

Utredningsplan för Liu Idsson (Liuid123)

Allmän information om mallen

Detta är en mall för att skriva en utredningsplan i kursen TDDD92-2020. Mallen beskriver de olika avsnitt som ska finnas och hur man ska fylla i dem. Normal storlek är 1–1.5 sidor.

Vi har också gjort en noggrann genomgång av de kommentarer vi har behövt ge tidigare år, och fyllt på mallen med det vi ser att ni som kursdeltagare kan behöva tänka extra på. Vi hoppas att detta ska vara till hjälp och inte låter alltför överväldigande. Meningen är helt enkelt att ni ska få en rimlig chans att veta vad vi är ute efter och åt vilket håll ni ska sikta, så att ni förstår vad vi menar med t.ex. “problemställning” – inte att allt måste vara fullständigt perfekt från början, även om ni så klart ska ta uppgiften på allvar.

De numrerade rubrikerna nedan ska vara kvar och ska inte ändras, men alla instruktioner (inklusive denna och “allmän info”-rubriken) ska så klart tas bort innan inlämningen.

För att undvika att skriva en alltför luddig utredningsplan är det bra om du tänker dig att du beskriver ett arbetsområde för *någon annan*, som är den som faktiskt ska genomföra arbetet utan att behöva kontakta dig för att reda ut detaljerna!

På <https://www.ida.liu.se/~TDDD92/2020/utredning.shtml> får du mer information kring utredningen.

1 Problemställning

Här ska du beskriva ett tydligt och välmotiverat **problem man vill lösa** i StarCraft.

Tänk på att problemet *inte* är att “jag vill applicera den här *tekniken*”. Istället handlar det om att du har ett *problem* i StarCraft: Du vill se till att agenten kan göra, åstadkomma, bestämma, ... X för att den ska kunna spela bättre. Vad är det för *förmåga* du vill ge agenten?

Tydlighet. Det är viktigt att vara tydlig och konkret. Det räcker inte med att man vill att “agenten ska vinna oftare” eller att “marines ska överleva längre” – det ligger på en så hög nivå att det inte ger dig någon särskild vägledning och då blir det svårt för dig själv att styra din utredning åt rätt håll. Du behöver komma fram till en mycket tydligare problembeskrivning än detta. Vi upprepar: Tänk dig att du ska lämna det här arbetet till *någon annan* och att de verkligen behöver förstå dina tankar – så tankarna får inte stanna kvar i huvudet utan behöver komma ner på papper.

Definera dina termer och begrepp. Tala om i vilka situationer problemet behöver lösas. Tala om vilka antaganden du gör om världen. Om du vill välja strategi, tala om vilka al-

ternativa strategier du har tänkt dig (du kan ändra dig i det senare utredningsarbetet, så klart).

Tänk på indata och utdata/resultat. Vilken information tänker du dig kommer att finnas tillgänglig? Vilken information tänker du dig ska skapas? Man kan också lyfta upp “indata och utdata” en abstraktionsnivå och prata om *vad man behöver veta* och *vilken information eller vilka beslut man får från lösningen*. Man kan också tänka sig att detta sker *statistiskt* (en uppsättning indata ges till en funktion som så småningom returnerar utdata) eller dynamiskt över tiden (problemet är att ständigt övervaka vissa delar av världen, “indata”, för att kontinuerligt uppdatera information om någonting eller kontinuerligt ta nya beslut, “utdata”).

I det mycket generella problemet *automatisk planering* är indata (a) information om exakt vilka handlingar som kan utföras och deras precisa exekveringsvillkor (preconditions) och effekter, (b) fullständig information om världens nuvarande tillstånd (vilka fakta som är sanna och vilka som är falska), och (c) ett målvillkor uttryckt i termer av vilka fakta som ska bli sanna efter att en lösningsplan har exekverats. Utdata är en sekvens av utvalda handlingar sådan att om världen är i det angivna tillståndet, och sekvensen exekveras i den angivna sekvensen, kommer målvillkoret att bli sant i det resulterande tillståndet. Vad är tänkta indata och utdata i ditt problem? Om du inte vet det har du kanske inte tänkt genom problemet tillräckligt ännu, och då är det svårt att hitta bra lösningar.

Ditt problem behöver också ligga på lagom nivå i “taxonomi”. Problemställningen är alltså ett konkret problem man vill lösa i StarCraft. Att “jämföra Q-learning med supervised learning” är inte något som direkt uppkommer från StarCraft utan handlar om att undersöka egenskaper hos tekniker. Att göra “opponent modeling” (punkt) är inte tillräckligt specifikt och konkret. Man behöver ha något som direkt bidrar till att spela bättre StarCraft.

Motivering. Du behöver också tala om tydligt *varför* man vill kunna göra/åstadkomma/... något “X”. Det *övergripande målet* är samma för alla – ungefär att maximera möjligheterna att vinna i spel mot andra agenter eller människor – så problemställningen behöver då fokusera på *hur* man tänker sig att “X” ska kunna bidra till detta. Varför är det överhuvudtaget relevant för *StarCraft-agenten* att klara “X”?

Vi rekommenderar *mycket starkt* att man baserar sina argument på en eller flera *specifika situationer* i en spelomgång, gärna med en eller flera *skärmbilder*: Nu är spelet i detta läge, vi är i läge A och motståndaren i läge B, här försöker vi göra

C och motståndaren vill göra D, i det läget är det viktigt att vi klarar av uppgift X, eftersom / för annars...”. Att börja med detta kan också göra uppgiften och dess avgränsningar och krav mycket mer konkreta för dig själv och det kommer du själv att ha mycket nytta av i resten av utredningen!

Avgränsning. Avgränsa ditt problem väl, så det syns vad du vill försöka lösa men också vad du *inte* tänker lösa. Du får gärna uttryckligen skriva att vissa problemställningar *inte* ingår! Tänk på att utredningen är på 2 hp och att du inte ska försöka lösa för många problem på en gång. Då hinner du inte gå på djupet eller hinner inte alls att göra din utredning färdig. Däremot kan du mycket väl utreda flera olika *lösningar* på ditt problem.

Omfattning. Samtidigt får problemet inte vara alldeles för enkelt. Om problemet är att ta reda på vilka *byggnadstyper* man måste konstruera innan man kan bygga något av typ X, är detta trivialt: Det kan lätt hårdkodas och lösas annars genom att titta på Tech Tree med en trivial algoritm. Du behöver få en chans att visa upp att du kan applicera AI-tekniker!

För att förklara det här bättre måste vi trots allt gå lite i förväg och diskutera lösningstekniker redan nu. Ett sätt att få ett djup i utredningen är nämligen att *algoritmerna* som krävs är lite mer komplexa. Ett annat är att du visserligen kan använda enklare algoritmer men behöver ett större djup i *modelleringen*.

Att räkna ut sannolikheter från ett Bayesianskt nätverk kan t.ex. vara enkelt, men det finns desto mer djup i modelleringsarbetet där du undersöker vilka noder och kopplingar som ska finnas i detta nätverk och där du kommer fram till bra sannolikhhetstabeller. Arbetar man med ungerska metoden kan man behöva fundera mycket på hur man definierar tasks och hur man ska definiera sin utilityfunktion. Men om man applicerar djupet-först-sökning på Tech Tree har man både en enkel algoritm och en enkel modell, vilket blir problematiskt.

2 Utvärderingskriterier

Du kommer att välja en eller flera lösningar som du ska utvärdera under HT1. Syftet med utvärderingen är att bestämma hur du ska gå vidare med implementationsarbetet under HT2 och vilken av flera lösningar som ska väljas, alternativt avgöra om en viss specifik metod är värd att implementera.

Under själva utredningsfasen ska de valda lösningarna alltid utvärderas efter följande gemensamma kriterier, med en bas i den vetenskapliga litteraturen:

1. *Hur väl verkar det gå att förstå den tänkta lösningsmetoden och att applicera den under implementationsfasen?* Här ingår att läsa på hur lösningsmetoden fungerar och vilka krav som finns för att den ska kunna användas. Det kan till exempel vara mer eller mindre svårt att förstå hur metoden ska implementeras i Python-kod, mer eller mindre svårt att förstå vilka indata som krävs och hur man ska använda resultatet av metoden (utdata) i en agent, mer eller mindre svårt att *få fram* indata givet den ofullständiga information man har tillgänglig, med mera.
2. *Hur väl verkar lösningsmetoden kunna lösa exakt det problem som ställdes upp i avsnitt 1?* Med andra ord,

om man förutsätter att man faktiskt kan förstå och implementera metoden, hur bra tror du att lösningen på problemet blir? Det kan hända att en metod du har valt visar sig lösa ett helt annat problem, och då framkommer det i den här delen av utvärderingen. Det kan hända att metoden löser problemet men ger mer eller mindre bra kvalitet på lösningen. Det kan hända att du inte kan få tag på indata med bra kvalitet och att du därför tror att metoden i praktiken inte kan lösa ditt problem i StarCraft. Och så vidare...

3. *Hur väl verkar lösningsmetoden kunna bidra till att agenten blir bättre på att spela StarCraft?* Detta är ganska nära kopplat till den förra punkten, och i vissa fall kan man tänkas hänvisa direkt till svaret på den punkten. I andra fall kan det hända att man visserligen löser problemet som det var uppställt, men att olika lösningar ändå bidrar olika mycket till spelkvaliteten.
4. *Hur “säkert” verkar det vara att välja den tänkta lösningsmetoden?* Exempelvis kan det finnas metoder där det är lätt att komma fram till en punkt där allt fungerar, varefter man kan förbättra och finputsa till tiden tar slut, medan andra metoder har en hög första tröskel att komma över, vilket är mer riskabelt. Det kan också vara mer eller mindre säkert att man lyckas få tag på de indata man behöver för att faktiskt använda metoden, speciellt om det är någon annan som ska ta fram dessa indata. Metoder kan också ha andra beroenden på sådant som är svårare att förutsäga, t.ex. maskininlärning.

Dessa kriterier ska som sagt alla användas. Anledningen till att vi tar upp dem här är att vi gärna vill att du lägger till något ytterligare kriterium som du själv tycker är intressant för ditt specifika problem och som du vill använda i din utvärdering. Fyll i så fall helt enkelt på listan ovan med fler punkter på slutet, men låt alltid de ursprungliga gemensamma kriterierna finnas kvar med den ursprungliga numreringen.

Tänk på att “ja-nej”-kriterier kan ge alltför grovkornig information och kan vara svåra att utvärdera. “Kan algoritm X implementeras”? Ja, det kan den väl.

Teoretisk utvärdering. Tänk på att kriterierna ska användas i utredningsrapporten och ska alltså gå att utvärdera i en *teoretisk* utredning med hjälp av en ren *litteraturstudie*. Sådant som du bara kan besvara genom att implementera en lösning och testa den i spelet hör inte hemma här, även om det kan vara intressant att utvärdera i slutrapporten som skrivs *efter* implementationen och projektarbetet.

Det är också därför vi skriver “verkar” i kriterierna ovan: Den rena teoretiska utredningen kommer inte att ge definitiva svar, utan är till för att du ska välja bland tänkbara metoder och hitta en som verkar rimlig att implementera för en djupare utvärdering.

3 Tekniker och algoritmer

Det är först nu, när du vet vilket problem du vill lösa och vilka kriterier du tycker att en bra lösning ska uppfylla, som du kan gå vidare och titta på vilka specifika tekniker, algoritmer, ... som *kan tänkas* vara användbara.

I detta avsnitt diskuterar du därför vilka tekniker, algoritmer, lösningsmetoder du vill undersöka och utvärdera. Dessa

tänkbara lösningar får du från att gå genom litteraturen – artiklar som vi har föreslagit och gärna andra alternativ som du hittar själv – och skumma genom dem i kanske upp till 20–30 minuter per artikel. Därifrån bildar du en uppfattning om vilka metoder du vill titta närmare på under utredningen.

Det kan fungera att göra en djupdykning i *en* teknik, men då bör du vara relativt säker på att den faktiskt fungerar för problemet. Annars kan du få problem i slutet när du bestämmer dig för att tekniken *inte* fungerar och behöver hitta en ny. För att utvärderingen ska vara tillräckligt djup behöver du då också fokusera extra mycket på att utvärdera *hur* bra lösningen uppfyller de olika kriterierna.

Det finns annars fördelar med att jämföra *flera* olika lösningar. Till exempel kan det hända att du inte behöver utvärdera vissa aspekter riktigt lika djupt för att komma fram till att lösning A är bättre än lösning B på kriterium X.

Oavsett antal lösningar måste du också läsa avsnittet “Att använda artiklar som baseras på StarCraft II” på websidan om utredningar!

Ange tydligt om du har tänkt dig att vissa tekniker är *alternativ* eller *komplement* till varandra. Fyll i en numrerad lista på följande sätt, där varje punkt är ett tänkbart *alternativ*. Om du bara utvärderar ett alternativ – är detta bra eller dåligt? – blir det alltså bara en punkt (nummer 1) i listan. Låt ändå din information vara kvar i listform så din punkt behåller sitt nummer!

1. Teknik/lösning A tillsammans med teknik/lösning B.
2. Teknik/lösning C, ensam.
3. Teknik/lösning D tillsammans med teknik/lösning B.

Här ska du inkludera *vetenskapliga referenser* till dina lösningar. Wikipedia kan fungera som komplement men är inte en vetenskaplig referens. Ett separat dokument kommer att beskriva hur man hanterar referenser med BibTeX.

Vad är en teknik / lösningsmetod? Ibland får vi istället in *problem* i denna lista, men det är just lösningarna som är fokus här. Ibland får vi istället in *representationer*, alltså något som har att göra med hur man specificerar och lagrar informationen om ett problem. Det är inte heller en lösningsmetod. I automatisk planering är t.ex. *PDDL* ett språk som används för att beskriva ett planeringsproblem, men det är inte en lösningsmetod i sig.

Tänk på indata och utdata. Främst ska du tänka på den delen under utredningen, men redan här kan du behöva fundera lite på vilka indata en lösning behöver och var den kommer ifrån, och hur resultatet av din lösningsmetod ska användas i agenten.

Risker. Tänk på att vissa typer av tekniker har sina “risker” – de kan mycket väl fungera alldeles utmärkt, men vi har sett att några (inte alla) av de som satsar på dessa tekniker behöver lägga ner mycket arbete under den senare projektdelen av kursen.

Om du väljer att *utvärdera* sådana tekniker kan det vara bra att ta med alternativa tekniker i utvärderingen. Då kan du senare, när du har läst på och vet mera, ta ett välgrundat beslut: Vilken av dina utvärderade tekniker vill du *gå vidare med* under projektet? Det är ju detta som är meningen med utvärderingen!

- Tekniker som baseras på maskininlärning är ofta intressanta. Samtidigt blir resultatet alltid beroende av hur väl inlärningen lyckas, och detta är något som man inte alltid kan påverka till 100% själv. I media ser vi bara de lyckade exemplen.

Deep Learning syns speciellt mycket i media och kräver särskilt mycket tid, inklusive datortid. Vi rekommenderar mycket starkt att man inte ger sig in i det området i den här kursen. Andra former av inlärning kan mycket väl fungera bra men det händer att det uppstår problem.

Man kan så klart bli godkänd även om utfallet (i fråga om hur bra agenten spelar) inte blir det man har önskat sig, men man kan då behöva lägga ner mer arbete på att förklara varför det hände och att visa att den inte helt enkelt är den grundläggande implementationen som är fel.

- Användandet av replays / reprisar kan ta lång tid och vi har själva inte någon direkt erfarenhet av det. Under 2020 har en sommarjobbare arbetat på att underlätta detta, men vi kan inte garantera att det går smidigt.

Tänk på att supervised learning kräver att man har en form av “facit”. Vill du lära dig i vilka situationer man bör spela offensivt respektive defensivt? Du kan säkert hitta gott om intressanta *indata* i replays – hur många enheter du har, hur många enheter motståndaren har, med mera – men det är desto svårare att automatiskt analysera om spelaren spelade offensivt eller defensivt vid en specifik punkt i en specifik replay. Då blir det upp till dig att manuellt skapa sådana “etiketter/labels” för en mindre uppsättning replays.

- Vi har främst använt en och samma karta under kursen. Det finns en karteditor, men att använda den kan vara mer knepigt.

4 Övrigt

Här kan du – om du vill – ta upp annat som du tror kan vara intressant för att andra ska förstå vad du tänker utreda.