Fuzzy logic programs and attribute implications

Tomas Kuhr, Vilem Vychodil

DATA ANALYSIS AND MODELING LAB

Palacky University, Olomouc, Czech Republic

european social fund in the czech republic

EUROPEAN UNION

MINISTRY OF EDUCATION, YOUTH AND SPORTS

OP Education for Competitiveness

INVESTMENTS IN EDUCATION DEVELOPMENT

Problem setting

- In this talk we will deal with two systems for logic reasoning with graded if-then rules:
- Fuzzy Attribute Logic (FAL)
  - Developed primarily for describing if-then dependencies in fuzzy object-attribute relational data.
  - Introduced by Belohlavek and Vychodil in 2006.
- Fuzzy Logic Programming (FLP)
  - Generalization of the ordinary logic programming.
  - Developed and investigated by J. Medina, M. Ojeda-Aciego, P. Vojtáš, ...
- These systems are technically different and were developed to serve different purposes, but they share some common features.

Outline

Complete residuated lattice

We use complete residuated lattices as structures of truth degrees.

Definition

A *complete residuated lattice* is an algebra $\mathbf{L} = \langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ such that

- $\langle L, \wedge, \vee, 0, 1\rangle$ is a complete lattice,
- $\langle L, \otimes, 1\rangle$ is a commutative monoid,
- $\otimes$ and $\rightarrow$ satisfies $a \otimes b \leq c$ iff $a \leq b \rightarrow c$, for all $a, b, c \in L$.

Fuzzy set and subsethood

Definition

An $\mathbf{L}$-*set* A in universe U is a map $A\colon U \to L$. $A(u)$ is interpreted as "the degree to which u belongs to A".

L^U denotes the collection of all $\mathbf{L}$-sets in U.

Definition

For $\mathbf{L}$-sets $A, B \in L^U$, we define a *subsethood* degree of A in B by
$S(A, B) = \bigwedge_{u \in U} \big(A(u) \to B(u)\big)$.

In addition, we write $A \subseteq B$ iff $S(A, B) = 1$.

Truth-stressing hedge

Sometime we equip the structure of truth degrees with an unary operation, which can be seen as a truth function of a logical connective "very true".

Definition

A *truth-stressing hedge* * is an additional unary operation on L satisfying the following conditions:

- $1^* = 1$,
- $a^* \leq a$,
- $(a \rightarrow b)^* \leq a^* \rightarrow b^*$, and
- $a^{**} = a^*$ for all $a, b \in L$.

Semantics of FAIs

Definition

Let Y be a nonempty set of *attributes*. A *fuzzy attribute implication* (or shortly a FAI) is an expression $A \Rightarrow B$, where $A, B \in L^Y$.

Definition

For an $\mathbf{L}$-set $M \in L^Y$ of attributes, we define a degree $||A \Rightarrow B||_M \in L$ to which $A \Rightarrow B$ is true in M by $||A \Rightarrow B||_M = S(A, M)^* \rightarrow S(B, M)$.

Definition

Let T be an $\mathbf{L}$-set T of FAIs (theory). M is a *model* of T if $T(A \Rightarrow B) \leq ||A \Rightarrow B||_M$ for all $A, B \in L^Y$. The set of all models of T is denoted by $\mathrm{Mod}(T)$.

Definition

We define a degree $||A \Rightarrow B||_T$ to which $A \Rightarrow B$ *semantically follows from* T by $||A \Rightarrow B||_T = \bigwedge_{M \in \mathrm{Mod}(T)} ||A \Rightarrow B||_M$.

Language

Definition

Language of a fuzzy logic program (FLP) is given by

- a finite nonempty set R of *relation symbols* (predicates),
- a finite set F of *function symbols* (functors),
- *arities* of these symbols,
- a denumerable set V of *variables*,
- symbols for binary logical connectives
  - conjunctors: $\mathbin{\&}_1, \mathbin{\&}_2, \ldots$,
  - disjunctors: $\vee_1, \vee_2, \ldots$,
  - implicators: $\leftarrow_1, \leftarrow_2, \ldots$,
- and symbols for aggregations $\mathfrak{ag}_1, \mathfrak{ag}_2, \ldots$

Term and formula

Definition

For given language of FLP, *term* is defined recursively:

- Each variable $\mathbb{X} \in V$ is a term.
- If $t_1, \ldots, t_n$ are terms, then $\mathsf{f}(t_1, \ldots, t_n)$ is term for each functor $\mathsf{f} \in F$.

Definition

For given language of FLP, *formula* is defined as follows:

- If $t_1, \ldots, t_n$ are terms, then $p(t_1, \ldots, t_n)$ is an *atomic formula* for each predicate $p \in \ P$.
- If $f_1, \ldots, f_n$ are formulas, then $(f_1 \mathbin{\&}_i f_2)$, $(f_1 \vee_j f_2)$, $(f_1 \Leftarrow_k f_2)$, $\mathfrak{ag}_l(f_1, \ldots, f_n)$ are formulas.

Multi-adjoint lattice

Definition

A *complete multi-adjoint lattice* is an algebra $\langle L, \wedge, \vee, \otimes_1, \leftarrow_1, \ldots, \otimes_n, \leftarrow_n, 0, 1\rangle$, where

- $\langle L, \wedge, \vee, 0, 1\rangle$ is a complete lattice,
- $\langle L, \otimes_i, 1\rangle$ is a commutative monoid for each $i \in \{1, \ldots, n\}$,
- each adjoint pair $\langle \otimes_i, \leftarrow_i\rangle$ satisfies $a \otimes_i b \leq c$ iff $a \leq c \leftarrow_i b$ for all $a, b, c \in L$.

Fuzzy logic program

Definition

A *fuzzy logic program* for a given language with values from a given multi-adjoint lattice is a finite set P containing *rules* in the form of $\langle A \Leftarrow_i B, \vartheta \rangle$ and *facts* in the form of $\langle A, \vartheta \rangle$, where

- the *head* A is an atomic formula,
- the *tail* B is a formula without any implication
- and $\vartheta \in L \setminus \{0\}$.

Herbrand universe and base

Definition

We define *Herbrand universe* as a set of all ground terms (terms with no free occurrences of variables). Herbrand universe of FLP P will be denoted by $\mathcal{U}_P$.

Definition

Herbrand base is defined as a set of all atomic ground formulas. Herbrand base of P will be denoted by $\mathcal{B}_P$.

We usually assume that F contains at least one symbol for a *constant* $(\mathrm{ar}(\mathsf{f}) = 0)$ and R is nonempty or that R contains at least one *propositional symbol* $(\mathrm{ar}(p) = 0)$.
Due to this assumptions, $\mathcal{B}_P$ is nonempty.

Structure for FLP

- A *structure* for FLP P is any L-set in $\mathcal{B}_P$.
- If M is a structure for P, $M(\varphi)$ is interpreted as a degree to which the atomic ground formula φ is true under M.
- This notion can be extended to all formulas. First, lets define $M^\sharp$ as an $\mathbf{L}$-set of all ground formulas by
  - $M^\sharp(\varphi) = M(\varphi)$ if φ is a ground atomic formula,
  - $M^\sharp(\varphi \mathbin{\&}_i \psi) = M^\sharp(\varphi) \otimes_j M^\sharp(\psi)$ where both φ and ψ are ground and $\otimes_j$ is a truth function interpreting $\mathbin{\&}_i$,
  - analogously for the other binary connectives and aggregators.
- Then, we define $M^\sharp_\forall$ to extend the notion to all formulas by $M^\sharp_\forall(\varphi) = \bigwedge\{M^\sharp(\varphi\theta) \mid \theta \text{ is a substitution and } \varphi\theta \text{ is ground}\}$.

Model and correct answer

Definition

Structure M is called a *model* for program P if $P(\chi) \leq M^{\sharp}_{\forall}(\chi)$ for each formula χ where $P(\chi) = a$ if $\langle \chi, a \rangle \in P$ and $P(\chi) = 0$ otherwise.

The collection of all models of P will be denoted by $\mathrm{Mod}(P)$.

Definition

A pair $\langle a, \theta \rangle$ consisting of $a \in L$ and a substitution θ is a *correct answer* for a definite program P and an atomic formula φ (called a query) if $M^{\sharp}_{\forall}(\varphi\theta) \geq a$ for each $M \in \mathrm{Mod}(P)$.

Admissible rules

A computation for program P and query φ starts with $\langle \varphi, \emptyset \rangle$. Then, following rules can be used.

1. Overwrite $\langle \alpha A \beta, \Theta \rangle$ to $\langle \alpha (B\Theta' \otimes_i \vartheta) \beta, \Theta \circ \Theta' \rangle$, when
   - A is an atomic formula,
   - Θ' is the most general unifier of A and A',
   - there is a rule $\langle A' \Leftarrow_j B, \vartheta \rangle$ in P,
   - $\otimes_i$ is a multiplication which corresponds to the residuum $\leftarrow_i$ interpreting $\Leftarrow_j$.
2. Overwrite $\langle \alpha A \beta, \Theta \rangle$ to $\langle \alpha \vartheta \beta, \Theta \circ \Theta' \rangle$, when
   - A is an atomic formula,
   - Θ' is the most general unifier of A and A',
   - there is a fact $\langle A', \vartheta \rangle$.
3. Overwrite $\langle \alpha A \beta, \Theta \rangle$ to $\langle \alpha 0 \beta, \Theta \rangle$, when A is an atomic formula.
4. Compute the truth value of formula and let substitution remain the same.

Computed answer

Definition

A pair $\langle a, \Theta \rangle$, where Θ is a substitution and $a \in L$, is called a *computed answer* for query φ and program P, if there is a sequence $G_0, \ldots, G_n$ such that

- $G_0 = \langle \varphi, \emptyset \rangle$,
- each G_{i+1} we get from G_i by one of the admissible rules,
- $G_n = \langle a, \Theta' \rangle$,
- Θ is Θ' restricted to variables which occur in φ.

Soundness and completeness

Theorem (Soundness)

Each computed answer for fuzzy logic program P and query φ is a correct answer for the same program and query.

Theorem (Completeness)

For every correct answer $\langle a, \Theta \rangle$ for program P and query φ, there exist a sequence of elements $a_i \in L$ such that

- $a \leq \bigvee_i a_i$
- *and for an arbitrary i_0 there exists a computed answer $\langle b, \Theta \rangle$ such that $a_{i_0} \leq b$.*

Representing FAIs by FLPs – results

Theorem

For each finite theory T of finitely presented FAIs and finitely presented $A \Rightarrow B$ there is a definite program P such that $||A \Rightarrow B||_T \geq a$ iff for each attribute $y \in Y$ such that $a \otimes B(y) > 0$, the pair $\langle a \otimes B(y), \emptyset \rangle$ is a correct answer for the program P and y.

Corollary

For each finite theory T of finitely presented FAIs and a finitely presented $A \Rightarrow B$ there is a definite program P such that $||A \Rightarrow B||_T$ is the greatest degree $a \in L$ for which the following condition holds: for any $y \in Y$, $\langle a \otimes B(y), \emptyset \rangle$ is a correct answer for P and any $y \in Y$ provided that $a \otimes B(y) > 0$.

Sketch of the proof 1/2

- Consider a language (of FLP) without functors and with only nullary relation symbols $R = \{y_1, y_2, \ldots, y_k\}$ that correspond to attributes which appear in FAIs from T to a nonzero degree.
- Clealy, R is a finite set and Herbrand base of any program is equal to R.
- Moreover, we consider the following logical connectives:
  - implicator $\Leftarrow$ (interpreted by the residuum $\rightarrow$),
  - conjunctor $\mathbin{\&}$ (interpreted by the infimum $\wedge$),
  - a unary aggregation $\mathbb{ts}$ (interpreted by the hedge *, i.e. $M(\mathbb{ts}(\varphi)) = M(\varphi)^*$),
  - for each $a \in L \setminus \{0\}$ a unary aggregation $\mathbb{sh}_a$ called an a-shift aggregation (interpreted by $M(\mathbb{sh}_a(\varphi)) = a \rightarrow M(\varphi)$).

Sketch of the proof 2/2

- For any $C \Rightarrow D \in T$ and $y \in Y$ such that $D(y) > 0$ and all attributes $z \in Y$ satisfying $C(z) > 0$ being exactly $z_1, \ldots, z_n$, consider a rule

$$\langle y \Leftarrow \mathbb{ts}\big(\mathbb{sh}_{A(z_1)}(z_1) \mathbin{\&} \cdots \mathbin{\&} \mathbb{sh}_{A(z_n)}(z_n)\big), D(y)\rangle.$$

- The fuzzy logic program P_T generated by T consists only all these rules.

- The proof then continues by observing that $||A \Rightarrow B||_T = a > 0$ iff $||A \Rightarrow a{\otimes}B||_T = 1$ iff $||\emptyset \Rightarrow a{\otimes}B||_{T \cup \{\emptyset \Rightarrow A\}} = 1$ iff $a \otimes B(y) \leq ||\emptyset \Rightarrow \{{}^1/y\}||_{T \cup \{\emptyset \Rightarrow A\}}$ for all $y \in Y$ such that $B(y) > 0$.

- The latter is true iff for each $y \in Y$ such that $B(y) > 0$, the pair $\langle a \otimes B(y), \emptyset\rangle$ is a correct answer for the program $P_{T \cup \{\emptyset \Rightarrow A\}}$ and query y.

Example 1/3

- In the sequel, we will depict rules $\langle H \Leftarrow_i T, a\rangle$ of FLP P as $H \stackrel{a}{\Leftarrow}_i T$ and facts $\langle H, a\rangle \in P$ as $H \stackrel{a}{\Leftarrow}$.
- Let $\mathbf{L}$ be the standard Łukasievicz structure of truth degrees with * being the identity.
- Consider a set of attributes $Y = \{lA, lM, hAT, hFE, hP\}$.
- Let T being a set containing the following FAIs over Y:
  - $\{{}^{0.7}/lA, {}^{0.9}/lM, {}^{0.4}/hAT\} \Rightarrow \{{}^{0.6}/hFE, {}^{0.9}/hP\}$,
  - $\{{}^{0.8}/lA\} \Rightarrow \{{}^{0.7}/lM\}$.
- Using the presented Theorem, we can find a FLP P_T that corresponds to FAIs from T:
  - $hFE \stackrel{0.6}{\Leftarrow} \mathbb{ts}(\mathbb{sh}_{0.7}(lA) \wedge \mathbb{sh}_{0.9}(lM) \wedge \mathbb{sh}_{0.4}(hAT))$,
  - $hP \stackrel{0.9}{\Leftarrow} \mathbb{ts}(\mathbb{sh}_{0.7}(lA) \wedge \mathbb{sh}_{0.9}(lM) \wedge \mathbb{sh}_{0.4}(hAT))$,
  - $lM \stackrel{0.7}{\Leftarrow} \mathbb{ts}(\mathbb{sh}_{0.8}(lA))$.
- Obviously, the aggregator $\mathbb{ts}$ interpreted by identity can be omitted.

Example 2/3

- All aggregation interpreting $\mathbb{sh}_a(y)$ as well as the function $\wedge$ interpreting conjunctor $\mathbin{\&}$ are left-semicontinuous in this case. Thus, we can characterize $||A \Rightarrow B||_T$ using computed answers for program $P_{T\cup\{\emptyset\Rightarrow A\}}$ and queries $y \in Y$ with $B(y) > 0$.
- For example, someone can ask "How much expensive are quite new cars with automatic transmission?", i.e., more precisely "To which degree $a \in L$, is the FAI $\{{}^{0.6}/lA, {}^{1}/hAT\} \Rightarrow \{{}^{a}/hP\}$ true in T?".
- First, expand P_T to $P_{T\cup\{\emptyset\Rightarrow A\}}$ by adding facts:
  - $lA \overset{0.6}{\Leftarrow}$,
  - $hAT \overset{1}{\Leftarrow}$.

Example 3/3

- Then, we can compute an answer to query hP using the usual admissible rules of FLPs:

$$\begin{aligned}
&\langle hP, \emptyset\rangle \\
&\langle 0.9 \otimes \big(\mathbb{sh}_{0.7}(lA) \wedge \mathbb{sh}_{0.9}(lM) \wedge \mathbb{sh}_{0.4}(hAT)\big), \emptyset\rangle, \\
&\langle 0.9 \otimes \big(\mathbb{sh}_{0.7}(lA) \wedge \mathbb{sh}_{0.9}(0.7 \otimes \mathbb{sh}_{0.8}(lA)) \wedge \mathbb{sh}_{0.4}(hAT)\big), \emptyset\rangle, \\
&\langle 0.9 \otimes \big(\mathbb{sh}_{0.7}(0.6) \wedge \mathbb{sh}_{0.9}(0.7 \otimes \mathbb{sh}_{0.8}(0.6)) \wedge \mathbb{sh}_{0.4}(1)\big), \emptyset\rangle, \\
&\langle 0.9 \otimes \big(0.7 \rightarrow 0.6 \wedge 0.9 \rightarrow (0.7 \otimes (0.8 \rightarrow 0.6)) \wedge 0.4 \rightarrow 1\big), \emptyset\rangle, \\
&\langle 0.5, \emptyset\rangle.
\end{aligned}$$

- Using this computed answer $\langle 0.5, \emptyset\rangle$, we immediately get $||\{{}^{0.6}/lA, {}^{1}/hAT\} \Rightarrow \{{}^{1}/hP\}||_T = 0.5$, i.e., $||\{{}^{0.6}/lA, {}^{1}/hAT\} \Rightarrow \{{}^{0.5}/hP\}||_T = 1$.

Representing FLPs by FAIs – results

Theorem

For every definite program P *there is a set* T *of FAIs such that for each atomic formula* φ *and substitution* θ *there is a crisp* $B_\varphi \in L^{\mathcal{B}_P}$ *so that* $\langle a, \theta \rangle$ *is a correct answer for* P *and* φ *iff* $T \vdash \emptyset \Rightarrow a \otimes B_\varphi$ *and* $a > 0$.

Sketch of the proof

- Let P be a fuzzy logic program. We will construct a theory T_P of FAIs over $\mathcal{B}_P$.
- For each $A \in L^{\mathcal{B}_P}$ such that $A \neq \emptyset$, we put $A^\circ(\chi) = \bigvee\{P(\psi \Leftarrow_j \xi) \otimes_i A^\sharp(\xi\eta) \mid \xi\eta$ is ground and $\psi\eta$ equals $\chi\}$, for all $\chi \in \mathcal{B}_P$.
- The multiplication $\otimes_i$ is adjoint to the residuum $\leftarrow_i$ interpreting $\Leftarrow_j$.
- In addition, we put $\emptyset^\circ(\chi) = \bigvee\{P(\psi) \mid \psi\eta$ equals $\chi\}$ for all $\chi \in \mathcal{B}_P$.
- Put $T_P = \{A \Rightarrow A^\circ \mid A \in L^{\mathcal{B}_P}\}$.
- For the atomic formula φ and each $\psi \in \mathcal{B}_P$, we can introduce a crisp $\mathbf{L}$-set B_φ as follows:

$$B_\varphi(\psi) = \begin{cases} 1, & \text{if } \psi \text{ is a ground instance of } \varphi\theta, \\ 0, & \text{otherwise.} \end{cases}$$

- Now, we can show that $\langle a, \theta\rangle$ is a correct answer for P and the query φ iff $||\emptyset \Rightarrow B_\varphi||_{T_P} \geq a$ which is shown by proving that $\mathrm{Mod}(P) = \mathrm{Mod}(T_P)$.
- The latter is true iff $||\emptyset \Rightarrow a{\otimes}B_\varphi||_{T_P} = 1$ iff $T_P \vdash \emptyset \Rightarrow a{\otimes}B_\varphi$.

Example 1/4

- Let $L = \{0, 0.1, 0.2, \ldots, 0.9, 1\}$ be a chain of 11 elements
- equipped with Łukasiewicz $\langle \otimes_L, \leftarrow_L \rangle$ and Gödel $\langle \otimes_G, \leftarrow_G \rangle$ pair of truth functions.
- We define a language for a simple fuzzy logic program describing properties of hotels and their suitability for a sport fan.
- $R = \{near, cost, suitable\}$, $F = \{\text{hotel}, \text{center}, \text{stadium}\}$
- with $\mathrm{ar}(near) = 2$, $\mathrm{ar}(cost) = \mathrm{ar}(suitable) = 1$ and $\mathrm{ar}(\text{hotel}) = \mathrm{ar}(\text{center}) = \mathrm{ar}(\text{stadium}) = 0$.
- Using sorts of constants and variables in this example, we can specify that all constants in F are locations, but only hotel can be used for accommodation. Thus, atomic formulas $cost(\text{stadium})$, $cost(\text{center})$, $suitable(\text{stadium})$ and $suitable(\text{center})$ does not exist in our language.

$$\begin{aligned}\mathcal{B}_P = \{ & near(\text{hotel}, \text{hotel}), near(\text{hotel}, \text{center}), near(\text{hotel}, \text{stadium}), \\ & near(\text{center}, \text{hotel}), near(\text{center}, \text{center}), near(\text{center}, \text{stadium}), \\ & near(\text{stadium}, \text{hotel}), near(\text{stadium}, \text{center}), near(\text{stadium}, \text{stadium}), \\ & cost(\text{hotel}), suitable(\text{hotel})\} .\end{aligned}$$

Example 2/4

- Now, we describe a program P:

$$
\begin{aligned}
near(\text{hotel}, \text{center}) &\overset{0.8}{\Leftarrow} \\
near(\text{stadium}, \text{center}) &\overset{0.6}{\Leftarrow} \\
cost(\text{hotel}) &\overset{0.7}{\Leftarrow} \\
near(\mathbb{X}, \mathbb{X}) &\overset{1}{\Leftarrow} \\
near(\mathbb{X}, \mathbb{Y}) &\overset{1}{\Leftarrow} near(\mathbb{Y}, \mathbb{X}) \\
near(\mathbb{X}, \mathbb{Z}) &\overset{0.7}{\Leftarrow_L} near(\mathbb{X}, \mathbb{Y}) \wedge_L near(\mathbb{Y}, \mathbb{Z}) \\
suitable(\mathbb{X}) &\overset{0.8}{\Leftarrow_L} \mathbb{avg}(near(\mathbb{X}, \text{stadium}) \wedge_G near(\mathbb{X}, \text{center}), cost(\mathbb{X}))
\end{aligned}
$$

- where $\Leftarrow_L$ is interpreted by $\leftarrow_L$, $\wedge_L$ by $\otimes_L$, $\wedge_G$ by $\otimes_G$, $\mathbb{avg}$ by rounded arithmetic average and $\Leftarrow$ can be interpreted by an arbitrary residuum on L.

Example 3/4

- We construct a theory T_P of FAIs whose models are exactly models of P. First,

$$\emptyset^{\circ} = \{{}^{1}/near(\text{hotel}, \text{hotel}), {}^{0.8}/near(\text{hotel}, \text{center}), {}^{1}/near(\text{center}, \text{center}), {}^{0.6}/near(\text{stadium}, \text{center}), {}^{1}/near(\text{stadium}, \text{stadium}), {}^{0.7}/cost(\text{hotel})\}\,.$$

- Then, we can take an arbitrary $\mathbf{L}$-set A on $\mathcal{B}_P$, i.e.,

$$A = \{{}^{a_1}/near(\text{hotel}, \text{hotel}), {}^{a_2}/near(\text{hotel}, \text{center}), {}^{a_3}/near(\text{hotel}, \text{stadium}), {}^{a_4}/near(\text{center}, \text{hotel}), {}^{a_5}/near(\text{center}, \text{center}), {}^{a_6}/near(\text{center}, \text{stadium}), {}^{a_7}/near(\text{stadium}, \text{hotel}), {}^{a_8}/near(\text{stadium}, \text{center}), {}^{a_9}/near(\text{stadium}, \text{stadium}), {}^{a_{10}}/cost(\text{hotel}), {}^{a_{11}}/suitable(\text{hotel})\}\,,$$

- where $a_1, \ldots, a_{11} \in L$ are arbitrary elements such that there is some $a_i > 0$.

Example 4/4

- Then, we compute the corresponding A°.

$$A^\circ = \Big\{ {}^{b_1}\!/\mathit{near}(\mathsf{hotel}, \mathsf{hotel}), {}^{b_2}\!/\mathit{near}(\mathsf{hotel}, \mathsf{center}), {}^{b_3}\!/\mathit{near}(\mathsf{hotel}, \mathsf{stadium}), {}^{b_4}\!/\mathit{near}(\mathsf{center}, \mathsf{hotel}), {}^{b_5}\!/\mathit{near}(\mathsf{center}, \mathsf{center}), {}^{b_6}\!/\mathit{near}(\mathsf{center}, \mathsf{stadium}), {}^{b_7}\!/\mathit{near}(\mathsf{stadium}, \mathsf{hotel}), {}^{b_8}\!/\mathit{near}(\mathsf{stadium}, \mathsf{center}), {}^{b_9}\!/\mathit{near}(\mathsf{stadium}, \mathsf{stadium}), {}^{b_{10}}\!/\mathit{suitable}(\mathsf{hotel}) \Big\},$$

- where $b_1, \ldots, b_{10} \in L$ can be computed as follows:

$$b_1 = \bigvee \{a_2 \otimes_L a_4 \otimes_L 0.7, a_3 \otimes_L a_7 \otimes_L 0.7, a_1\},$$

$$b_2 = \bigvee \{a_1 \otimes_L a_2 \otimes_L 0.7, a_2 \otimes_L a_5 \otimes_L 0.7, a_3 \otimes_L a_8 \otimes_L 0.7, a_4\},$$

$$\ldots$$

$$b_{10} = \left\lfloor \frac{(a_3 \otimes_G a_2) + a_{10}}{2} \right\rfloor .$$

- The theory T_P consist of 121 ($|\mathcal{B}_P| \cdot |L|$) FAIs: $\emptyset \Rightarrow \emptyset^\circ$ and all possible $A \Rightarrow A^\circ$.

Summary

- We have studied two logic systems for reasoning with graded if-then rules:
  - the system of fuzzy logic programming (FLP) in sense of Vojtáš
  - and the system of fuzzy attribute logic (FAL) in sense of Belohlavek and Vychodil.
- We have shown that fuzzy attribute implications and fuzzy logic programs are mutually reducible and correct answers for fuzzy logic programs and queries can be described by semantic entailment of fuzzy attribute implications and *vice versa*.
- Using the link, we can transport results from FAL to FLP and *vice versa*.
- These preliminary results will be presented on The 9th International Conference on Modeling Decisions for Artificial Intelligence (MDAI 2012). A paper on these results is in preparation.

References

- R. Belohlavek, V. Vychodil: Data tables with similarity relations: Functional dependencies, complete rules and non-redundant bases. *Database Systems for Advanced Applications*, 3882, Springer, 2006.
- R. Belohlavek, V. Vychodil: Fuzzy attribute logic over complete residuated lattices. *Journal of Experimental & Theoretical Articial Intelligence*, 18(4), 2006.
- R. Belohlavek, V. Vychodil, V.: Query systems in similarity-based databases: logical foundations, expressive power, and completeness. *ACM Symposium on Applied Computing*, ACM, 2010.
- P. Vojtáš: Fuzzy logic programming. *Fuzzy Sets and Systems*, 124, 2001.
- S. Krajči, R. Lencses, J. Medina, M. Ojeda-Aciego, A, Valverde, and P. Vojtáš: Non-Commutativity and Expressive Deductive Logic Databases. *Logics in Artificial Intelligence, JELIA'02*, 2002.
- J. Medina, M. Ojeda-Aciego, and P. Vojtáš: Similarity-based unification: a multi-adjoint approach. *Fuzzy Sets and Systems*, 146(1), 2004.