
ARTIFICIELL INTELLIGENS ELLER AVANCERAD MJUKVARUUTVECKLING

Text: Jan Eric Larsson

Redan på 1830-talet utvecklade den brittiske matematikern **Charles Babbage** idéer och skisser på maskiner som skulle kunna *programmeras* att utföra olika beräkningar. Den elektroniska datorn utvecklades under andra världskriget. Sedan dess har utvecklingen gått i en rasande fart och idag är informationsteknologin en integrerad del av mänsklighetens vardag. Redan Babbage och han medarbetare **Ada Lovelace** insåg att programmering innebär att en dator i princip kan lösa väldigt svåra problem. Det gäller bara att veta hur det ska gå till och att implementera rätt algoritm. När **Alan Turing** skrev sin artikel "Computing Machinery and Intelligence" hade man redan insett att det vore teoretiskt möjligt för en dator att lösa problem som hittills varit förbehållet människor.

År 1956 förslog **John McCarthy** en konferens på Dartmouth College om det ämnesområde han kallade *artificiell intelligens* (AI). Redan vid denna tidpunkt talade man om datorer, naturligt språk, neurala nätverk, kreativitet och schackspelade program. Under de första åren hade detta forskningsområde stora framsteg, bland annat inom spelprogrammering och översättning av naturligt språk. Men i takt med att man angrep svårare problem, avstannade succén under sjuttioalet.

Nästa stora våg inom AI var regelbaserade expertsystem, som blev populära under början av åttiotalet, tätt följda av framsteg inom neurala nätverk och sedan agentbaserade system. För närvarande är det naturligt språk, "big data" och webbsökning som är stora områden inom AI.

Det går inte att komma ifrån att namnet artificiell intelligens är valt delvis för att göra att området ska låta mer spännande; mer som science fiction. Det är också denna sida som spridits sig till populärkulturen, i filmer som *2001: A Space Odyssey*, *Artificial Intelligence*, och *Her*. Under de senaste åren har en diskussion växt fram om *den teknologiska singulariteten*, alltså frågan om AI kan utvecklas så mycket att den tar över världen.

Verkligheten inom forskningsområdet AI är inte lika dramatisk. AI har alltid varit ett av de områden som legat i framkanten vad avser nya idéer, men efterhand har tekniker och metoder mognat och blivit till egna discipliner. Numera räk-

nas heuristisk sökning, regelbaserade expertsystem och neurala nätverk som etablerade metoder inom datavetenskap.

När olika AI-tekniker nått toppen av sin popularitet och intresset därefter gått ned, har det ofta hävdats att AI har misslyckats. Till exempel var det många som hävdade att AI "hade misslyckats" när intresset för expertsystem falnade i slutet av åttiotalet. Detta är inte korrekt. Numera är expertsystem en mogen teknologi som används bland annat i de program som automatiskt handlar med aktier på världens börser.

Man kan kanske vänta sig att AI ska återkomma till mediescenen när nya områden mognar. Jag gissar att detta särskilt kommer att gälla robotik. Man ser redan nu fler artiklar om nya robotar, där särskilt Japan är ledande.

Men under tiden arbetas det med AI i det tysta. Resten av denna artikel kommer att handla om hur en "intelligent" algoritm används för att hantera problem i kontrollrum för komplexa tekniska processer.

Ju mer etablerad teknologin blir, desto mindre kommer den att kallas artificiell intelligens. Snarare kommer man att kalla den avancerad mjukvara i stället.

En stor mängd tekniska system och processer styrs i dag med datoriserade styrsystem som finns i kontrollrum. Det kan röra sig om konventionella kraftverk, kärnkraftverk, petrokemiska industrier, kraftnät, gruvindustri och pappersbruk, men även om fordon och trafik, till exempel cockpit i ett flygplan eller kommandobryggan på ett fartyg.

I samtliga dessa fall är systemet uppbyggt så att det finns sensorer ute i processen, vars signaler tas emot av ett datorbaserat styrsystem, som lagrar processdata och presenterar dem för den personal som arbetar i kontrollrummet, så kallade operatörer. Styrsystemet kan också utföra automatisk reglering och styrning, genom att beräkna styrsignaler och skicka dessa till olika ställdon som påverkar processen. Enkla exempel på detta är en termostat, farthållaren i en modern bil, eller ett flygplans autopilot.

Operatörernas roll i kontrollrummet är att övervaka processen genom att övervaka de data som styrsystemet presenterar, för att kunna vidta åtgärder om något går fel. För att understödja denna process innehåller styrsystemet ett antal larm, vilket innebär att systemet jämför mätvärden från processen med förinställda gränser, och om en gräns passeras, genererar systemet en varning till operatörerna. Det kan vara ett meddelande i textformat i en lista men även en markering på en grafisk skärm. Ofta åtföljs varningen av ett ljud, vilket vanligen är den lekmannamässiga associationen till ett larm (en varningstuta).

Vad som är mindre känt är att det är *svårt* att använda larmen som information, särskilt när det blir komplexa felsituationer och det kommer många larm i snabb takt. Problemet är att varje larm utgör information om en liten del av problemet, men operatörerna måste själva sammanställa informationen från ett stort antal larm till en mental bild av vad som egentligen har gått fel.

Tyvärr är det ofta så att ett fel leder till många larm. Detta beror på att ett fel ofta orsakar följdfelet. Det blir alltså en kedja av konsekvenser i processen, och detta leder ofta till en skur av larm. I stället för att det kommer ett larm från det verkliga felet, kan det komma tiotals, hundratals eller till och med tusentals larm i kontrollrummet. Detta kallas larmkaskader och gör det svårt för operatörerna att förstå vad som egentligen har hänt.

Den 23 september 2003 drabbades södra Sverige av ett stort strömavbrott. I kontrollrummet för det Svenska kraftnätet i Stockholm kom det över femhundra larm inom ett par minuter. Detta gjorde det omöjligt för operatörerna att begripa vad som hänt. Det var två fel inom fem minuter som orsakade hela avbrottet; ett stopp i reaktorn Oskarshamn 3 på östkusten, och en kortslutning i en station nära Horred på västkusten. Det tog närmare fyra timmar för personalen att räkna ut att det fanns ett fel just i stationen Horred.

Den 28 mars 1979 uppstod ett fel vid ett snabbstopp vid kärnkraftverket i Three-Mile Island, Pennsylvania. Det var en ventil som kärvade och inte stängdes när den skulle, men sensorn visade felaktigt att ventilen skulle ha stängts. Detta ledde till en kaskad på över 100 000 felindikeringar, vilket i sin tur gjorde att operatörerna inte kunde förstå vad som pågick. Hade det inte kommit en sådan larmskur, hade Harrisburgolyckan aldrig inträffat.

Den 27 december 1991 fick ett SAS-plan problem med motorerna cirka 30 sekunder efter starten från Arlanda. Efter cirka tre minuter flygning kraschade planet på ett fält vid Gottröra. Under den korta flygtiden kom det över 1 000 larm på den lilla larmpanel som fanns i cockpit. Piloterna hade naturligtvis ingen nytta av detta, och exemplet visar att larmsystemet blev värdelöst i en komplex felsituation.

Under min forskningskarriär har jag ägnat mig åt att utveckla en algoritm för att avläsa larm och utföra en analys som talar om vad som är ursprunget i en larmkaskad, alltså den egentliga orsaken eller rotfelet.

Grundidén är ett grafiskt språk som beskriver flöden av massa, energi och information, i väldigt enkla termer, nämligen källor, transporter, lager, greningspunkter och sänkor. Detta språk kallas för *multilevel flow models* (MFM) och uppfanns av professor **Morten Lind** vid Danmark tekniska universitet i Lyngby i slutet på sjuttioalet. Med detta språk är det lätt och effektivt att beskriva (eller modellera) även komplexa tekniska processer.

Mitt eget bidrag är att jag utvecklade en algoritm som baserar sig på en MFM-modell och tar emot larmvärden i form av lågfel och högfel, och som räknar ut vilka fel som

oberoende, respektive vilka fel som är en konsekvens av något annat fel.

Låt oss ta ett enkelt exempel. Antag att ett flöde transporteras in i ett lager. Det kan till exempel vara en pump som pumpar in gas i en tank. I MFM beskrivs detta som en transportfunktion som är kopplad till en lagerfunktion. Om algoritmen ser att flödet genom pumpen är för lågt och nivån i tanken också är för låg, drar den slutsatsen att det låga inflödet är orsaken, och den låga volymen är ett konsekvensfel. Om i stället inflödet är för lågt men nivå för hög, dras slutsatsen att den höga volymen är orsaken och inflödet en konsekvens. Tanken är överfull och blockerar inflödet.

En verklig process, till exempel ett kraftverk eller ett kraftnät, innehåller inte bara en transport och ett lager, utan kan bestå av tusentals källor, transporter, lager och sänkor. Poängen är att det inte är annorlunda, utan bara fler av samma sorts objekt och kopplingar. När vi väl funderat ut hur två kopplade objekt påverkar varandra, kan vi använda samma resonemang om och om igen.

Det visar sig att om man designar algoritmen rätt, så kan man analysera alla tänkbara felsituationer korrekt, oavsett hur många oberoende rotfel det finns och oavsett hur stort systemet är. Det är helt enkelt en komplett lösning på problemet rotfelsanalys.

Denna algoritm är nu huvudprodukten för företaget GoalArt, som är en avknoppning av min tidigare forskning vid Lunds tekniska högskola. Vi har provat algoritmen med data från strömavbrottet i Sverige 2003, och den identifierar de två oberoende egentliga felen, och bara dessa, redan under den första minuten efter att felet i Horred kommit in.

En stor praktisk fördel med denna algoritm är att man för vissa typer av processer kan skapa modellen automatiskt. Detta gäller bland annat för kraftnät, banljusbelysning för flygplatser, samt för bagagehantering. I dessa fall blir det väldigt enkelt att bygga och installera ett GoalArt-system. I andra fall måste man tillverka modellerna för hand. Detta gäller till exempel för kraftverk och pappersbruk.

Vårt system finns för närvarande installerat hos Svenska Kraftnät (som driver det svenska kraftnätet) och HOPS (det kroatiska kraftnätet). Det har också testats för kraftverk, pappersbruk, flygplatser, bagagehantering och inom oljeindustrin, det senare tillsammans med Chevron i Kalifornien.

När vårt system är installerat i ett kontrollrum och det uppstår en larmkaskad, beräknar systemet omedelbart den egentliga orsaken och visar denna i en separat larmlista. Operatörerna kan kasta en blick på denna lista och får då en genomtänkt förklaring till vad som händer eller har hänt. Datorn är som en automatisk hjälpre, som tänker ut den rätta förklaringen och hjälper personalen att förstå situation och därmed också gör att de kan vidta åtgärder snabbt och korrekt.

Förhoppningsvis kommer denna form av artificiell intelligens att slå igenom inom processindustrin, och därmed ta bort en stor källa till misstag, incidenter och olyckor.

Ju mer etablerad teknologin blir, desto mindre kommer den att kallas artificiell intelligens. Snarare kommer man att kalla den avancerad mjukvara i stället.



Jan Eric Larsson är professor i informationsteknologi vid Lunds Tekniska Högskola, samt entreprenören bakom företaget GoalArt.