Databaser fra et logikkperspektiv

Evgenij Thorstensen

IFI, UiO

Høst 2013

Outline

Innledning

Logikk kan brukes til å bevise ting — og så da?

Vi skal se på hvordan søk og optimisering kan reformuleres som bevis-problemer.

Abstrakte logiske *objekter* kan representere mer konkrete objekter.

For eksempel databaser og spørringer.

Matematisk modellering

Vi skal frem til algoritmer for å jobbe med databaser.

Hvor kommer algoritmer fra? (fritt etter Jostein Gaarder)

Matematisk modellering

Vi skal frem til algoritmer for å jobbe med databaser.

Hvor kommer algoritmer fra? (fritt etter Jostein Gaarder)
Mitt svar: De kommer (ofte) fra teoremer.

Matematisk *modell* $\rightarrow$ teorem om *korrekthet* $\rightarrow$ *resonnering* og nye teoremer $\rightarrow$ *anvendelse*.

Representasjon

Vi skal jobbe mye med representasjon, oversette mellom objekter.

For eksempel skal vi oversette

- en konkret database D til
- en førsteordens modell $\mathrm{D}^{\mathcal{M}}$ til
- en førsteordens formel $\mathrm{f}(\mathrm{D}^{\mathcal{M}})$.

Vi skal lære oss å sjonglere slike transformasjoner, og å bevise deres egenskaper.

Relasjonsdatabaser

Litt uformelt er en relasjonsdatabase en samling tabeller.

Brukernavn	Navn	FDato	Stillingskode
evgenit	Evgenij Thorstensen	05.07.1987	SKO1352

Kode	Tittel	Ansvarlig	Semester
INF3170	Logikk	arild	H
INF1050	Systemutvikling	dagsj	V

Relasjonsdatabaser

Litt uformelt er en relasjonsdatabase en samling tabeller.

Brukernavn	Navn	FDato	Stillingskode
evgenit	Evgenij Thorstensen	05.07.1987	SKO1352

Kode	Tittel	Ansvarlig	Semester
INF3170	Logikk	arild	H
INF1050	Systemutvikling	dagsj	V

Informasjonen hentes via spørringer, for eksempel
"SELECT Kode, Navn FROM Kurs, Personer WHERE Ansvarlig=Brukernavn AND Brukernavn='arild';"

Kode	Navn
INF3170	Arild Waaler

Databaseteori

Mer presist, så består en relasjonsdatabase av to deler, et *skjema* og en *instans* over skjemaet.

Disse svarer til *data* og *metadata* (data om data).

Skjema: Signatur

Består av en *signatur* og en mengde *integritetsregler* (constraints).

Signaturen er en samling relasjonssymboler med navngitte attributter, hver med sitt domene.

Personer(Brukernavn, Navn, FDato, Stillingskode)
Kurs(Kode, Tittel, Ansvarlig, Semester)

Domenet lar vi være implisitt; typisk int, string, etc.

Skjema: Constraints

Integritetsreglene er restriksjoner på hvordan instansen skal se ut.

"Kurskoden er unik" eller "Alle kursansvarlige skal finnes i tabellen Personer"

Man skriver gjerne slikt som formler i et logisk språk.

$$\forall x \exists y \exists z \exists v. \Big(\mathrm{Kurs}(x, y, z, v) \rightarrow \exists a \exists b \exists c. \mathit{Personer}(x, a, b, c) \Big)$$

Vanlig å ha signaturen implisitt: Skjemaet Σ er da en mengde integritetsregler.

Instans

Gitt et skjema Σ, så er en instans over Σ en mengde tupler (rader) for hvert relasjonssymbol i skjemaet, med

- riktig aritet og domener, og som
- oppfyller integritetsreglene i Σ!

Vi kaller elementene i tuplene verdier. Kan droppe navngitte attributter, og bruke indekser istedenfor (aritet må angis).

$\mathrm{Kurs}_4 = \{\langle \mathrm{INF3170}, \mathrm{Logikk}, \mathrm{arild}, \mathrm{H}\rangle,$
$\langle \mathrm{INF1050}, \mathrm{Systemutvikling}, \mathrm{dagsj}, \mathrm{V}\rangle\}$

Kode	Tittel	Ansvarlig	Semester
INF3170	Logikk	arild	H
INF1050	Systemutvikling	dagsj	V

Spørringer

Flere språk for spørringer: Relational algebra, relational calculus, SQL.

I SQL, noe a la "SELECT $A_1, A_2, A_3 \ldots$ FROM $T_1, T_2, T_3 \ldots$ WHERE BETINGELSER;"

Svaret er en ny tabell, gitt ved tupler fra $T_1 \times T_2 \times T_3 \cdots$ som oppfyller betingelsene.

Typiske betingelser er likhet/ulikhet mellom attributter, matematiske operasjoner, og aggregering (summering og slikt), med konnektiver som AND, OR, NOT.

Problemstillinger

Besvare spørringer: Finn svarene til en spørring Q over en instans D.

Optimisere spørringer: Kan en spørring forenkles?

Special case: *Query containtment*. Er svarene til Q_1 alltid en del av svarene til Q_2?

Database analysis: Egenskaper ved Σ og D. Finnes det en D som oppfyller betingelsene i Σ?

Fra databaser til logikk

Vi antar inntil videre at $\Sigma = \emptyset$. Bare signatur, ingen constraints.

Vi kaller instansene for databaser, og snakker gjerne om database over Σ.

Fra database til modell

Vi kan tolke et skjema som en (logisk) signatur, ved å la relasjonene være predikatsymboler.

Så kan vi oversette en instans til en Herbrandmodell $\mathcal{M}$:

- Alle verdier er konstanter, og tolkes som seg selv.
- For predikatsymboler lar vi
  $P^{\mathcal{M}} = \{\langle c_1^{\mathcal{M}}, \ldots, c_n^{\mathcal{M}} \rangle \mid \langle c_1, \ldots, c_n \rangle \text{ er i relasjonen til } P\}$.

La oss skrive $D^{\mathcal{M}}$ for modellen vi får fra instansen D. Dette er en *endelig* modell!

Fra spørringer til formler

Siden en database svarer til en (endelig) modell, så bør formler svare til spørringer.

Gitt en lukket førsteordens formel ϕ og en database D, så kan vi sjekke om $\mathrm{D}^{\mathcal{M}} \models \phi$.

Dette er en *boolsk* spørring, "finnes det noe i D som passer?".

Fra spørringer til formler

Siden en database svarer til en (endelig) modell, så bør formler svare til spørringer.

Gitt en lukket førsteordens formel ϕ og en database D, så kan vi sjekke om $D^{\mathcal{M}} \models \phi$.

Dette er en *boolsk* spørring, "finnes det noe i D som passer?".
For eksempel, har alle kurs en ansvarlig som står i persontabellen?

$\forall$ Kode, Tittel, Ansvarlig, Semester
Kurs(Kode, Tittel, Ansvarlig, Semester) $\rightarrow$
$\exists$ Navn, FDato, Stillingskode
Personer(Ansvarlig, Navn, FDato, Stillingskode)

$$\forall x, y, z, v. \text{Kurs}(x, y, z, v) \rightarrow \exists x', y', z'. \text{Personer}(z, x', y', z')$$

Spørringer som returnerer data

SQL-spørringen i begynnelsen returnerer data. Slike spørringer svarer til formler med frie variable.

Vi husker at en variabel er *fri* i en formel hvis den ikke er innefor skopet til en kvantor. Vi skriver $\mathrm{FV}(\phi)$ for mengden av de frie variablene i ϕ.

For eksempel, hvis
$\phi = \forall x, y, v.\mathrm{Kurs}(x, y, z, v) \rightarrow \exists y', z'.\mathrm{Personer}(z, x', y', z')$, så er
$\mathrm{FV}(\phi) = \{z, x'\}$

Intuitivt, finn alle z, x' slik at...

Spørringer med frie variabler

La D være en database over Σ, og ϕ en formel med $FV(\phi) = \{x_1, \ldots, x_k\}$.

Vi definerer svarene til ϕ over D som mengden av tupler $\langle c_1, \ldots, c_k \rangle$ med konstanter fra Σ slik at $D^{\mathcal{M}} \models \phi[c_1/x_1, \ldots, c_k/x_k]$.

Vi skriver $Ans(\phi, D)$ for denne mengden.

Spørringer med frie variabler

La D være en database over Σ, og ϕ en formel med $FV(\phi) = \{x_1, \ldots, x_k\}$.

Vi definerer svarene til ϕ over D som mengden av tupler $\langle c_1, \ldots, c_k \rangle$ med konstanter fra Σ slik at $D^{\mathcal{M}} \models \phi[c_1/x_1, \ldots, c_k/x_k]$.

Vi skriver $Ans(\phi, D)$ for denne mengden.

La D være $\{Pab, Pbc\}$ og $\phi = \exists y(Pxy \wedge Pyz)$. Da er $Ans(\phi, D) = \{\langle a, c \rangle\}$.

Query answering

Vi har sett at besvare spørring = sjekke hvorvidt $\mathrm{D}^{\mathcal{M}} \models \phi$. Hvordan gjør man det?

Siden $\mathrm{D}^{\mathcal{M}}$ er en endelig modell, kan vi skrive den ned — som en stor konjunksjon $\mathrm{f}(\mathrm{D}^{\mathcal{M}})$ av alt som er sant og usant.

Da kan vi sjekke $\mathrm{D}^{\mathcal{M}} \models \phi$ ved å sjekke $\mathrm{f}(\mathrm{D}^{\mathcal{M}}) \models \phi$ via $\mathrm{f}(\mathrm{D}^{\mathcal{M}}) \vdash \phi$.

Databaser som formler

La $\mathcal{M}$ være endelig. For en relasjon $P^{\mathcal{M}}$ fra $\mathcal{M}$ lar vi

$$f^{+}(P^{\mathcal{M}}) = \bigwedge \left\{ P(\bar{c_1}, \ldots, \bar{c_n}) \mid \langle c_1, \ldots, c_n \rangle \in P^{\mathcal{M}} \right\}$$

og

$$f^{-}(P^{\mathcal{M}}) = \bigwedge \left\{ \neg P(\bar{c_1}, \ldots, \bar{c_n}) \mid \langle c_1, \ldots, c_n \rangle \notin P^{\mathcal{M}} \right\}$$

Så lar vi $f(\mathcal{M})$ være konjunksjonen av $f^{+}(P^{\mathcal{M}}) \wedge f^{-}(P^{\mathcal{M}})$ for alle relasjoner i $\mathcal{M}$.

Funker ikke på uendelige modeller.

Sammenheng endelige modeller og deres formler

Theorem ($\mathfrak{f}(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $\mathfrak{f}(\mathcal{M}) \models \phi$.

Sammenheng endelige modeller og deres formler

Theorem ($\mathfrak{f}(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $\mathfrak{f}(\mathcal{M}) \models \phi$.

Hvis-delen: Hvis $\mathfrak{f}(\mathcal{M}) \models \phi$, så er ϕ sann i alle modeller av $\mathfrak{f}(\mathcal{M})$.

Holder å vise at $\mathcal{M}$ er en modell av $\mathfrak{f}(\mathcal{M})$. Da er ϕ sann i $\mathcal{M}$.

Sammenheng endelige modeller og deres formler

Theorem ($\mathfrak{f}(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $\mathfrak{f}(\mathcal{M}) \models \phi$.

Hvis-delen: Hvis $\mathfrak{f}(\mathcal{M}) \models \phi$, så er ϕ sann i alle modeller av $\mathfrak{f}(\mathcal{M})$.

Holder å vise at $\mathcal{M}$ er en modell av $\mathfrak{f}(\mathcal{M})$. Da er ϕ sann i $\mathcal{M}$.

$\mathfrak{f}(\mathcal{M})$ er en konjunksjon av grunne literaler. La A være et vilkårlig slikt literal.

Sammenheng endelige modeller og deres formler

Theorem ($f(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $f(\mathcal{M}) \models \phi$.

Hvis-delen: Hvis $f(\mathcal{M}) \models \phi$, så er ϕ sann i alle modeller av $f(\mathcal{M})$.

Holder å vise at $\mathcal{M}$ er en modell av $f(\mathcal{M})$. Da er ϕ sann i $\mathcal{M}$.

$f(\mathcal{M})$ er en konjunksjon av grunne literaler. La A være et vilkårlig slikt literal.

Hvis $A = P(c_1, \ldots, c_n)$, så har vi at $\mathcal{M} \models P(c_1, \ldots, c_n)$.

Tilsvarende for $A = \neg P(c_1, \ldots, c_n)$.

Sammenheng endelige modeller og deres formler

Theorem ($\mathfrak{f}(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $\mathfrak{f}(\mathcal{M}) \models \phi$.

Bare hvis-delen: Induksjon på formler.

Sammenheng endelige modeller og deres formler

Theorem ($\mathfrak{f}(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $\mathfrak{f}(\mathcal{M}) \models \phi$.

Bare hvis-delen: Induksjon på formler.

Base case: Hvis ϕ er atomær, argument som over. Siden $\mathcal{M} \models \phi$, så er ϕ en konjunkt i $\mathfrak{f}(\mathcal{M})$.

Ergo må alle modeller som oppfyller $\mathfrak{f}(\mathcal{M})$ også oppfylle ϕ.

Sammenheng endelige modeller og deres formler

Theorem ($\mathfrak{f}(\mathcal{M})$ er korrekt)

La $\mathcal{M}$ være en endelig modell, og ϕ en lukket formel. Da har vi at $\mathcal{M} \models \phi$ hvis og bare hvis $\mathfrak{f}(\mathcal{M}) \models \phi$.

Bare hvis-delen: Induksjon på formler.

Base case: Hvis ϕ er atomær, argument som over. Siden $\mathcal{M} \models \phi$, så er ϕ en konjunkt i $\mathfrak{f}(\mathcal{M})$.

Ergo må alle modeller som oppfyller $\mathfrak{f}(\mathcal{M})$ også oppfylle ϕ.

Resten likner på kompletthetsbeviset for LK.

Oppsummering QA

Databaser kan representeres som endelige modeller, og spørringer som formler med frie variable.

For å besvare en spørring, kan vi sjekke $\mathrm{D}^{\mathcal{M}} \models \phi$.

Dette kan sjekkes via bevissøk, siden endelige modeller kan representeres som formler.

Optimisering av spørringer

Hvis vi har en komplisert spørring, eller flere spørringer å kjøre, kan deler være overflødige (redundant).

Da kan spørringen optimiseres. For eksempel er $P(x, x) \wedge \exists y P(x, y)$ ekvivalent til $P(x, x)$.

$P(x, y) \wedge Q(y, y) \wedge R(y, z)$ er redundant for $P(x, y) \wedge Q(y, z) \wedge R(z, v)$.

Vi ønsker å oppdage slike ting uten å evaluere spørringene.

Query containment

Definisjon (Query containment)

La ϕ og ψ være to spørringer slik at $FV(\phi) = FV(\psi)$. Vi sier at ϕ er inneholdt (contained) i ψ hvis $Ans(\phi, D) \subseteq Ans(\psi, D)$ for alle D.

Med andre ord, svarene til ϕ er alltid en del av svarene til ψ.

Hvis $\phi \subseteq \psi$ og $\psi \subseteq \phi$, så er spørringene ekvivalente.

Med et logisk perspektiv kan vi lett finne en måte å teste query containment på.

Query containment, semantikk

La ϕ og ψ være to lukkede formler.

Theorem

Vi har $\phi \subseteq \psi$ *hvis og bare hvis* $\phi \models \psi$.

Hvorfor det? Jo, fordi alle databaser = alle endelige modeller.

$\phi \subseteq \psi$ betyr at for alle $\mathrm{D}^{\mathcal{M}}$, hvis $\mathrm{D}^{\mathcal{M}} \models \phi$, så $\mathrm{D}^{\mathcal{M}} \models \psi$. Det er definisjonen av logisk konsekvens.

Hva hvis vi har frie variable?

QC med frie variable

Siden QC skal gjelde for *alle* svar $\vec{c}$, må $\phi[\vec{c}] \models \psi[\vec{c}]$ gjelde for alle $\vec{c}$.

Per deduksjonsteorem er dette det samme som $\models \phi[\vec{c}] \rightarrow \psi[\vec{c}]$.

Kan fange dette via *universell tillukning*.

QC med frie variable

Siden QC skal gjelde for *alle* svar $\vec{c}$, må $\phi[\vec{c}] \models \psi[\vec{c}]$ gjelde for alle $\vec{c}$.

Per deduksjonsteorem er dette det samme som $\models \phi[\vec{c}] \rightarrow \psi[\vec{c}]$.

Kan fange dette via *universell tillukning*.

La ϕ være en formel med frie variable $FV(\phi) = \{x_1, \ldots, x_k\}$. Den universelle tillukningen av ϕ er $\forall x_1, \ldots, x_k \phi$.

La ϕ og ψ være to formler med $FV(\phi) = FV(\psi) = \{x_1, \ldots, x_k\}$.

Theorem (QC og logisk konsekvens)

Vi har $\phi \subseteq \psi$ *hvis og bare hvis* $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

QC med frie variable, bevis del en

La ϕ og ψ være to formler med $FV(\phi) = FV(\psi) = \{x_1, \ldots, x_k\}$.

Theorem (QC og logisk konsekvens)

Vi har $\phi \subseteq \psi$ *hvis og bare hvis* $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

Hvorfor det? Vel, bare hvis-delen funker greit. Anta $\phi \subseteq \psi$. Ta vilkårlige $D^{\mathcal{M}}$ og $c_1, \ldots, c_k$ slik at $D^{\mathcal{M}} \models \phi[c_1/x_1, \ldots, c_k/x_k]$, som vil si at $\langle c_1, \ldots, c_k \rangle \in Ans(\phi, D)$.

Siden $\phi \subseteq \psi$, så er $\langle c_1, \ldots, c_k \rangle$ også et svar til ψ, og da har vi at $D^{\mathcal{M}} \models \psi[c_1/x_1, \ldots, c_k/x_k]$.

Siden $D^{\mathcal{M}}$ og $c_1, \ldots, c_k$ var vilkårlige, har vi vist at $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

QC med frie variable, bevis del to

La ϕ og ψ være to formler med $FV(\phi) = FV(\psi) = \{x_1, \ldots, x_k\}$.

Theorem (QC og logisk konsekvens)

Vi har $\phi \subseteq \psi$ *hvis og bare hvis* $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

Hvis-delen vises ved kontraposisjon. Anta at $\phi \not\subseteq \psi$. Da finnes en D slik at $Ans(\phi, D) \not\subseteq Ans(\psi, D)$.

QC med frie variable, bevis del to

La ϕ og ψ være to formler med $FV(\phi) = FV(\psi) = \{x_1, \ldots, x_k\}$.

Theorem (QC og logisk konsekvens)

Vi har $\phi \subseteq \psi$ *hvis og bare hvis* $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

Hvis-delen vises ved kontraposisjon. Anta at $\phi \not\subseteq \psi$. Da finnes en D slik at $Ans(\phi, D) \not\subseteq Ans(\psi, D)$.

Da kan vi ta $\langle c_1, \ldots, c_k \rangle \in Ans(\phi, D) - Ans(\psi, D)$. Siden $\langle c_1, \ldots, c_k \rangle \in Ans(\phi, D)$, så har vi at $D^{\mathcal{M}} \models \phi[c_1/x_1, \ldots, c_k/x_k]$, men at $D^{\mathcal{M}} \not\models \psi[c_1/x_1, \ldots, c_k/x_k]$.

Ergo har vi at $D^{\mathcal{M}} \not\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

Oppsummering QC

Teoremet gir oss følgende algoritme: For å sjekke $\phi \subseteq \psi$, sjekk $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

Vi vet fra før at QA er $D^{\mathcal{M}} \models \phi$, ekvivalent $f(D^{\mathcal{M}}) \models \phi$.

Oppsummering QC

Teoremet gir oss følgende algoritme: For å sjekke $\phi \subseteq \psi$, sjekk $\models \forall x_1, \ldots, x_k(\phi \rightarrow \psi)$.

Vi vet fra før at QA er $D^{\mathcal{M}} \models \phi$, ekvivalent $f(D^{\mathcal{M}}) \models \phi$.

Theorem (QA er QC)

La D *være en instans og* ϕ *en formel med* $FV(\phi) = \{x_1, \ldots, x_n\}$. *Da har vi at* $\langle c_1, \ldots, c_n \rangle \in Ans(\phi, D)$ *hvis og bare hvis* $f(D^{\mathcal{M}}) \subseteq \phi[c_1/x_1, \ldots, c_n/x_n]$.

Bevis: Ukeoppgave!

Men, QC er *ikke* QA (hvorfor?)

Integritetsregler

Vi kan bruke logikk til å analysere et skjema Σ med constraints.

Gyldighet av instans $= \mathrm{D}^{\mathcal{M}} \models \bigwedge \Sigma$.

Integritetsregler

Vi kan bruke logikk til å analysere et skjema Σ med constraints.

Gyldighet av instans $= \mathrm{D}^{\mathcal{M}} \models \bigwedge \Sigma$.

Men, "finnes det en instans som er gyldig" $\neq \bigwedge \Sigma \models \bot$ (hvorfor?)

Finnes faktisk ikke noen formel ϕ som funker for vilkårlig Σ.

Noen dårlige nyheter

Theorem (Trakhtenbrot 1949)

For førsteordens formler ϕ og ψ er $\phi \models \psi$ ikke avgjørbart, selv over endelige modeller.

Per deduksjonsteorem, $\models \phi \rightarrow \psi$ er ekvivalent med $\phi \models \psi$.

Med andre ord, ingen QC in the general case.

Kan vi finne et pent spesialtilfelle?

Og hva med QA?