Julie Orzea
Betreuer: Heiko Schepperle
Universität Karlsruhe
Wintersemester 2003/2004

Seminar:

Imperfektion und Datenbanken

Fuzzy funktionale Abhängigkeiten (FFD)

Inhaltsverzeichnis

I) Einführung zu Fuzzy Datenbanken

1) Motivation

Datenbanken werden verwendet, um die Realität zu modellieren und um gewisse Informationen zu beschreiben. Die Realität ist aber nicht immer genau bekannt und exakt. Deswegen möchte man manchmal fuzzy Informationen modellieren und verwenden. Unter „fuzzy" versteht man eine ungenaue oder unscharfe Information, wie zum Beispiel „jung" statt einer genauen Altersangabe oder ein Zeitintervall statt einer Jahresangabe.

2) Fuzzy Datenbanken

Es gibt 3 verschiedenen Hauptkategorien von fuzzy Informationen, die man modellieren will, wie es in [3] erklärt ist:

- Ein Attribut, das nicht sicher ist: {Karine; (Französisch/0,3)}, Karine kann Französisch sprechen mit einer Wahrscheinlichkeit von 0,3.
- Ein Attribut mit mehreren Schwellwerten: {Karine;(Französisch/0,8; Englisch/0,2)}, Karine kann entweder Französisch oder Englisch. Sie kann Französisch mit einer Wahrscheinlichkeit von 0,8 und Englisch mit einer Wahrscheinlichkeit von 0,2.
- Ein Attribut, das nicht genau bekannt ist: {Karine, jung}, Karine ist jung, aber ihr genaues Alter ist nicht bekant.

Diese Informationen will man modellieren, um sie später benützen zu können. Wie im nicht-fuzzy Fall möchte man diese Information in Datenbanken strukturieren, wie zum Beispiel in Normalformen. Um Datenbanken zu strukturieren sind funktionale Abhängigkeiten nützlich. Aber normale (nicht-fuzzy) funktionale Abhängigkeiten kann man mit fuzzy Informationen nicht gut benützen, da man oft ähnliche Werte an Stelle gleicher Werte finden wird.

3) Funktionale Abhängigkeit

Eine funktionale Abhängigkeit („Functional Dependency“, FD) (vgl. [4] und [5]) ist eine Funktion, die auf den Attributen einer Relation definiert ist:

$$\mathrm{A} \to \mathrm{B} \iff \forall (t, t') \in R,\ \mathrm{t.A} = \mathrm{t'.A} \Rightarrow \mathrm{t.B} = \mathrm{t'.B}$$

Man möchte diese Funktion auch für den fuzzy Fall verwenden, aber es gibt ein Problem für den Fall, für den ein oder mehrere Attribute unsicher sind. Zum Beispiel, wenn man weiß, dass ein 15jähriger in die Schule geht und man wissen will, ob ein Jugendlicher auch in die Schule geht. Die Gleichheit beider Aussagen ist nicht bestätigt, aber man kann vielleicht doch eine Antwort geben.

II)Verschiedene (fuzzy) funktionale Abhängigkeiten

Man will die Idee modellieren, dass wenn t.A und t‘.A ähnlich sind, auch t.B und t‘.B ähnlich sein müssen (vgl. [2]).

$$\mathrm{A} \sim> \mathrm{B} <=> \forall(t,t') \in R,\ \mathrm{t.A} \approx \mathrm{t'.A}\ => \mathrm{t.B} \approx \mathrm{t'.B}$$

Dafür muss man aber genau sagen, was zwei ähnliche Werte sind und was man durch eine Implikation versteht. Diese zwei Begriffe werden durch zwei Funktionen definiert, nämlich durch die Ähnlichkeitsfunktion (≈) und durch die Implikationsfunktion (~>).

1) Die Ähnlichkeitsfunktion

Die Ähnlichkeit oder Resemblance (≈) wird (in [1]) durch eine Funktion, RES_x: [dom(X)][dom(X)]->[0,1] definiert. Man braucht eine Ähnlichkeitsfunktion RES_A, die auf der Domäne von A definiert ist, und eine Ähnlichkeitsfunktion RES_B, die auf der Domäne von B definiert ist. Die zwei Funktionen sind meistens verschieden. Man kann jede als eine Wahrscheinlichkeitsfunktion interpretieren, nämlich als die Wahrscheinlichkeit, dass t.A und t'.A gleich sind.

2) Die Implikationsfunktion

a) Der allgemeine Fall

Um im fuzzy Fall eine fuzzy funktionale Abhängigkeit zu beschreiben braucht man eine Implikationsfunktion. Da man nicht die Implikationsfunktion des nicht-fuzzy Falls benützen kann, definiert man den allgemeinen Fall dieser Funktion (vgl. [2] und [3]). Dafür benützt man die Ähnlichkeitsfunktion. Eine Implikationsfunktion I wird folgendermaßen definiert:

I : [0,1] [0,1] -> [0,1] ,

A $\sim>_\theta$B ist gültig auf R(U) <=> $(\min_{t,t'\in R} I[t.A \approx t'.A; t.B \approx t'B]) \geq \theta$

θ ist die minimale Wahrscheinlichkeit, dass t.A≈ t.A' und t.B≈ t.B' gilt.

b) Der nicht-fuzzy Fall der Implikationsfunktion

Diese Implikationsfunktion beschreibt den allgemeinen Fall. Es gibt einen Spezialfall, der genau der nicht-fuzzy funktionalen Abhängigkeit entspricht. Die Implikationsfunktion, die man gewöhnlich benutzt, im nicht-fuzzy Fall, ist also ein Spezialfall dieser Funktion. Um auf dieses Spezialfall zu kommen, muss man folgende Parameter wählen:

Der allgemeine Fall:	Der nicht-fuzzy Fall:
θ	1
~> θ	->
≈	=
I[X,Y]	X=>Y (true=1,false=0)
A ~>θB ist gültig auf R(U) <=>	A -> B ist gültig auf R(U) <=>
$(\min_{t,t' \in R} I[t.A \approx t'.A; t.B \approx t'B]) \geq \theta$	$(\min_{t,t' \in R} I[t.A = t'.A; t.B = t'B]) \geq 1$ (t . A = t ' . A) => (t . B = t ' B)

Mit diesen Parametern kommt man auf die gewöhnliche Implikations-Funktion,

(t . A = t ' . A) => (t . B = t ' B)

c) Beispiele von Implikationsfunktionen im fuzzy Fall

Im fuzzy Fall kann man verschiedene Parameter auswählen. Man kann sie entweder so wählen, dass sie leicht nachvollziehbar und verständlich sind, oder dass die Regeln für funktionale Abhängigkeiten (FDs) leicht auf fuzzy funktionale Abhängigkeiten (FFDs) übertragbar sind.

Die folgenden fuzzy funktionalen Abhängigkeiten wurden in [1] erklärt und können benützt werden:

Fuzzy funktionale Abhängigkeiten von Raju und Majumdar:

A $\sim>_{rg}$B ist gültig auf R(U)

<=> $\forall(t,t') \in R,\ \mathrm{RES}_A(\mathrm{t.A}, \mathrm{t'.A}) =>_{\mathrm{rg}}\ \mathrm{RES}_B(\mathrm{t.B}, \mathrm{t'.B})$

a $=>_{rg}$ b = 1, wenn a≤b; 0, sonst

Diese fuzzy funktionale Abhängigkeit ist leicht zu verstehen und ist trotzdem ein allgemeiner Fall von funktionalen Abhängigkeiten. Wenn a und b nur die Werte 0 und 1 haben können, kommt man auf den nicht-fuzzy Fall. Mit diesen fuzzy funktionalen Abhängigkeiten kann man Relationen zerlegen.

Fuzzy funktionale Abhängigkeiten von G. Chen et al:

A $\sim>_\lambda$ B ist gültig auf R(U) mit λ in]0;1]

<=> $\forall(t,t') \in R,\ \mathrm{RES}_A(\mathrm{t.A}, \mathrm{t'.A}) =>_{\lambda}\ \mathrm{RES}_B(\mathrm{t.B}, \mathrm{t'.B})$

a $=>_\lambda$ b = 1 wenn a≤b; b sonst

Diese Implikation ist sehr ähnlich zur ersten und kann auch gut benützt werden, um Eigenschaften des nicht-fuzzy Falls auf den fuzzy Fall zu erweitern.

Fuzzy funktionale Abhängigkeiten von Cubero et al:

A $\sim>_{(\lambda,\mu)}$ B ist gültig auf R(U) mit λ und μ in]0;1]

<=> $\forall(t,t') \in R,\ \mathrm{RES}_A(\mathrm{t.A}, \mathrm{t'.A}) =>_{(\lambda,\mu)}\ \mathrm{RES}_B(\mathrm{t.B}, \mathrm{t'.B})$

a $=>_{(\lambda,\mu)}$ b = 1 wenn a>λ und b>μ; 0 sonst

Diese Implikationsfunktion hat auch gute Eigenschaften, um Erweitungen des nicht-fuzzy Falls anzuwenden. Sie hat 2 verschiedene Werte, einen pro Domäne. Das kann Vorteile haben, zum Beispiel wenn die Werte einer Domäne unabhängig von der anderen sind.

3) Erweitungen vom nicht-fuzzy Fall zum fuzzy Fall

Mit jeder Implikationsfunktion kann man die folgenden Erweitungen vom nicht-fuzzy Fall zum fuzzy Fall machen. Diese Erweitungen wurden in [3] erklärt und der nicht-fuzzy Fall in [4] und [5].

a) Armstrong-Axiome und Ableitungsregeln

Man kann die bekanten Armstrong-Axiome auf den fuzzy Fall erweitern und damit auch auf Ableitungs-Regeln für den fuzzy Fall kommen. Die Armstrong-Axiome und die Ableitungsregeln sind sehr ähnlich zu den Regeln, die man im nicht-fuzzy Fall verwendet. Man muss aber beachten, dass es immer einen Schwellwert gibt. Wenn zwei Regeln vorkommen (dass heißt, wenn man zwei verschiede Schwellwerte hat), verwendet man den kleinsten Schwellwert als Schwellwert für das Ergebnis.

Armstrong-Axiome (Analog zu FDs):

- Wenn Y in X, dann $X \sim>_q Y$ („Reflexivität")
- Wenn $X \sim>_q Y$, dann $XZ \sim>_q YZ$ („Expansivität")
- Wenn $X \sim>_q Y$ und $Y \sim>_\gamma Z$, dann $X \sim>_\mu Z$ mit $\mu=\min(q, \gamma)$ („Transitivität")

Daraus lassen sich die folgenden Ableitungsregeln generieren:

- Wenn $X \sim>_q Y$ und $Y \sim>_\gamma Z$, dann $X \sim>_\mu YZ$ mit $\mu=\min(q, \gamma)$
- Wenn $X \sim>_q Y$ und $WY \sim>_\gamma Z$, dann $XW \sim>_\mu Z$ mit $\mu=\min(q, \gamma)$
- Wenn $X \sim>_q Y$ und Z in Y, dann $X \sim>_q Z$
- Wenn $X \sim>_q Y$, dann $X \sim>_\gamma Y$ mit alle $\gamma \leq \theta$
- $X \sim>_q Y \Leftrightarrow X \sim>_q Y1, Y2, ..., YN \Leftrightarrow X \sim>_q Yi$ für i=1..k

Mit Hilfe dieser Regeln kann man, so wie im nicht-fuzzy Fall, Datenbanken strukturieren.

b) Voll fuzzy funktional abhängig

Y ist von X voll fuzzy funktional abhängig mit einen Grad q ($X \approx>_q Y$), wenn $X \sim>_q Y$ und keine echte Teilmenge A von X gibt, so dass $A \neq ø$ und $A \sim> Y$.

c) Partiell fuzzy funktional abhängig

Y ist von X partiell fuzzy funktional abhängig, wenn Y von X fuzzy funktional abhängig ist ($X \sim>_q Y$), aber nicht voll fuzzy funktional abhängig ($X \approx>_q Y$).

d) Hülle

F^+ ist die Menge aller fuzzy funktionalen Abhängigkeiten, die man durch Ableitungsregeln erzeugen kann.

e) Äquivalenz zweier Mengen von FFDs

Zwei Mengen von FFDs, F und G, sind äquivalent (F <=> G), wenn $F^+=G^+$.

f) Schlüssel und q-Schlüssel

Eine Menge von Attributen K ist ein *Schlüsselkandidat* in der Relation R(U), wobei U die Menge aller Attribute der Relation ist, wenn gilt: K=>U („voll funktional abhängig"). Dieselben Werte von K definieren dieselben Werte von U.

Eine Menge von Attributen K ist ein *q-Schlüsselkandidat* in der fuzzy Relation R(U) wenn gilt: $K \approx>_q U$ („voll fuzzy funktional abhängig"). Dieselben Schwellwerte von K definieren dieselben Schwellwerte von U und ähnliche Werte von K definieren ähnliche Werte von U.

Ein *Schlüssel* (bzw. ein *q-Schlüssel*) wird aus der Menge der Schlüsselkandidaten (bzw. der q-Schlüsselkandidaten) gewählt. Er ist der *Primärschlüssel* (bzw. *q-Primärschlüssel*) dieser Relation. Ein Schlüssel (bzw. ein q-Schlüssel) kann als Referenz in einer anderen Relation benützt werden. In diesem Fall spricht man von einem *Fremdschlüssel*.

Ein *Superschlüssel* (bzw. ein *q-Superschlüssel*) ist eine Menge von Attributen, die mindestens alle Attribute eines Schlüsselkandidaten enthält.

III) Verwendung

1) Integritätsbeschreibung

Fuzzy funktionale Abhängigkeiten kann man gut verwenden, um Integritätsbeschreibungen zu beschreiben. Zum Beispiel kann man sagen, dass die Wahrscheinlichkeit, dass eine junge Person studiert groß ist, während die Wahrscheinlichkeit, dass eine junge Person bereits pensioniert ist, klein ist. Wenn man eine junge Person in der Datenbank einträgt und sagt, dass sie pensioniert ist, kann die Datenbank eine Warnung hervorrufen. Es soll aber kein Fehler geworfen werden, da dieser Fall nicht unmöglich ist. (vgl. [1])

2) Zerlegung in Normalformen

Wie im nicht-fuzzy Fall, kann man in fuzzy Datenbanken Redundanzen haben und deswegen auf Inkohärenzen kommen (vgl. [5]). Deswegen will man manchmal Relationen in Normalformen zerlegen. Die Normalformen, die man im fuzzy Fall benützt, sind sehr ähnlich zu denen, die man im nicht-fuzzy Fall benützt. Man muss aber darauf achten, dass im fuzzy Fall immer ein Schwellwert ins Spiel kommt. (vgl. [3]) Wenn man einige Relationen zerlegt, ohne auf die Zerlegung in Normalformen zu achten, kann ein Teil der Information verloren gehen.

a) 1. q-Normalform

Eine Relation R ist in der *1. q-Normalform (1NF)* genau dann, wenn jedes Attribut in der Relation einen einzigen Wert hat. Das heißt, dass alle Attribute atomar sein müssen. Sie können aber, im fuzzy Fall, eine Wahrscheinlichkeit enthalten oder es können mehrere Schwellwerte existieren, von denen genau einer wahr ist.

Zum Beispiel die folgende Relation, {Karine, Französisch/0,7 und Englisch/0,7}, (Karine kann Französisch mit einer Wahrscheinlichkeit von 0,7 UND Englisch mit einer Wahrscheinlichkeit von 0,7) geht nicht, während die Relation {Karine, Deutsch/0,5 ODER Englisch/0,5} (Karine kann Französisch mit einer Wahrscheinlichkeit von 0,5 oder Englisch mit eine Wahrscheinlichkeit von 0,5) in 1. q-Normalform ist.

b) 2. q-Normalform

Eine Relation R ist in *2. q-Normalform (2NF)* genau dann, wenn R in 1. q-Normalform ist und wenn alle Attribute voll fuzzy funktional abhängig vom Schlüssel sind. Für den q-Schlüssel K von R und für eine Menge von Nichtschlüsselattributen A gilt:

wenn $K \sim>_q A$ dann $K \approx>_q A$.

c) 3. q-Normalform

Eine Relation ist in *3. q-Normalform (3NF)* genau dann, wenn R in 2. q-Normalform ist und wenn für jede funktionale Abhängigkeit $X \sim>_q A$ mit A Nichtschlüsselattribut gilt: X ist Superschlüssel.

Als Folgerung muss *A* von jedem Schlüsselkandidaten voll funktional abhängen. Kein Nichtschlüsselattribut kann transitiv von einem Schlüsselkandidaten abhängen.

Zum Beispiel:

R={P,H,S,C},

$\{P,H \sim>_q C;\ H,S \sim>_q P;\ H,C \sim>_q S;\ S \sim>_q P\}$

d) q-Boyce-Codd Normalform

Eine Relation R ist in *q-Boyce-Codd-Normalform (BCNF)* genau dann, wenn R in 3. q-Normalform ist und für jede voll funktionale Abhängigkeit X≈>qY gilt: X ist Schlüssel.

3) Anwendungsbeispiele

Mit einer fuzzy Datenbank und gutgewählten fuzzy funktionalen Abhängigkeiten kann man unbekannte Attribute extrapolieren oder interpolieren.

Zum Beispiel kann man eine Liste von Personen haben mit Name, Alter, Studienzeitraum, Diplom, Aktivität und Gehalt. Man beachtet folgende fuzzy funktionale Abhängigkeiten:

R1: Studienzeitraum, Diplom, Aktivität ~>0,95 Entgelt

R2: Studienzeitraum, Diplom ~> 0,8 Aktivität

R3: Alter ~> 0,8 Aktivität

R4: Alter ~> 0,85 Studienzeitraum

R5: Studienzeitraum ~>0,6 Alter

Man kennt die folgenden Personen:

Name	Alter	Studienzeitraum	Diplom	Aktivität	Gehalt
Julie	jung	2030iger Jahre	Ing	Arbeitet	wenig
Jerome		2030iger Jahre	Ing		
Nico		2020iger Jahre		Student	0
Karine	mittelalt	1980iger Jahre	Ing	Arbeitet	viel
Mars		1960iger Jahre	Ing	Rente	wenig

Mit der Beziehung R5 kann man interpolieren, dass Jerome jung sein muss, dass er wahrscheinlich arbeitet, aber ziemlich wenig Gehalt bekommt.

Mit derselben Beziehung kann man extrapolieren, dass Nico auch jung ist, während Mars alt ist.

Diese Eigenschaften von das fuzzy Datenbanken kann man auch in anderen Bereichen verwenden, wie zum Beispiel für das logisches Schließen ("analogical reasoning and cooperative answering“).

IV) Zusammenfassung

In dieser Arbeit wurde erklärt, was eine fuzzy Datenbank ist, wie man fuzzy funktionale Abhängigkeiten definiert und wie man sie benützen kann. Fuzzy funktionale Abhängigkeiten definiert man mit Hilfe von Ähnlichkeitsfunktionen und einer allgemeinen Anwendung der Implikationsfunktion. Mit den so definierten fuzzy funktionalen Abhängigkeiten kann man eine Datenbank strukturieren, um Redundanz und Inkohärenzen zu vermeiden. Man kann auch Integritätsbedingenung beschreiben, unbekannte Attribute extrapolieren oder interpolieren. Mit diese Datenbanken und den passenden fuzzy funktionalen Abhängigkeiten kann man auch logisches Schließen durchführen.

Um diese Datenbanken darzustellen und zu benützen muss man sehr auf die Ähnlichkeitsfunktion aufpassen. Diese hat einen sehr großen Einfluss auf das Ergebnis. Wenn sie nicht gut gewählt ist, kommt man auf Ergebnisse, die nicht verwendbar sind. Mit einer guten Funktion kann man aber an interessante und nützliche Ergebnisse kommen.

V) Literatur

[1] Patrick Bosc, Didier Dubois, Henri Prade: Fuzzy functional dependencies - an overview and a critical discussion. In: Proceedings of the Third IEEE International Conference on Fuzzy Systems, pp. 325-330. IEEE, 1994.

[2] Guoqing Chen, P. Bosc, J. Kacprzyk, Fuzzy Functional Dependency and a series of design issues of fuzzy relational databases. In: Fuzziness in Database Management Systems.

[3] Guoqing Chen, Etienne E. Kerre, Jaques Vandenbulcke, Normalization based on fuzzy functional dependency in a fuzzy relational data model, in Information Systems 21 (3), 1996, http://portal.acm.org/citation.cfm?id=229539&jmp=cit&dl=GUIDE&dl=guide.

[4] Sebastian Abeck, Peter C. Lockemann, Jochen Schiller. Verteilte Informationssysteme. dpunkt Verlag, 2002.

[5] Peter C. Lockermann, Vorlesungsfolien zu „Datenbankeinsatz“, Universität Karlsruhe, 2003. http://www.ipd.uka.de/~dbe/folien.html.