

Rapportskrivningsguide för TDDD92 AI-projektkursen

Jonas Kvarnström Daniel de Leng

2023-08-21

Den här guiden är tänkt som stöd för rapportskrivningen (UPG2) och kompletterar kurswebbsidorna. Här hittar du mer information om hur man kan tänka angående innehållet och strukturen, men vi tittar även på vanliga misstag och hur man undviker dem.

1 Omfattning och format

Du måste skriva på svenska eftersom rapporten även ska språkgranskas. En lagom längd på rapporten kan vara 4–6 sidor text i ett typiskt konferensartikelformat med dubbla kolumner. En mall i \LaTeX -format ('Rapport' i materialkatalogen) kommer att finnas tillgänglig och ska användas av samtliga deltagare. Den innehåller de tänkta sektionrubrikerna, som liknar den standardiserade indelningen av vetenskapliga rapporter men delvis är anpassad till utredningens specifika behov. (Rapportmallen är baserad på den för *IJCAI-19 Proceedings*, en av de högst rankade AI-konferenserna. Vi har gjort några mindre förändringar, ändrat rapportstrukturen, satt pappersstorleken till A4 och lagt till stöd för svenska.) Vissa rekommenderar Overleaf för att arbeta med \LaTeX på webben. Bilder är mycket välkomna. Dock ersätter de inte det skrivna ordet, så en rapport med många stora bilder behöver fortfarande normalt omkring 4–6 sidor text. Även referenslistan, eventuell innehållsförteckning med mera ligger utanför de 4–6 sidorna. Skulle du känna att du har så mycket att säga att du vill ha någon sida till kan det vara OK, speciellt om det gör att hela rapporten blir mer lättläst eller att du får bättre underlag för de slutsatser du drar. Men då behöver du först se till att du skriver på ett "lagom kortfattat och koncist" sätt, så att rapporten inte blir längre på grund av överflödigt material. Försök inte att anpassa formatet eller texten för att få en alltför kort rapport att se större ut eller tvärtom. Det minskar bara läsbarheten.

2 Innehåll och övergripande struktur

Det finns ett förväntat flöde i en utredning, och man behöver tänka på att följa detta flöde i rapporten och att tydligt separera olika aspekter av utredningen!

Detta leder till olika avsnitt i utredningsrapporten. Avsnitten finns redan med i mallen (hittas på kurshemsidan), så titta gärna på den parallellt med att du läser här.

2.1 Problemställning

I de flesta fall kan man använda problemställningen från utredningsplanen som grund för detta avsnitt – men man behöver dels ta hänsyn till de kommentarer man kan ha fått i granskningen av planen, dels arbeta vidare med en tydligare och mer formell beskrivning av problemställningen. Var noggrann och tydlig. Definiera dina begrepp. ”Jag vill veta vart jag ska gå” – när, i vilken situation, ...? Vad menar du med ”arbetare”, ”jobb”, ”plats”, och så vidare (om du använder dessa begrepp)? Vad är ”kostnad” i din problemställning? Tänk dig att någon annan ska lösa problemet utifrån din beskrivning – problemställningen är då en kravspecifikation som måste vara extremt tydlig. Här ser vi ofta problem som gör det svårt att utvärdera din utvärdering (!), och det leder ofta till komplettering. Samtidigt får du gärna koppla det specifika problemet till en generell (men väldefinierad) problemtyp – ”Detta kan ses som ett planeringsproblem”, ”detta är en form av *path finding*”, ”detta är ...”. Var också mycket noggrann och tydlig med exakt vilken information du tänker dig ska finnas tillgänglig (”indata”, allt det som lösningsmetoden får använda sig av för att lösa problemet) och hur den definieras, samt exakt vad du tänker dig att en lösning skulle åstadkomma (form på utdata alternativt vilka ”styrsignaler” lösningen ger). Motivera också din problemställning: Tala om tydligt varför man vill kunna lösa det här problemet. Det övergripande målet är samma för alla – ungefär att maximera möjligheterna att vinna i spel mot andra agenter eller människor – så det behöver inte diskuteras här, utan problemställningen ska fokusera på hur man tänker sig att det 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, helst 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!

2.2 Utvärderingskriterier

Målet med utredningen är att hitta en eller flera lösningar som uppfyller vissa kriterier. Dessa kriterier måste man då komma fram till i förväg, innan man letar efter lösningar; annars vet man inte vad man ska utreda, och det kan hända att man börjar utvärdera utefter vad som är enkelt istället för vad som egentligen är viktigt. Kriterierna har inget att göra med vilka lösningstekniker man har hittat, så fortfarande hålls diskussionen fokuserad på problemet. Man ska t.ex. inte nämna något om inläring, sökning, potentialfält, eller liknande.

Eftersom det inte är så lätt att välja rimliga kriterier, och lätt att missa något viktigt, ger vi redan på förhand 4 kriterier som *alla* ska använda. Man får sedan gärna lägga till flera kriterier på egen hand.

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? 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? Det vi är ute efter här är en bedömning av sannolikheten att det leder till ett bra resultat för kursen, med en rimlig arbetsinsats: Även en väldigt bra lösning kan vara olämplig om den inte passar in i kursarbetet. Förståelsen (punkt 1) är en del av detta, men det finns också andra aspekter som är viktiga. Exempelvis kan det finnas metoder där det ser ut att vara lätt att komma fram till en punkt där allt fungerar, varefter man kan förbättra och finputsa till tiden tar slut, vilket gör att man kan anpassa sig till oväntade bakslag eller framsteg. Andra metoder kan se ut att ha 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 i gruppen som ska ta fram dessa indata. Metoder kan också ha andra beroenden på sådant som är svårare att förutsäga, t.ex. maskininlärning.

Kriterierna är numrerade och ska vara det även i rapporten, så du på ett tydligt sätt kan hänvisa till dem i utvärderingen.

I utredningsrapporten ska lösningar utvärderas efter de här kriterierna utan att man har provat att implementera lösningen eller applicera den på StarCraft. Det är främst därför vi skriver ”verkar” i kriterierna ovan: Den rena teoretiska utredningen kommer inte att ge definitiva svar om hur bra lösningen är, utan är till för att du ska välja bland tänkbara metoder och hitta en som verkar rimlig att implementera för en djupare utvärdering.

2.3 Tekniker och algoritmer

När man har satt upp de kriterierna går man vidare och letar efter tekniker som kanske uppfyller dem. I detta avsnitt av utvärderingsrapporten ska du:

- Beskriva samtliga dessa tekniker på en sådan nivå att läsaren kan förstå dem från beskrivningen, utan att gå till en källartikel för att förstå dem. En sådan beskrivning kan givetvis inte vara precis lika detaljerad som källartikeln. Istället ska man hitta en lämplig detaljnivå som ger en tillräcklig bas för läsaren att (a) förstå hur tekniken kan appliceras på det givna problemet, (b) förstå den kommande utvärderingen och varför du som skriver utvärderingen har dragit dina slutsatser.
- I normalfallet beskriva algoritmer i pseudokod! Beskriver man dem helt och hållet på svenska blir det luddigt och svårförstått; detta passar mycket bättre som komplement till pseudokod, för att förklara vad den egentligen gör (på en hög nivå), varför den gör det, och hur det hela hänger ihop. Om en algoritm är beskriven i pseudokod i en artikel är det fullt tillåtet (och rekommenderat) att citera denna pseudokod exakt som den såg ut och helt enkelt lägga till en citering och en egen förklaring av pseudokoden. (Pseudokod är bästa lösningen när man går genom en algoritm steg för steg, men om man inte orkar, kan nästa alternativ vara punktlistor som kan vara nästlade i upp till två nivåer. Nej, det blir inte samma ”flyt” som i en löpande text... men detta är inte heller en roman, och det är väldigt viktigt att läsaren får en översikt. Det är mycket lättare att hålla reda på var man är i en punktlista än i ett stycke på 40 rader.)

Tänk på: Vi har fortfarande inte kommit till utvärderingssteget! Blanda alltså inte in några egna värderingar av teknikerna ännu, utan håll dig helt och hållet till en objektiv beskrivning av valda tekniker utan att explicit koppla det till StarCraft II.

Exempel: Anta att ett av dina kriterier är att en teknik måste fungera bra i en värld där kartan dynamiskt ändras under tidens gång. När du i detta avsnitt beskriver en teknik kan du tala om att den appliceras på en statisk karta, eller att den har vissa specifika dynamiska egenskaper. Detta är en objektiv egenskap som inte har med StarCraft II att göra. I nästa avsnitt blir det dags att diskutera om och hur tekniken kan appliceras i StarCraft II (”teknik A appliceras på en statisk karta och om vi då ska använda denna i StarCraft II måste vi göra så här”) och konsekvenserna av detta. I utvärderingsavsnittet blir det sedan dags att diskutera konsekvensernas koppling till kriterierna.

Kom också ihåg att t.ex. supervised learning och reinforced learning snarare är metatekniker än tekniker – de är generella idéer som sedan kaninstansieras till specifika tekniker och algoritmer såsom Q-learning. Det räcker alltså inte att ange ”supervised learning” utan man behöver ange en faktisk konkret teknik.

2.4 Användning i StarCraft II

För att använda de tekniker som just beskrevs behöver man applicera dem på just StarCraft II. Det räcker ju inte att säga ”jag har beskrivit A* och nu kör jag A*”. Alla implementerade tekniker behöver någon form av input, och de ger någon form av output, antingen data/information eller ”styr signaler” som kontrollerar spelet.

I de allra flesta rapporter behöver man därför beskriva hur en vald teknik skulle kunna tänkas appliceras. Använder man A* för att hitta vägar behöver man t.ex. ange hur man i så fall skulle generera en graf att söka i, vad som är start- och målnoder, och hur man kommer fram till detta utifrån det API som man använder i StarCraft II. Man behöver också tala om vad det är som ska röra sig enligt den väg som genereras. Använder man potentialfält kan läsaren fråga sig var potentialen kommer ifrån och exakt vad det är som sedan styrs av den uträknade potentialen i varje punkt... och så vidare.

Tänk på att det kan vara viktigt att veta vilken information som faktiskt finns tillgänglig från StarCraft. Vad kan en agent veta, och vad kan den inte veta? Om en algoritm kräver en viss sorts information, kan du få tag på den, och i så fall hur? Är informationen pålitlig?

I utredningsrapporten behöver detta avsnitt så klart inte gå ner på den allra mest detaljerade nivån (”ända till kod”), men man behöver definitivt få en god bild av vad du som utredare har tänkt dig. Annars kan tekniken och slutsatserna ”hänga löst”.

Om du försöker applicera en vald teknik men kommer fram till att den inte alls är applicerbar (inte bara att den bryter mot dina utvärderingskriterier utan att den överhuvudtaget inte gör det du vill, eller kräver information du inte har):

- Upptäcks detta ”tillräckligt” tidigt bör du välja att ta bort tekniken helt och införa en annan teknik som alternativ.
- Upptäcks det när du har arbetat en hel del på beskrivningen och att försöka applicera tekniken, kan det vara orimligt att börja om med en annan teknik. Då kan detta utredningsarbete finnas kvar i rapporten. Du måste då ändå ha tillräckligt mycket material i rapporten i sin helhet. Kanske du även har en annan teknik att utreda. Diskutera gärna med handledarna.

2.5 Utvärdering

Man utvärderar de valda teknikerna, så som de är tänkta att appliceras på StarCraft II, efter de kriterier som man redan tidigare satte upp. Utvärderingen

ska alltså ange hur väl varje vald teknik uppfyller varje angivet kriterium enligt det sätt man har tänkt att applicera tekniken. Övriga slutsatser får vänta till senare. Gå genom alla de numrerade kriterierna du satte upp, och se till att samtliga kriterier faktiskt utvärderas i tur och ordning, för samtliga tekniker / lösningar som ingår i din studie. Du får lägga till nya kriterier, om du upptäcker något som missades i planeringen och som verkar vara viktigt att utvärdera efter. Du får däremot inte ignorera något av de kriterier som du ställde upp i din utvärderingsplan. Om de visar sig svåra att utvärdera, ska du istället motivera detta.

I de flesta fall är det givetvis omöjligt att vara helt säker på hur bra en viss teknik fungerar i StarCraft II utan att faktiskt implementera den och testa (ett undantag kan vara när man upptäcker att tekniken faktiskt inte alls var applicerbar för att den löser ett helt annat problem). Arbeta då för att komma fram till så bra slutsatser du kan, och sök gärna efter mer underlag om detta är möjligt – ge inte direkt upp och säg ”vet inte” – men gör inte utvärderingen ”för stark” bara för att du vill kunna säga något definitivt!

Skriv till exempel inte ” X är definitivt bra på kriterium Y ” om du bara har underlag för ” X kan nog vara ganska bra på kriterium Y , men det finns en del osäkerheter av dessa anledningar”. Om det inte finns mer evidens än så, är det ju detta som är den korrekta utvärderingen.

Var också försiktig med att jämföra två tekniker utifrån vad olika författare har skrivit om dem, eller utifrån deras resultat i olika världar och situationer. Om en författare beskrev teknik A som ganska effektiv, och en annan beskrev teknik B som lite långsam, är då A bättre än B eller hade författarna olika mätstickor? Om teknik A fungerade bra (i schack) och teknik B var dålig (i Fia med knuff), måste teknik A då vara bättre i StarCraft II?

Att komma fram till uttalanden av rimlig styrka, varken alldeles för starka eller alldeles för svaga, är viktigt i rapporten. Försök hitta en lagom nivå, och argumentera sedan för dina uttalanden så att läsaren övertygas om att det du faktiskt säger verkar korrekt. Om du säger att en algoritm definitivt är bra, behöver du övertyga läsaren om detta. Om du säger att en algoritm verkar bra men att det inte är helt säkert, behöver du övertyga läsaren om att det finns goda egenskaper (”verkar bra”) men också motivera varför artiklarna inte kunde ge tillräckligt underlag för att vara säkra (”har bara utvärderats i situation X och den situationen är liknande min situation, men eftersom situationerna skiljer sig på punkt Y och Z är det möjligt att det inte fungerar lika bra här”). Man kan alltså inte komma undan med att helt enkelt säga ”kanske bra, vet inte riktigt” på alla kriterier; både säkerhet och osäkerhet behöver motiveras på något sätt!

Tänk på att inte komma med ny information om hur teknikerna och algoritmerna fungerar. Detta hör hemma i den tidigare beskrivningen av teknikerna och algoritmerna. Lägg inte heller till information om hur du har tänkt applicera dem här, utan uppdatera i så fall de tidigare avsnitten. Varje avsnitt har sitt eget ämne, och här handlar det om just utvärdering.

2.6 Slutsats

Till slut drar man slutsatser. Medan utvärderingen i det förra avsnittet redan har fokuserat på hur bra olika lösningstekniker verkar kunna uppfylla de uppsatta kriterierna, ska slutsatserna handla om hur utvärderingen påverkar det du tänker göra i det fortsatta arbetet i projektet. Exempel: Väljer du att implementera någon av de utvärderade teknikerna, och vilka aspekter av utvärderingen är det i så fall som gör att du väljer just denna teknik?

Denna ordning och detta flöde ska följas i rapporten. Sedan kan det i praktiken vara så att ni egentligen har varit intresserade av en viss teknik eller algoritm och först därefter letat problem som den kan lösa, men vid rapportskrivandet får man se det som ett förstadium till den verkliga studien. Rapporten ska alltså starta med problem och kriterier och sedan gå vidare till teknikerna, och så vidare.

3 Om bakgrundsmaterial

- Utgå från att läsaren redan känner till *StarCraft II* i allmänhet och kanske har spelat spelet i någon timme. Du behöver alltså inte beskriva de mer grundläggande tankarna bakom spelet eller tala om vad en *Terran* är, men räkna inte med att läsaren känner till en *Ultralisk* eller *Unit Dancing*.
- Utgå från att läsaren redan vet i vilken kontext du skriver rapporten: Du är medlem i en projektgrupp som bygger en agent som ska spela *StarCraft II*. Detta behöver inte beskrivas på nytt i rapporten. Om det ändå beskrivs, räknas det inte med i rapportens omfattning.
- Utgå från att läsaren har gått ett par år på U-programmet (och alltså är programmeringskunnig och allmänt ”tekniskt kunnig”), och att läsaren redan känner till AI-begrepp motsvarande kursen TDDC17. Det finns alltså en hel del grundläggande AI-kunskap som normalt inte behöver repeteras i rapporten, inklusive grundläggande begrepp som agenter, sökning, planering, vägplanering med mera. Undantaget är att de problem och tekniker som du utvärderar i rapporten måste beskrivas tydligt i teknikavsnittet även om teknikerna ingår i TDDC17. Exempel: Om A^* är en av teknikerna du utvärderar för användning i projektet behöver du alltså beskriva algoritmen, men om du bara vill nämna den i en kort jämförelse (”min utvärderade teknik X är bättre än A^* , eftersom ...”) kan du hänvisa till att A^* ingår i TDDC17. Däremot skulle du behöva beskriva *Focused D** till en högre grad även om den bara ingick i en kort jämförelse, eftersom den inte diskuteras i TDDC17.

4 Referenser

Utredningen ska baseras på vetenskapliga artiklar. Alltså behöver man hitta sådana artiklar och även referera till dem på lämpliga ställen i texten. Vi skriver

”artiklar” i plural, eftersom en utredning av den här typen normalt inte kan baseras bara på en enda artikel. Ett normalt antal kan vara 3–5 olika referenser (varav någon kan vara en bok), i vilka man t.ex. kan hitta olika information om samma lösningstekniker, eller mer information om flera olika tekniker.

- Läs på om att referera, citera och parafasera i universitetets *NoPlagiat*-självstudie, som biblioteket numera har flyttat till Lisam under ”Test”. Även om du inte genomgår hela *NoPlagiat*-studierna behöver du veta hur man ska använda citeringar. Det räcker t.ex. inte bara att en artikel refereras till någonstans i texten, utan man behöver veta vad som kommer från just den artikeln, genom hela rapporten!
- Var noga med att skilja på vad som är ditt material (dina egna åsikter och slutsatser) och vad som är taget mer eller mindre direkt från artiklarna. I vetenskapliga artiklar som bygger på andras material är det ofta tätt mellan referenserna. Med andra ord, om du plockar information från samma artikel på 20 olika ställen är det rätt att referera till artikeln på alla dessa 20 ställen, så det blir tydligt var informationen kommer från!
- När det gäller att hitta rätt artiklar: Det viktigaste är inte att man hittar den allra bästa, nyaste artikeln att referera till. Kursen är inte tillräckligt stor för att vi ska kunna lägga ner ett enormt arbete på det (skulle lätt kunna ta en termin att göra en rejäl undersökning). Istället är det viktigaste att de man kan utvärdera på ett bra sätt efter de relevanta referenser och studier man faktiskt har tillgång till.
- Om du märker att information som du är intresserad av saknas i litteraturen, eller att underlaget inte räcker till, får du gärna (kortfattat) nämna vad du inte har kunnat ta reda på. Det visar att du inte bara missade viktiga aspekter, utan att informationen saknades.
- Egna undersökningar kan finnas med till viss del, men tyngdpunkten ska vara på att undersöka vilka slutsatser man kan dra från publicerade artiklar och böcker.

5 Svengelska

När man skriver en rapport av detta slag är det lätt hänt att man kommer får med en del svengelska. Här samlar vi en del förslag och exempel runt detta ämne. Om du undrar över översättningen av något ord som inte finns med här: Sök gärna på nätet och fråga oss annars. Om ett ord inte kommer genom din stavningskontroll (för en sådan gör du väl?) kan det bero på att det är svengelska.

- Bayesian networks - Bayesianska nätverk (uppkallat efter Thomas Bayes)
- Hidden Markov Model - dold markovmodell

- Hungarian method - ungerska metoden
- Q-learning - Q-inlärning eller Q-learning i kursiv stil
- action - handling (inte aktion!)
- back propagation - bakåtpropagering
- barracks - kasern (en barack är något annat)
- branch-and-bound – översätts inte, skrivs i kursiv stil. "Branch" är att förgrena ut sökningen i ett sökträd, "bound" är att begränsa längden på grenarna då man ser att man inte kommer att kunna hitta bättre lösningar i den gren man arbetar med.
- breadth first - bredden först (bredden-först-sökning)
- build order - byggordning
- choke point - finns ingen perfekt översättning; kan skriva flaskhals ("choke point") första gången det används, och sedan använda flaskhals, så har man kopplat det tydligt till ursprungsbegreppet och det blir varken missförstånd eller svengelska.
- concept - begrepp (oftast – "koncept" kan ha en något annorlunda betydelse!)
- constant - konstant
- depth first - djupet först (djupet-först-sökning)
- determine - t.ex. bestämma (inte determinera)
- deterministic - deterministisk
- discretization - diskretisering
- edge - båge
- entries - poster
- flow fields - flödesfält
- fuzzy set - suddig mängd, oskarp mängd (motsatsen kan kallas "skarp mängd")
- gradient descent - gradientsökning
- graph - graf
- greedy - girig
- grid - rutnät

- grid cell, cell - ruta
- harvest - skörd, skörda
- hidden layer - gömt/gömda lager
- layer - lager
- make a decision - ta (ett) beslut (inte "göra")
- map editor - karteditor
- marine - marinsoldat
- node - nod
- ordered pair - ordnat par (som har en ordning)
- potential fields - potentialfält (inte "potentiella"!)
- potential flow - potentialflöde
- range - räckvidd (om det handlar om att nå fienden med vapen)
- ratio - kvot
- reinforcement learning - förstärkningsinlärning (inte "förstärkt inlärning")
- reward - belöning
- scouting - utforskning, patrullering, rekognosering
- simplify - förenkla (inte simplificera!)
- space - rymd (state space = tillståndsrymd)
- state - tillstånd
- state variable - tillståndsvariabel
- supervised learning - övervakad inlärning
- supply depot - depå, förråd, lager
- temporal - temporal
- temporary - temporär
- time step - tidssteg
- unit - enhet
- unit collision - att man kolliderar med en annan enhet
- utility function - nyttofunktion

- variable elimination - variabeleliminering
- weighted - viktad
- win/lose - vinna/förlora

Samtidigt finns också en del terminologi där det inte finns någon vedertagen svensk översättning. Där kan det vara minst förvirrande att fortsätta använda de engelska orden. Det kan rekommenderas att sätta dem i kursiv stil, *reaper*, eller inom citattecken, "reaper", för att markera att man använder specifika termer snarare än att man bara har missat att översätta något.

- Tekniker och algoritmer med "egennamn", som Q-learning, reinforcement learning, supervised learning, ...
- Reaper, marauder, ... – dessa kan vi också se som egennamn som man inte nödvändigtvis förstår om man översätter dem "direkt". Ett undantag är marine som bör översättas till marinsoldat.
- Kiting
- ...

6 Tänkbara problem

Här är en kortfattad lista på några fler saker att vara försiktig med.

- Definiera problemet tydligt. "Byggnadsplacering" är i sig ingen tydlig beskrivning. Är problemet (1) jag tänker placera ut någon sorts byggnad och vill veta var den ska vara, (2) jag tänker placera ut en byggnad av typ X och vill veta var den ska vara, (3) jag tänker placera ut n byggnader och vill veta var de ska vara, ...? Tänk dig gärna att du definierar problemet för någon annan som utifrån din beskrivning ska förstå vilket problem som ska lösas. Otydligheter gör att den personer kan spendera veckor på att lösa fel problem!
- Kriterier och deras utvärdering
 - Tydlighet: Man behöver förstå exakt vad kriterierna betyder. Om du diskuterar tidskrav, diskutera då tydligt vad du menar. Ska det köras på 1/60 sekund? Ska det kunna uppdateras på 1 sekund inför att man kör igång spelet? Vad är "snabbt" egentligen? Skilj också t.ex. på att maximera sannolikheten att få information och att maximera mängden information man förväntas få. Tänk dig gärna att du definierar kriterierna för någon annan som ska utvärdera teknikerna. Otydligheter om vad som egentligen ska utvärderas kan göra att fel teknik kommer att väljas för projektet!
 - "Klärar tekniken att lösa problemet"? – är detta rätt fråga? Ofta är det "Hur bra klarar tekniken att lösa problemet?".

- Dra inte **FÖR** starka slutsatser. Man kan komma till slutsatsen att ”jag vill testa med Q-Learning” utan att för den skull behöva dra slutsatsen att ”Q-learning är definitivt bäst”, om det inte går att förankra den senare slutsatsen i underlaget.
- Om du ser i litteraturen att teknik A är bra på något, men saknar uppgifter om teknik B , har du ingen stark grund för att säga att A är bättre än B på detta.
- Terminologi och tekniska begrepp
 - Optimal = perfekt. Ibland vill man ha ”bra” och kräver inte ”optimal”.
 - Tidskomplexitet = hur tidsåtgången varierar med storleken på problemet. Sortering kan göras med tidskomplexitet $n \log n$, där n är antalet element. Ibland vill man undersöka ungefärlig uppskattad tidsåtgång snarare än formell tidskomplexitet.
 - Var noga med skillnad mellan olika tekniska begrepp som tillstånd, stokastiska variabler vars värden är tillstånd, med mera. Och om du pratar om tillstånd, vilken sorts information ingår i ett tillstånd?
 - Använd genomgående samma ord för att beskriva samma sak. Kalla inte något ett ”område” ibland och en ”zon” ibland; i tekniskt skrivande är vi inte ute efter variation, utan vill ha maximal tydlighet.
- Skrivfel och ’oversittings’
 - eftersom att – ”eftersom” eller ”för att” eller ”på grund av att”.
 - ”Human” = mänsklig i betydelsen god, barmhärtig, ”inte grym”.
 - ”Flertalet” = ”de flesta” – ”ett flertal” = ”många”