود.

۱۴ تا به بیاری می‌رسد این تابع بهینه می‌شود کدام نقاط از فضای انتخاب می‌شوند؟ اگر این کار ادامه پیدا کند تا به جواب برسیم بعد از چندین نسل، حتماً دارد تکامل پیدا می‌کند به همین دلیل روش تکاملی نامیده می‌شود. GA یک مسئله تکاملی که هدفش است.

۱۶ **Cross Over** یعنی یک نقطه تصادفی را در طول رشته انتخاب می‌کنیم. نسبت به نقطه‌ای که انتخاب می‌کنیم تغییراتی را در رشته اولیه ایجاد می‌کنیم و آن را رشته‌ای نو می‌توانیم بنویسیم.

۱۸ **mutation** یعنی یکی از رشته‌ها را (پدر یا مادر) را انتخاب می‌کنیم و در آن جهشی ایجاد می‌کنیم.

معمولاً هر دو عمل در الگوریتم ژنتیک انجام می‌شود.

۲۰ مثلاً در مسئله ۸ وزیر ژن‌ها ۸ کد می‌زنیم. مسئله ۸ مقدار است. در هر رتبه قرار می‌گیرد تا به بیاری می‌رسد.

۲۰ صلیب اولی والدینی که کمترین جمله به هم دارند به‌همین مورد اند برای ما حتی فرزند از آن‌ها.

اگر بعد از اولین نسل به جواب رسیدیم می‌توانیم اکتفا به متوقف می‌شویم و نسل بعدی تولید نمی‌شود.

人

1.

15

19

19

1A

۲.

—

•

سیستم توصیه‌گر مشکل من این است که cold start است یعنی شروع ندارد. مثلاً فیلم جدیدی بپایید که هیچ‌کس آن را ندیده یا اینکه یک کاربر جدیدی بپایید که هیچ کاری نداشته باشد تا الان.

روش سوم مشکل کاربر جدید را ندارد. روش اول مشکل کاربر جدید را دارد. روش دوم مشکل سیستم جدید را دارد.

Constraint Satisfaction Problem. مسائلی که با یک سری شرط همراه هستند.

دسته CSP یک سری متغیر x داریم که می‌خواهیم مقدار دهی کنیم و باید نت گسسته که آن محدودیت‌ها به

وجود نیامده باشد. می‌توانیم با متغیرها مسئله را فرموله کنیم و از یک سری الگوریتم استفاده کنیم.

۱۲ به عنوان مثال map coloring، محدودیتش همین است که هر دو رنگ هم نمی‌توانند هم‌رنگ باشند. در

CSP نسبت‌ها با متغیرها معلوم می‌شود. domain همان سبدها هستند. مثلاً اعداد باید گراف تشکیل

۱۴ دهیم. لیکن حالتی می‌رسد که هر دو هم‌جوار نیستند. نودها سبدها هستند و محدودیت‌ها حالتی

بین دو نود باشند. binary اند و روی یک نود unary است و می‌تواند تعداد بیشتری نود را ملاک

کند.

۱۶ مثلاً مسئله CSP است و این حالت جمع می‌شوند باید درست جمع شوند به شرط

Auxiliary Variable متغیرهای کمکی هستند به عنوان carry

۱۸ حروف O C F محدودیت‌ها این است که اعداد و کسان را نمی‌توانیم به آن نسبت دهیم.

طریقی. \square یعنی محدودیت که باید با هم جمع شوند. \square سطر اول یعنی این محدودیت که اعداد متفاوت به آن هانسیب

داده می‌شود. \square طریقی یعنی چگونگی درست باشد.

۲۰ محدودیت مسئله را پیچیده می‌کند. مثل Δ و زیر یک سری محدودیت یا قید دارد.

۲۱ R باید R باشد تا محاسبه درست باشد. x هم تولید می‌شود پس x هم در محدودیت می‌آید.

تا صحت و سقم مسائلی روشن نشده، کسی را مواظده نکنید.

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| ۳۰ | ۲۹ | ۲۸ | ۲۷ | ۲۶ | ۲۵ | ۲۴ | ۲۳ | ۲۲ | ۲۱ | ۲۰ | ۱۹ | ۱۸ | ۱۷ | ۱۶ |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

مشتبه چون `fast` می‌نویسد `back track` می‌کند.

در مرحله داریم این حرکت را `enum` می‌کنیم. در آن `enum` حساب می‌کنیم چقدر کنیم.

در `enum` حساب می‌کنیم چقدر `enum` می‌کنیم. در این `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

`enum` می‌کنیم. `enum` می‌کنیم. `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

`enum` می‌کنیم. `enum` می‌کنیم. `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

در `enum` `enum` می‌کنیم چقدر `enum` می‌کنیم. در `enum` `enum` می‌کنیم.

Planning

۸ روشی از عملیات که این عملیات بست هم انجام می شوند تا به هدف برسیم. محاسباتی که باید

Deterministic, fully observable, محیط کاملاً static است و تغییرات فقط با action که agent انجام می دهد

۱۰ روی محیط ارجح می دهد و هم چنین گسسته در زمان است.

Search Based agent روی یک کد که به جا می آید و از یون انجام می دهد تا به جواب برسد.

در agent, planning یک داری نگاه می کند.

۱۲ اهداف مسئله را می توان conjunct کرد. به بیان end گذاشت.

برای تعریف کردن state می توان از یک ری logic استفاده کرد. state ها و در action ها را باید بتوانیم

۱۴ تعریف کنیم برای planning, از predicate برای این کار استفاده کنیم.

معنی زبان ها func. ها به عنوان predicate نفر می بینند. است هایی که منجر به صرف

می شوند را هم می توانیم در تعریف اهداف از ری logic قرار دهیم. اگر یک state قرار

۱۶ یابد goal شود باید atomic های در میان یابد.

precondition یعنی چه در ادیتی باید برقرار باشد تا action انجام شود. برای اینکه action انجام شود
این ها باید بررسی شوند. ۱۸ effect تأثیرات action روی محیط است.

At(p, from) یعنی p باید در آن موجود باشد. p باید هوایی باشد و from باید

منزود باشد.

۲۰ برای هر action باید precondition و effect را مشخص کنیم.

state ها با data structure مشخص می شوند در search

در سرچ, plan همان دنباله است که از است اولیه به هدف وجود دارد.

سالروز حماسه مردم آمل

سعی کنید اسامی کارمندان را به خاطر بسپارید و در حین صحبت کردن با آنان، اسمشان را به زبان بیاورید.

۱۲ چون بین شرایط این action برقرار است، Fly می تواند انجام شود. و بهر استیجایی که می شود.

۱۳ $At(p)$ یعنی م باید یک مکان باشد. بعد از اینکه x را خریدی ماکت آن هستی $\leftarrow Have(x)$

زبان STRIPS از $Mean-ends\ analysis$ استفاده می کند. در هر مرحله استیج را از هدف کم می کنیم

۱۰ و آنکه اختلاف ضرر بود یعنی جواب سببی.

در این زبان effect را در قالب $addlist, deletelist$ مطرح می کند یعنی fact ها، اضافه

یا کم می کند. هر چیزی که بیان نمی شود را $False$ در نظر می گیرد. در این زبان type یا = نمی توان گذاشت.

۱۲ به همین دلیل هم چیز را باید Predicate تعریف کنیم.

زبان ADL چیزهایی که تعریف شده را unknown می گوید. برای اوصاف هم می توان Quantifier

۱۴ استفاده کرد. (Quantifier) امکان تعریف $same-place$ را می دهد (۱۵)

۱۵ می توان گفت اگر P برقرار باشد E برقرار می شود. می توان تأثیر یک action را در بعضی مواقع تعریف

کنیم و همیشه باشد. امکان تعریف type را هم می دهد.

۱۶ مث هدف این است که C در فردگاه فلان و C در فلان جا باشد.

مث برای اینکه action انجام شود باید بین شرایط برقرار باشد.

۱۸ یک معموله از یک فردگاه می تواند به فردگاه دیگر برود با این predicate ها.

۱۹ مث این $plan$ ما برای گدن هدف است.

۲۰ مث جایگاه قطعات:

۲۱ $clear(A)$ یعنی روی قطعه A چیزی قرار نگرفته باشد.

$On(A, B)$ یعنی A روی B است.

$Block(A)$ یعنی A یک قطعه است.

اولیه :

| | | |
|---|---|---|
| A | B | C |
|---|---|---|

 table

هدف :

| | | |
|---|---|---|
| A | B | C |
|---|---|---|

 table

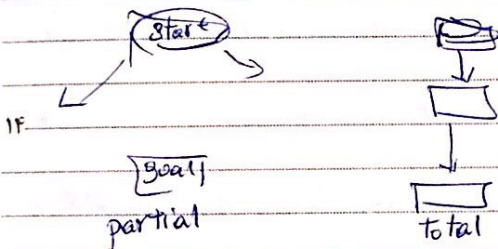
۲۳ ص ۸ هدفن کرده ایم table به اندازه کافی بزرگ است. بعداً clear table است. اگر نبود باید این کارها

را هم بزاریم. یاد می تواند table با block

۲۳ ص ۱۰ ADL می توانیم بنویسیم ولی با STRIPS نه.

- الانجا total plan را گفتیم. در Partial Plan، به پیری order وجود دارد. ترتیب action ها بر این مهم است.

۲۴ ص ۱۲ باید وقت کنیم که precondition action دکشن بعدی را نقض نکند.



۲۳ ص ۱۶ op3 باعث شود effect آن Prelo. ۲ را تغییر دهد و نتواند. Prelo. ۱ آن برقرار باشد.

در آن صورت op2، op3 را بهتر می کند. و می توانیم ordering سخت راست را انجام دهد.

نیزهای bold حالت exclusive دارند یعنی در هر لحظه فقط یکی از آن ها باید برقرار شود یا انجام شود. (mutex)

۱۸ لنید مربع داریم این شرایط اولیه با حالت آخری برابر است.

state ها و action ها و goal باید بازبینی شوند. با استفاده از Predicate

ص ۱۶ action می توانیم انجام دهیم. یکی انتخاب می کنیم، current state مشخص می شود. اگر به هدف

نرسیدیم یک action دیگر را انتخاب می کنیم عبار back track.

STRIPS از روی کور، کورانده استفاده می کند یک هیوریک به کار می رود به نام Mean ends

STRIPS اختلاف است کوفی را goal را حساب کند ، action ای را انجام می دهد که این اختلاف را کم کند. چون C clear نیست action ای را انجام می دهد که C clear شود ، effect حالتی اثر ب را روی A بزرگ برای خالی کردن C ، چون precondition $moveOn(C, A)$ را ضایع می کند پس backtrack می کند و B را روی میز می گذارد. هر موقع plan را ریخت شروع به اجرای کند. backtrack می کند یعنی می فهمد که این جزو plan نیست.

uncertainty :

تئوری احتمال : در این روش درجه ی باور را با احتمال مشخص می کنیم. $P(X=A)$ متغیر تصادفی
 روش fuzzy ، CF در نظرسنجی که دو بار دارد.

مث ۱۴ $P(A)$ درجه ی باور به A است. unconditional

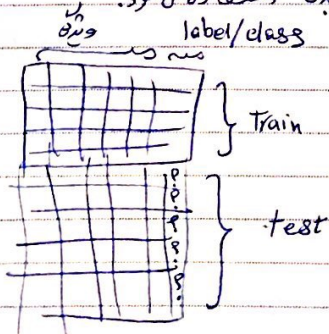
مث توزیع احتمال روی متغیر weather unconditional

conditional یعنی حتماً B اتفاق افتاده در صورت اتفاق افتادن A

exclusive یعنی XOR یعنی همزمان نمی تواند رخ دهند.

در یادگیری ماشین ، هدف بستن این است که از روی داده هایی که در سیستم هست ها را کند.

از base و تئوری احتمالات برای classification یا دسته بندی استفاده می شود.



از همین داده ها استفاده می کنند تا گمان کنند که این ها را

نمی بیند. است classification label دارد ولی

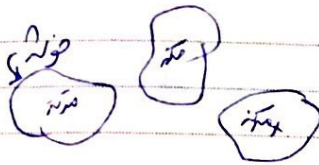
خوب ها label ندارند. خوشه بندی یا clustering

می کنیم و با روش های توزیع احتمال سببیت های داده ها

می کنیم و داده های شبیه به هم را یک خوشه می گذارد.

با قدردانی به موقع از کارمندان، انرژی کاری آنان را افزایش دهید و حسن خلاقیت را در آنان تقویت کنید.

| | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| ۱۶ | ۱۷ | ۱۸ | ۱۹ | ۲۰ | ۲۱ | ۲۲ | ۲۳ | ۲۴ | ۲۵ | ۲۶ | ۲۷ | ۲۸ | ۲۹ | ۳۰ |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|



بیشتر
هر داده نسبت به کدام متر است، پس مال خونریز است.

شبکه ی Bayesian:

یک شبکه جهت دار دوره ای است. به تعداد متغیر رندم داریم که نودهای گراف هستند. اندوخته ها و اسبک
باشند با یک شبکه جهت دار به هم وصل می شوند.

۱۳ earthquake و Burglary بران alarm اند یعنی حتماً باید زمین لرزه اتفاق بیفتد و

۱۲ (احتمال) P آن ها وجود داشته باشد تا alarm رخ دهد. اما به علت خطای سیستم است.

مثلاً خطا از چون parent این فقط Alarm است این اتفاق می افتد.

۱۴ جهت حرکت به سمت شرایط احتمال است که احتمال شرطی را به هم وصل می کند.

$$P(B \text{ و } A \text{ و } E) = P(B) + P(E) + P(A|B)P(A|E)$$

درجه ی باور بالا می رود تا تاثیر یکی از دو والد را در یکی می کنیم

۱۶ اگر $P(A|B \text{ و } E)$ می دانستیم درجه ی باور را بین می آمد.

۱۹ صد احتمالاً داریم می گوییم $B \text{ و } A \text{ و } E$ است پس همین را می نویسیم وانی اگر می گفت E و B و

۱۸ باید جدا جدا هندس می کردیم

Game Playing

پایزای لا می توان به عنوان یک مسئله جهت صورت می گیرد. $Terminal$ مصحفی که کنونی بازی به اتمام می رسد.

۲۰ $Utility$ Fun. یک عدد است که به هر $state$ داده می شود و مصحفی می کند در هر مرحله بعد می کنیم.

$Game$ tree برای بازی های دو نفره است. هر $level$ حرکت یکی از دو بازیکن است

بند ها باخت و برد یا تساوی هستند.

خوش رست و خوش بیان باشید و در جمع با انرژی و اشتیاق حاضر شوید.

۱۵ ۱۴ ۱۳ ۱۲ ۱۱ ۱۰ ۹ ۸ ۷ ۶ ۵ ۴ ۳ ۲ ۱

مادربرد بازی‌هایی هستند که rational است یعنی هرگز عقلانی و خوشه‌ای نیست.

بازی zerosum است یعنی اگر من بردم حریف باخت.

الگوریتم minimax :

۱۰. تابع evaluator یک عددی را به هر leaf node می‌دهد. max و min را امتحان می‌کند.

حین بازیکن می‌خواهد بهترین امتیاز را بگیرد و در لول بعدی که می‌خواهد امتیاز

بازیکن را min کند. امتیازی که در رسته است بهترین امتیازی است که می‌خواهد در برابر

۱۲. حریف عقلانی می‌تواند بگیرد.

مثلاً اگر می‌خواهد بازی کند بهترین امتیاز را می‌خواهد پس اگر در اولین مرحله حرکتی انجام دهد

۱۴. که به صورت زیر باشد را سعی می‌کند برد امتیاز بیشتری بگیرد.

دیدیم که روش‌های سرچ برای بازی اصلاً خوب نیست. بازی α با این روش خیلی دقت بیشتری برای

باید بررسی می‌کنیم.

جمعه

۱۴۳۸ / ربیع الثانی / ۲۱

هفته ۴۴

شماره

20 / January / 2017

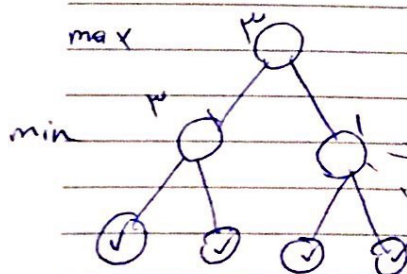
۳۰۷ گذشته / ۵۹ مانده

Alpha-beta Pruning

برای دو مقایسه و β به معنی کمترین و بیشترین

۱۸. راه‌ها را می‌کند و مورد ارزیابی قرار می‌دهد. ایده این است که اگر حرکتی به طور واضح بد بود آن را حذف

۲۰. حرکت کن و ادامه ده. α به بیشترین مقداری که تا الان می‌بینیم و کمتر از β کمتر نیست.



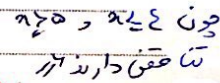
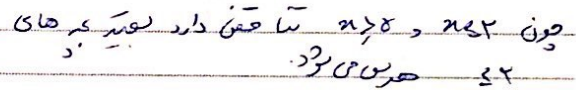
این‌ها را Prun می‌کنیم

چون می‌فایده‌ای اند.

در موقع امضا کردن نامه‌ها و مکتوبات آنها را به دقت مطالعه کنید و از امضا کردن آنها، زمانی که حوصله و تمرکز ندارید پرهیز کنید.

۳۰ ۲۹ ۲۸ ۲۷ ۲۶ ۲۵ ۲۴ ۲۳ ۲۲ ۲۱ ۲۰ ۱۹ ۱۸ ۱۷ ۱۶

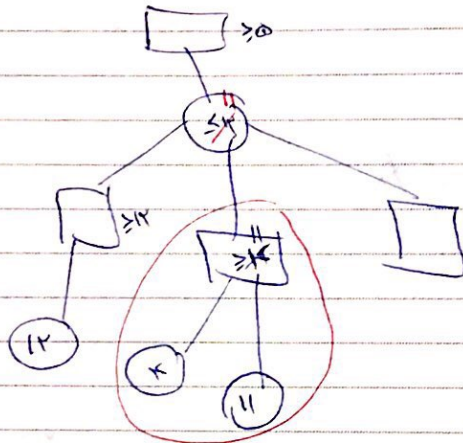
人



ب را اول برنامه می کشد و α را β می کشد.

اگر به جای (۳) داشته باشیم: (۱۱) در آن صورت چون α ۱۲ و β ۱۱ است و پدرمان

من خورد پس در بعد از (۱۱) فرزندان دیگری هم داشته باشد پدر من خورد و هر دو می کشیم.



چون α ۱۲ و β ۱۱
تأیید ندارد
فرزندان α ۱۱
هم می کشد.

حق کل این جد ۱۱
مردی که در پدر چپین

حالا وزن های بهینه در مسئله داده می شوند.

۳ یعنی ارزیابی ها پس در آن ها وجود دارد. روش Pruning که مطرح کردیم پس در آن ها وجود

۱۸ ندارد. می توانیم یک مقدار احتمالی در نظر بگیریم که شانس را وارد کنیم. یعنی node ها سه نوع

اند: min و max و chance. تابع ارزیابی به هر leaf node یک عدد می دهد.

Communication

۲۰ agent ها نیاز دارند که با هم ارتباط برقرار کنند.

تا قبل از ۱۹۵۳ دیدگاه کلاسیک داشتند، خواهی به قرار گرفتن ارتباط با جملاتی درست یا غلط در نظر نمی گرفتند.

و این نیز به تعامل داریم چون در action که agent که می خواهد انجام دهد تأثیر دارد است.

۸

communication و تعامل در multi-agent مطرح است.

Speech ممکن یکی از اهداف را داشته باشد: قول بدهد - تصدیق کند - دستور دهد - پرسش کند - اطلاعاتی بدهد.

۱۰. به سبب آنکه داد برای بیان سخن وجود دارد - موضوعیت باید بداند -

مثلاً ۲ ضمیمه است که می خواهیم اطلاع رسانی کنیم. ۱ سخن خود را شنیده است.

شنونده باید حرف های مرا تأثیر کند و مفهوم سخنم را بفهمد.

۱۲

اگر خود گوینده حرفش را باور نداشته باشد یا بلد نباشد - اگر به سبب اشتباهات در سخن بدهد

اطلاعاتی که در سخن وجود دارد - محتوایی که گوینده می گوید با محتوایی که شنونده دریافت می کند متفاوت

۱۴

باشد - این ها باعث می شود در سخن اشتباه رخ دهد.

به زبان فریبان و گفتار یا این ها که دارد.

NP عبارت اسم و VP عبارت فعلی است.

۱۶

۱ تا ۵ یک عبارت اسم در کنار یک عبارت فعلی است. نام ۲۰ کافی است.

هوش مصنوعی: agent بتواند چیزهایی را یاد بگیرد.

تواند ذخیره کند و باز یابی کند.

تواند تصمیم گیری کند و از اطلاعاتی که ذخیره کرده با توجه به شرایط تصمیم بگیرد. ۱۸ خود سازمانده یعنی اگر هر چند چیزی داشته است بتواند اطلاعاتش را تصحیح کند.

الگوریتم machine learning یعنی توانایی که انسان دارد و کاری که انسان می تواند انجام دهد

۲۰

ماستین انجام دهد.

Supervised یعنی یادگیری با نظارت یعنی قبلاً به جا Label خورده. یعنی دیگر را

کامل داریم با Label اس.

شهادت نواب صفوی، طهماسبی، برادران واحدی و ذوالقدر از فداییان اسلام (۱۳۳۴ هـ ش)

از سرزنش کردن دیگران در جمع خودداری کنید.

۱ ۲ ۳ ۴ ۵ ۶ ۷ ۸ ۹ ۱۰ ۱۱ ۱۲ ۱۳ ۱۴ ۱۵

در $unSupervised$ دیتاها $lable$ ندارند.

۸ $SemiSupervised$ کتب از دیتا $lable$ دارد و کتب دیگر ندارد.

$Reinforcement$ یعنی $action$ را که انجام می‌دهد با دیتا یا خبر می‌دهد و یاد می‌گیرد که کدام $action$ خوب است و کدام بد.

$Evolutionary$ یعنی همان الگوریتم‌های ژنتیک. که در حوزه بهینه‌سازی خیلی کاربرد دارد.

$Deep$ فقط تعداد لایه‌ها را خیلی زیاد می‌کند که $SemiSupervised$ در این حالت استفاده می‌شود.

۱۲ اما خطرات این می‌گردد و دقت سیستم‌ها را بسیار بالا می‌برد. خودی هر لایه به لایه دیگر فیلتر می‌کند.

این حادثه‌های مختلف الگوریتم در ماشین $learning$ بودند.

۱۴ $synapse$ از هر کدام از $learning$ ها که حافظه اند که در آن‌ها واقع است. بودن از طریق

$Dendrite$ ورودی می‌شود و اگر اختلاف پتانسیل از th بیشتر شد یک سیگنال می‌فرستد از طریق

$axon$

۱۶ مثلاً هر نورون از یک لایه به همه نورون‌های لایه بعدی وصل است و می‌تواند فرستد که این کار می‌کند پس

$Fully connected$ است. تعداد لایه‌ها و تعداد نورون‌ها در تقسیم‌گیری نقش دارند.

۱۸ اولین شبکه عصبی ۴ لایه است و بعدی ۳ لایه.



۱۹ مع دو لایه است.

مثلاً ورودی ۲ و ۲ و ۲ می‌دهیم خروجی ۸۶ می‌دهد.

۲۰ خاصیت ۶ نوردهی پس ۸۰ می‌شود خطای ما. از ۸۰ چیزی کمتر می‌شود $backward propagation$

هر کسین یعنی به سمت برگ در این خطا را w^2 را طوری اصلاح می‌کنیم که خطا کمتر می‌شود.

دو باره ورودی دهنده می‌دهیم و همین عملیات $learn$ کردن را انجام می‌دهیم.

فرار شاه معدوم (۱۳۵۷ ه.ش)

خشکی جلسات طولانی را با شوخ‌طبعی قابل تحمل کنید.

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| ۵۵ | ۵۴ | ۵۳ | ۵۲ | ۵۱ | ۵۰ | ۴۹ | ۴۸ | ۴۷ | ۴۶ | ۴۵ | ۴۴ | ۴۳ | ۴۲ | ۴۱ | ۴۰ | ۳۹ | ۳۸ | ۳۷ | ۳۶ | ۳۵ | ۳۴ | ۳۳ | ۳۲ | ۳۱ | ۳۰ |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

۸ استفاده از این کار را می‌کنیم که شبکه هم یاد بگیرد یعنی learn نه و شبکه را حفظ می‌کنیم.

چون می‌دانیم جواب این ۱۶ است ولی شبکه گفته ۸۶، به همین دلیل بهش گفته و وزن‌ها را تصحیح می‌کنند.

۱۱ الگوریتم‌های evolutionary برای تنظیم وزن‌ها مورد استفاده قرار می‌گیرند. synaps ها همان وزن‌ها هستند و learning روی این وزن‌ها انجام می‌شود.

۱۲ الگوریتم‌های روبرو به بحث مثل بازگشتی خیلی زمان بردارند ولی در این الگوریتم‌های Deep یا

evolutionary روبرو به بحث نمی‌رویم پس سرعت خیلی بالاست.

در انتها وزن‌های جوری تنظیم می‌شود که کل داده‌ها را می‌تواند حل کند و واحد می‌شود.

۱۴ حد حدی ندارد و خودهای هر لایه بیشتر یا کمتر شبکه را تغییر می‌دهد learn می‌شود.

شبکه خودش باید آموختن بپذیرد و داده‌ها را label بزند.

۱۶ مثل چون روبرو به حلوانست به آن feed می‌گویند. (براد بزرگ) backward propagation

دارد این است که error که جزئی گرفته شود برنگردد به مرحله‌های قبل می‌error کوچک شود

کوچک‌تر می‌شود و در مرحله‌های اول دیگر چیزی از error باقی نمی‌ماند.

۱۸ معمولاً ۸۰ درصد داده‌ها برای آموزش و learn استفاده می‌شوند و ۲۰٪ داده‌ها

برای تست استفاده می‌شوند. و تنظیم وزن‌ها

اول با هزاران قدم می‌دهد، آنقدر آن‌ها را بعد تغییر می‌دهد تا به خطای معقول برسد.

۲۰ Fuzzy logic یعنی نمی‌توان گفت هست یا نیست، بلکه می‌توسیم. همان در حد هست. همان در حد نیست.

۱۳ بخواند است چه منطوق boolean است و کمیت راست منطوق فازی

تاریخ: ۱۲ / January / ۲۰۱۷
 شماره: ۶۷ / مانده
 ۲۹۹
 تاریخ: ۱۲ / January / ۲۰۱۷
 شماره: ۶۷ / مانده
 ۲۹۹

۲۴

جمعہ

١٤٣٨ / ربيع الثاني / ١٤

۴۳ هفته

۵۵

13 / January / 2017

۳۰۰ گذشته / ۶۶ مانده

هیچگاه اجازه ندهید کسی حالت افسردگی و ناامیدی شما را ببیند.

٣٥ ٣٤ ٣٣ ٣٢ ٣١ ٣٠ ٢٩ ٢٨ ٢٧ ٢٦ ٢٥ ٢٤ ٢٣ ٢٢ ٢١ ٢٠ ١٩ ١٨ ١٧ ١٦