

Integración de Información usando Vistas Lógicas

Jeffrey D. Ullman

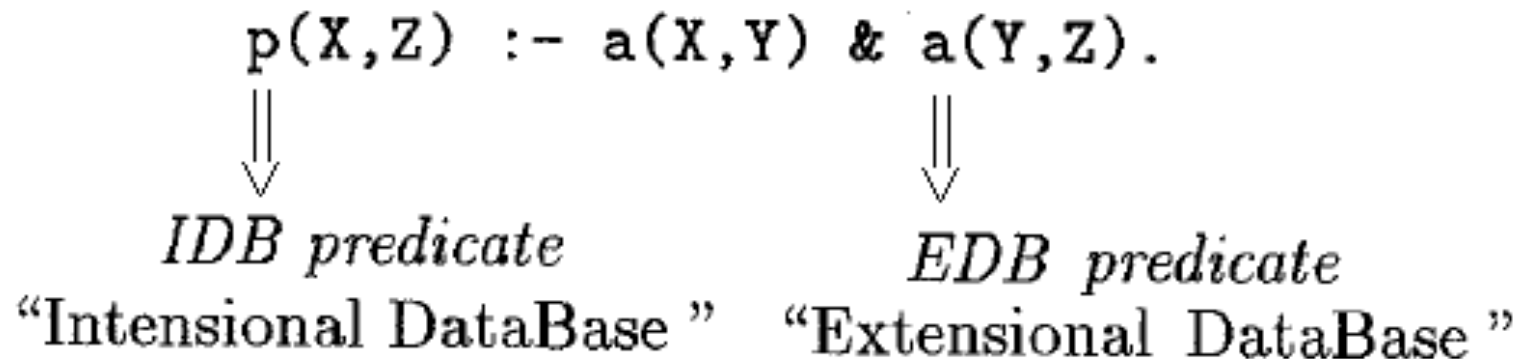
Presentación: Juan Quinteros

Abstracto

- Algunas ideas sobre Integración de Información pueden pensarse como la Construcción de Respuestas a Consultas usando Vistas que representan las capacidades de fuentes de información.
- Revisión de la base formal de estas técnicas, relacionadas con algoritmos de inclusión para Consultas Conjuntivas y/o Programas Datalog.
- Comparación entre “Information Manifold” y “Tsimmis”.

Base teórica

- Consultas conjuntivas, Programas Datalog, inclusión.
- Ejemplo 1- La siguiente es una regla que habla sobre:



Consultas conjuntivas

- inclusión: una cuestión crucial.
- Q_1, Q_2 Consultas conjuntivas (CQ).

$$Q_1 \subseteq Q_2$$

para todas las Bases de Datos D si $Q_1(D)$ es siempre subset de $Q_2(D)$.

- Equivalencia: inclusión mutua.

Prueba de inclusión

- Para probar si Q_1 está contenida en Q_2
 - "Congelamos" el cuerpo de Q_1 , convirtiendo cada uno de sus subobjetivos en hechos de la base de datos.
 - Aplicamos Q_2 a esta base de datos Canónica
 - Si la cabeza congelada de Q_1 se deriva de Q_2 , entonces Q_1 está contenida en Q_2 .

Ejemplo:

$$Q_1: p(X, Z) \text{ :- } a(X, Y) \ \& \ a(Y, Z)$$
$$Q_2: p(X, Z) \text{ :- } a(X, U) \ \& \ a(V, Z)$$

Prueba de inclusión

- Si es negativa, la base canónica encontrada es un contraejemplo.
- De ser positiva, existe un homomorfismo de las variables de $Q1$ a las de $Q2$, según asignamos. Para cualquier database D , podemos seguir una aplicación exitosa de $Q1$, desde los símbolos de D a las variables de $Q1$, a las variables de $Q2$. Estas deben ser también exitosas.
- La prueba de inclusión es NP-completa, pero lineal en muchos casos.

Consultas Conjuntivas con Negación

- Test de inclusión ligeramente más complejo.
Completo para la clase Π_2^P .
Son los problemas que se pueden expresar como $\{w | (\forall x)(\exists y)\phi(w, x, y)\}$, donde x e y son strings de largo polinomial con respecto a w , y ϕ es una función computable en tiempo polinomial.
- Involucra explorar un número exponencial de bases de datos canónicas, cualquiera de ellas puede dar un contraejemplo.

Prueba de inclusión para Consultas Conjuntivas con Negación

- - Hacer todas las agrupaciones de variables posibles, cada una será una base de datos canónica (a partir de sólo los subobjetivos positivos), D_i .
- - Para cada D_i , asignar a cada variable de un bloque una misma constante.

Prueba inclusión de CQ con negación (cont.)

- - Ver si D_i cumple los subobjetivos de Q_1 .
- - Si los cumple, probar si algún $Q_2(D_i')$ incluye la cabeza congelada de Q_1 . D_i' es cualquier superset de D_i formado al agregar otras tuplas que usen el mismo set de símbolos que D_i (excluyendo subobjetivos negativos de Q_1). Al determinar qué es la cabeza congelada de Q_1 , usamos la misma sustitución de constantes por variables que generó a D_i .

Ejemplo

$Q_1: p(X,Z) \text{ :- } a(X,Y) \ \& \ a(Y,Z) \ \& \ \text{NOT } a(X,Z)$
 $Q_2: p(A,C) \text{ :- } a(A,B) \ \& \ a(B,C) \ \& \ \text{NOT } a(A,D)$

	Partition	Canonical Database	Outcome
1)	$\{X\}\{Y\}\{Z\}$	$\{a(0,1), a(1,2)\}$	both yield head $p(0,2)$
2)	$\{X,Y\}\{Z\}$	$\{a(0,0), a(0,1)\}$	Q_1 body false
3)	$\{X\}\{Y,Z\}$	$\{a(0,1), a(1,1)\}$	Q_1 body false
4)	$\{X,Z\}\{Y\}$	$\{a(0,1), a(1,0)\}$	both yield head $p(0,0)$
5)	$\{X,Y,Z\}$	$\{a(0,0)\}$	Q_1 body false

CQs con comparaciones aritméticas

$Q_1: p(X,Z) :- a(X,Y) \ \& \ a(Y,Z) \ \& \ X < Y$

$Q_2: p(A,C) :- a(A,B) \ \& \ a(B,C) \ \& \ A < C$

- Similar al caso anterior, pero ahora, para cada partición, debemos ordenar la asignación de variables.

Programas Datalog

Regresando a las reglas originales, sin negación ni comparación...

- Son colecciones de reglas.
- Tienen una interpretación natural, en que consideramos que los IDB comienzan vacíos.
- Luego usamos las reglas para inferir nuevos hechos IDB, hasta que no se pueda más.

Ejemplo

- 1) $p(X, Z) \text{ :- } q(X, Y) \ \& \ b(Y, Z) .$
- 2) $q(X, Y) \text{ :- } a(X, Y) .$
- 3) $q(X, Z) \text{ :- } a(X, Y) \ \& \ p(Y, Z) .$

inclusión con Datalog

- inclusión de programas Datalog: indecidible.
- inclusión de un programa Datalog en una CQ: doblemente exponencial.
- Caso importante para Integración de Información: inclusión de CQ en un programa Datalog. No más complejo que inclusión en CQs.

Prueba de inclusión

- Nuevamente congelamos la cabeza de $Q1$ para hacer una base canónica D .
- Probamos si $P(D)$ contiene la cabeza congelada de $Q1$.
- Única diferencia: debemos seguir aplicando las reglas del programa hasta que se derive la cabeza, o no se infieran más hechos IDB.

Sintetizando Consultas a partir de Vistas

- Los algoritmos de inclusión se conectan a la Integración de Información a través de este proceso.
- Se tiene un número de predicados EDB, para los cuales se usa p en la figura. Representan los conceptos básicos usados en las consultas, pero en realidad no existen como relaciones guardadas.

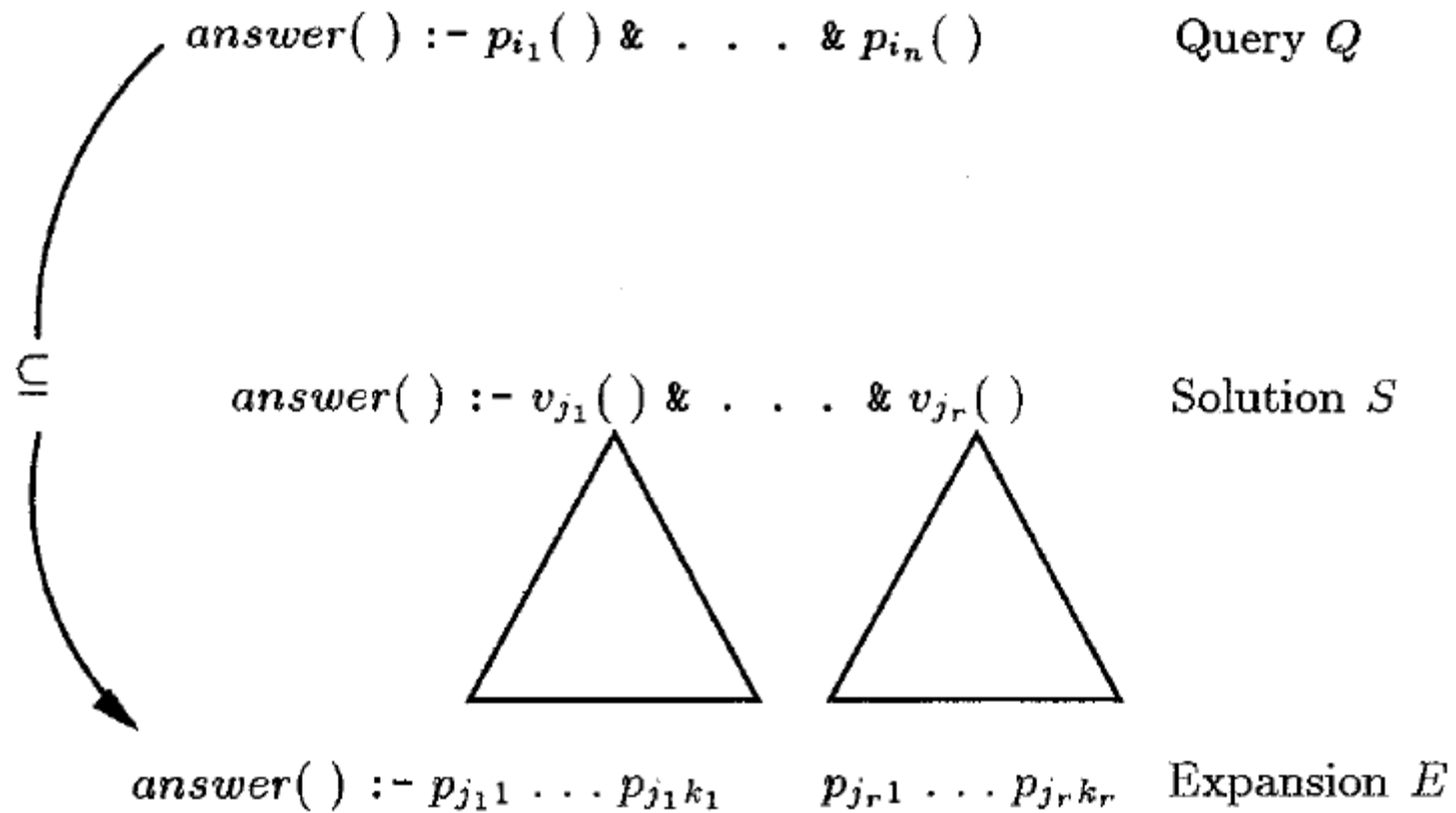
Sintetizando Consultas a partir de Vistas (cont.)

- También hay un número de Vistas, representadas por v en la figura.
- Representan recursos de los que dispone el Integrador para ayudar a contestar consultas.
- Tienen definiciones en base a predicados EDB. Se supone en el paper que son CQ.

Resolviendo Consultas mediante Vistas

- Una consulta Q se expresa mediante los predicados EDB.
- El problema consiste en encontrar una “solución” S para la consulta Q .
- Esta solución estará expresada en términos de las vistas.
- Para que sea valida, al expandir las vistas debe quedar una expansión E equivalente a Q .

Figura



Ejemplo

- Supongamos un solo predicado EDB $p(X,Y)$, que indica que Y es padre de X .
- Supongamos un par de vistas:

$$\begin{aligned}v1(Y,Z) &:- p(X,Y) \ \& \ p(Y,Z) . \\v2(X,Z) &:- p(X,Y) \ \& \ p(Y,Z) .\end{aligned}$$

- Supongamos que queremos hacer una consulta a este sistema de información sobre los abuelos de un individuo 0.

$$q(C) \text{ :- } p(0,A) \ \& \ p(A,B) \ \& \ p(B,C)$$

Ejemplo (cont.)

- El problema consiste en encontrar una CQ cuyos subobjetivos sólo usen los predicados V_1 y V_2
- Una solución es:

$$s1(C) \text{ :- } v2(O,D) \ \& \ v1(D,C)$$

- Ya que al expandir las vistas obtenemos la expansión:

$$e1(C) \text{ :- } p(O,E) \ \& \ p(E,D) \ \& \ p(F,D) \ \& \ p(D,C)$$

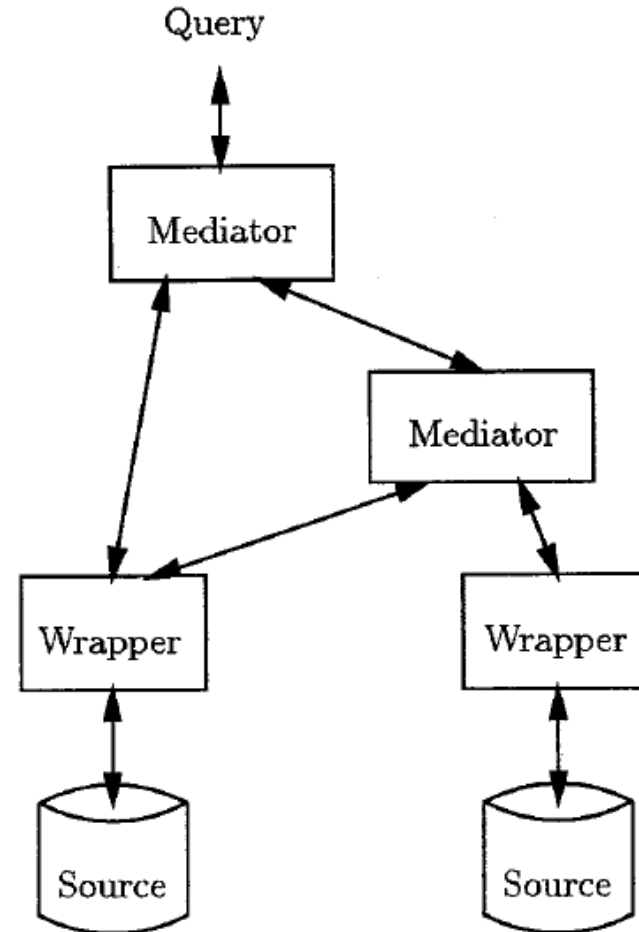
- Podemos usar inclusión mutua para probar que es equivalente a Q.

Teoremas de Solución Mínima

- Parecería por el ejemplo anterior que uno sólo puede adivinar soluciones potenciales y probarlas usando inclusión.
- Sin embargo hay teoremas que limitan la búsqueda, y demuestran que si bien el problema de expresar una consulta en vistas no es peor que NP-Completo.

Sistemas de Integración de Información

- Nuestro gran problema...
- Una arquitectura típica:
- Proyectos de investigación:
 - Information Manifold
 - Tsimmis



Information Manifold

- Se basa en un dialecto de lógica descriptiva llamado CARIN.
- Lógica descriptiva puede pensarse como un programa Datalog no-recursivo con predicados IDB restringidos a ser unarios, aunque tiene ciertas capacidades extras.
- El paper usa Datalog para explicarlo.

Arquitectura del Information Manifold

- Posee una colección de predicados globales, en cuyos términos se expresan todas las consultas.
- Cada fuente de información se asocia a una o más vistas (también expresadas en términos de los predicados globales).
- Las vistas sólo proveen *algunos* de los hechos derivables de su definición.

Arquitectura del Information Manifold (cont.)

- La solución de una consulta es la unión de todas las CQ mínimas (sobre las vistas) contenidas en ella.
- También se asocian a una fuente cero o más restricciones. Estas son garantías que ciertos hechos que podrían estar presentes en la vista, en realidad no aparecerán.

Tsimmis

- "The Stanford-IBM Manager of Multiple Information Sources"
- Sigue la arquitectura de Envolturas y Mediadores que se comunican unos con otros.
- La comunicación se basa en un modelo de datos llamado OEM (*Object-Exchange Model*) y un lenguaje de consultas llamado MSL (*Mediator Specification Language*).

OEM

- Está orientado a objetos y la data se asume está organizada en objetos.
- Un objeto consiste en:
 - -Un label
 - -Un tipo para el valor del objeto. (puede ser “set de objetos OEM
 - -Un valor para el objeto.

MSL

- Las oraciones en MSL son reglas lógicas, pero no son exactamente Datalog. Más bien usa una forma de lógica de objetos.

Comparando Tsimmis e IM.

- Las consultas en IM están en términos de los predicados globales, que se traducen en vistas. En Tsimmis, están en términos de predicados sintetizados en un mediador.
- Niveles de mediación: IM asume que hay dos niveles, las fuentes y un “mediador global”. Tsimmis asume un número ilimitado de niveles.

Comparando Tsimmis e IM

- Agregando fuentes: muy conveniente en IM, pero requiere una reestructuración en Tsimmis.
- Data semiestructurada: Tsimmis soporta la noción que la data no tiene un esquema uniforme.
- Restricciones: Sólo IM tiene un sistema explícito para describirlas.

Comparando Tsimmis e IM

- Generación automática de Componentes: Tsimmis se ha enfocado a la generación automática de Envolturas y Mediadores, IM, sin embargo, no lo hace pues sólo tiene dos niveles de abstracción.

Conclusión

- IM y Tsimmis son acercamientos interesantes al problema de la Integración de información.
- Ambos se basan en los mismos principios de lógica de base de datos.
- Difieren en la lógica interna, el acercamiento a la data semiestructurada y el algoritmo de procesamiento de consultas.