

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Vytvoření výukového programu pro výuku předmětu UZI

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 15. července 2015

Josef Strolený

Abstract

This bachelor thesis deals with creation of learning materials for knowledge engineering. The thesis describes ways of knowledge representation and methods of inference and Bayesian inference.

Student should know common ways of knowledge system creation and should know about methods advantages and disadvantages after reading this thesis.

Abstrakt

Tato práce se zabývá vytvořením výukových materiálů pro výuku znalostního inženýrství. Práce popisuje způsoby reprezentace znalostí, způsoby usuzování a Bayesovský přístup.

Student by měl po přečtení práce znát základní možnosti tvorby znalostních systémů a být seznámen s jejich výhodami a nevýhodami.

Obsah

1	Úvod	1
2	Cíl práce	2
2.1	Rozsah zpracované látky	2
2.2	Zpracovaná témata	2
2.3	Podklady pro vytvoření prezentací	4
3	Způsob zpracování	5
3.1	Podoba prezentací předmětu <i>KIV / UIR</i>	5
3.2	Použité nástroje	6
3.3	Webové stránky předmětu	6
4	Získávání a reprezentace znalostí	7
4.1	Znalost	7
4.2	Způsoby reprezentace znalostí	7
4.2.1	Formální logika	8
4.2.2	Pravidlové znalostní systémy	8
4.2.3	Nepřavidlové znalostní systémy	9
4.3	Hybridní systémy	13
4.4	Získávání znalostí	13
5	Inferenční metody	15
5.1	Inference	15
5.1.1	Metody inference	15
5.1.2	Pravidla inference	17
5.2	Inference v pravidlových systémech	19
5.3	Inferenční síť	19
5.3.1	Prohledávání do hloubky	20
5.3.2	Prohledávání do šířky	20
5.3.3	Heuristické prohledávání	21
5.4	Strategie procesu usuzování	21

5.4.1	Dopředné řetězení	21
5.4.2	Zpětné řetězení	25
6	Usuzování za neurčitosti	28
6.1	Neurčitost	28
6.1.1	Příčiny neurčitosti	28
6.1.2	Vyjádření neurčitosti	28
6.2	Bayesovský přístup	28
6.2.1	Bayesovy vzorce	29
6.2.2	Pravděpodobnostní šance	30
6.2.3	Míry postačitelnosti a nezbytnosti	31
6.2.4	Vztahy mezi mírou postačitelnosti a nezbytnosti	32
6.2.5	Kombinace pravidel	32
6.2.6	Kombinace předpokladů	33
6.2.7	Výhody a nevýhody Bayesovského přístupu	33
6.3	Bayesovská síť	33
6.3.1	Definici Bayesovské sítě	34
6.3.2	Vlastnosti Bayesovské sítě	35
6.3.3	Konstrukce Bayesovské sítě	36
7	Podoba vypracovaných prezentací	37
7.1	Získávání a reprezentace znalostí	37
7.2	Inferenční metody	38
7.3	Usuzování za neurčitosti	39
7.4	Internetové stránky předmětu	41
8	Zhodnocení výsledků	42
9	Závěr	43

1 Úvod

Předmět *Úvod do znalostního inženýrství* (zkráceně *KIV / UZI*) je novým předmětem *Katedry informatiky a výpočetní techniky*. Cílem předmětu je naučit studenty základům znalostních systémů a znalostního inženýrství.

Cílem práce pak bylo pro tento předmět zpracovat prezentace pro výuku předmětu a následně je začlenit do nově vzniklých stránek předmětu. Tyto prezentace by v následujících letech měly být použity jak při přednáškách předmětu, tak i při samostudiu a během příprav na zkoušku.

Zpracována byla tři témata, zabývající se problematikou získávání znalostní a reprezentace znalostí, inferenčními metodami a nakonec způsoby usuzování za neurčitosti. Vzhledem k hodinové dotaci předmětu (2 hodiny na přednášku) jsou uvedená témata zpracována ve třech samostatných prezentacích s rozsahem asi 35 stránek na přednášku.

Přednášky, na kterých by měly být zpracované prezentace použity, budou odpřednášeny asi uprostřed semestru. V prezentacích se proto předpokládá znalost některých základních pojmů z oboru znalostních systémů.

Pro lepší pochopení probírané látky bylo vypracováno i několik příkladů. Ty by měli studentům umožnit procvičit si část probrané látky.

2 Cíl práce

Cílem práce bylo vytvoření prezentací, které by studentům předmětu *Úvod do znalostního inženýrství (UZI)* posloužily jako učební podklad a byly by použity při přednáškách tohoto předmětu. Dalším bodem bylo umístění těchto prací na webové stránky předmětu.

2.1 Rozsah zpracované látky

Pro předmět *KIV / UZI* byla stanovena hodinová dotace dvou hodin přednášek na týden. To při 13 učebních týdnech činí 26 hodin přednášek, z čehož by zpracované přednášky měly zabrat 6 hodin, tedy celkem asi 270 minut.

Přednášky by měly být odpřednášeny ve třech po sobě jdoucích týdnech, dle plánu zřejmě od 6. do 8. týdne. Rozsah prezentací byl podle vedoucího práce a garanta předmětu, prof. Matouška, upraven na přibližně 35 snímků na přednášku. Toto množství by mělo být dvouhodinové dotaci adekvátní a mělo by umožnit kvalitní odpřednášení probírané látky. Na jeden snímek tak připadá asi 2 a půl minuty času.

2.2 Zpracovaná témata

Cílem bylo zpracovat celkem tři témata. Toto je jejich hrubý výčet:

1. Získávání a reprezentace znalostí

- **reprezentace znalostí** – možnosti reprezentace znalostí v rámci znalostních systémů
- **produkční pravidla** – fungování produkčních pravidel v pravidlových znalostních systémech
- **sémantické sítě** – využití sémantické sítě pro reprezentaci znalostí
- **získávání znalostí** – problematika získávání znalostí od expertů

2. Inferenční metody

- **inferenční metody** – představení metod inference
- **rezoluční systémy** – princip rezoluce a funkce rezolučních systémů
- **dopředné a zpětné řetězení** – seznámení se strategiemi procesu usuzování
- **výběr dotazu** – způsob výběru správného dotazu
- **nemonotónní usuzování** – možnost vyjádření výjimky z pravidla

3. Usuzování za neurčitosti

- **usuzování za neurčitosti** – pojem *neurčitost*, možnosti vyjádření a zpracování neurčitosti
- **hypotetické usuzování a zpětná indukce** – představení Bayesovského přístupu
- **míry postačitelnosti a nezbytnosti** – představení míry postačitelnosti a nezbytnosti a vztahů mezi nimi
- **kombinace důkazů** – způsob kombinace důkazů a předpokladů
- **šíření pravděpodobnosti v inferenčních sítích** – šíření pravděpodobnosti a Bayesovská síť

V rámci výuky předmětu pak tato látka bezprostředně předchází přednáškám o architektuře a o tvorbě znalostních systémů a představuje tak nutný základ pro pochopení probírané problematiky.

Předcházející přednášky by naopak měly být věnovány seznámení se se základními pojmy znalostních systémů, výuce logického programovacího jazyka *Prolog* a zopakování formální logiky.

2.3 Podklady pro vytvoření prezentací

Vzhledem k obtížnosti a specifičnosti probíraného tématu bylo potřeba využít kvalitní podklady.

Základní informace o expertních a znalostních systémech jsou studentům představeny již v předmětu KIV / UIR. Z těchto informací tedy mohlo být čerpáno.

Hlavním zdrojem informací pak byly materiály dodané vedoucím práce. Jedná se o více než dvacet dokumentů, často prezentací vytvořených vedoucím práce, které se více či méně zabývají jednotlivými zpracovávanými tématy. Většina vytvořeného materiálu vychází právě z těchto zdrojů.

Zbylá témata byla vytvořena podle internetové literatury z prostředí českých i zahraničních univerzit. Tato literatura posloužila i pro doplnění některých informací, které v předešlých materiálech chyběly.

Původní jsou také příklady, které byly vytvořeny pro procvičení části probírané látky.

3 Způsob zpracování

3.1 Podoba prezentací předmětu *KIV / UIR*

Zpracované prezentace by měly mít podobu známou z předmětu *KIV / UIR*, v němž jsou také některé základní informace o znalostních systémech uvedeny.

Náhled prezentace předmětu *KIV / UIR* je k nahlédnutí na obrázku 3.1.

3. Umělý život (Artificial Life)

Smysl orientace na využití evolučních strategií a algoritmů:

Existují úlohy, které příroda zvládá velmi snadno, zatímco člověkem navržené algoritmy se hroučí.

Dva druhy evolučních algoritmů:

- algoritmy napodobující přirozené (přírodní) evoluční procesy založené na teorii přirozeného výběru a „zvládající“ problémy adaptace na měnící se podmínky a učení – **genetické algoritmy**
- algoritmy napodobující činnost mozku – **umělé neuronové sítě**, tvořené z neuronů a propojení mezi nimi, kterými se simuluje chování nervové soustavy

© Petr Heller, Václav Matoušek
Umělá inteligence a rozpoznávání, LS 2015

3-11

Obrázek 3.1: Podoba prezentace předmětu *KIV / UIR*.

Prezentace by měla v záhlaví obsahovat nadpis a v zápatí informace o autorech a o předmětu, dále informaci o aktuálním semestru a číslo prezentace s číslem stránky. Vše je laděno do modré barvy s červeným či oranžovým zvýrazněním důležitých částí.

Důvodem použití této podoby je skutečnost, že prof. Matoušek oba předměty přednáší a zavedený styl mu vyhovuje. Podobně studenti předmětu by

měli mít předmět KIV / UIR již absolvovaný a měli by být na daný styl zvyklí. Dalším důvodem je pak samozřejmě jednoduchost a funkčnost použitého stylu.

3.2 Použité nástroje

Pro vytvoření prezentací byl použit program *Microsoft Office PowerPoint 2013*. Ten je k tvorbě prezentací určen a poskytuje prostředky pro snadnou a rychlou tvorbu. Prvním krokem bylo vytvoření předlohy, která by odpovídala požadovanému stylu. Tato předloha pak mohla být použita pro vytvoření všech tří prezentací.

Velkým přínosem byl rovněž editor vzorců, který především ve třetí prezentaci, která obsahuje více matematických formulí, umožnil snadný a přehledný zápis.

Po vytvoření byly prezentace převedeny do formátu PDF virtuální tiskárnou *BullZip PDF Printer*. Použití formátu PDF umožňuje snadnou distribuci a je zajištěno, že dokument se zobrazí na všech zařízeních stejně.

3.3 Webové stránky předmětu

Předmět KIV / UZI má své webové stránky dostupné na adrese:

<http://www.kiv.zcu.cz/studies/predmety/uzi/>

Stránky jsou svým členěním totožné se stránkami předmětu KIV / UIR. První sekce obsahuje sylabus předmětu a prezentace k jednotlivým přednáškám. Ve druhé části se nachází plán cvičení a ve třetí části informace o semestrálních pracích. Následují informace o termínech zkoušek a kontakty na učitele. V poslední části jsou odkazy na externí materiály a zajímavosti.

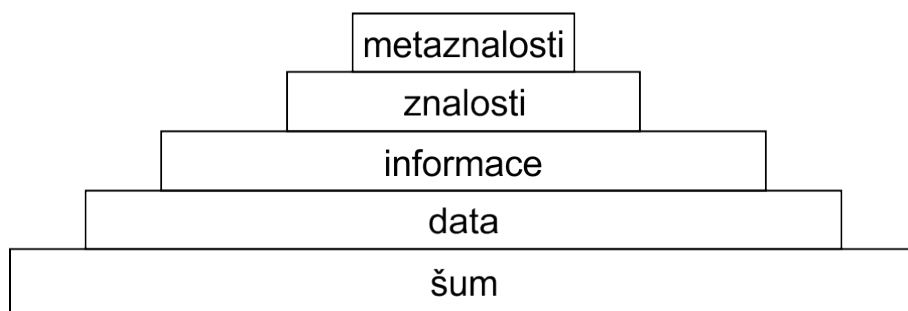
Stránky však byly vytvořeny ještě před začátkem psaní této práce minulými studenty předmětu.

4 Získávání a reprezentace znalostí

4.1 Znalost

V úvodní části první prezentace je představen stěžejní pojem *znalost*. Znalost je v počítačové terminologii považována za nejvyšší formu organizace strukturovaných dat. Každou znalost pak lze definovat pomocí *prvků dat*, *vlastnostmi* prvků dat, *relacemi* mezi prvky dat a *operacemi* nad prvky dat.

Pyramida, která ukazuje stupeň organizace dat, je uvedena na obrázku 4.1.



Obrázek 4.1: Pyramida stupně organizace dat, převzato z [MÜLLER(2002)].

Pyramida zobrazuje stupně organizace dat od stupně nejnižšího, tedy šumu, přes data a informace až ke znalostem. Výše než znalosti stojí pouze metaznalosti, což jsou znalosti o znalostech.

4.2 Způsoby reprezentace znalostí

Když víme, co znalost je, musíme ji umět reprezentovat. Existují dvě základní možnosti reprezentace:

deklarativní – co?

procedurální – jak?

Jako příklad deklarativního způsobu uveďme tvrzení "Zanesený čistič vzduchu způsobí zvýšení spotřeby pohonných hmot". Deklarativní způsob tedy říká *CO* způsobí zanesený čistič vzduchu.

V případě procedurálního způsobu bychom řekli "Má-li automobil zvýšenou spotřebu pohonných hmot, ověř, zda nemá zanesený čistič vzduchu". Procedurální způsob nám říká *JAK* máme postupovat, pokud nastanou specifické události.

Dále jsou probrány konkrétní způsoby reprezentace.

4.2.1 Formální logika

Prvním způsobem je formální (matematická) logika. Ta používá *termy* a *formule*. Termy se skládají z *proměnných* a *funkcí*. Formule jsou tvořeny z *predikátových symbolů*, *symbolů rovnosti* a *negace*, *binárních spojek* a *kvantifikátorů*.

4.2.2 Pravidlové znalostní systémy

Jako druhý způsob jsou uvedeny pravidlové (produkční) systémy [DVOŘÁK(2004)]. Těm je již věnován větší prostor, neboť se jich ve znalostních systémech hojně využívá.

Pravidlové systémy se, jak už název napovídá, skládají z pravidel. Ta se mohou vyskytovat ve více tvarech, podle konkrétního způsobu použití. Pravidla pak mohou mít například tvary:

- IF *předpoklad* THEN *závěr*
- IF *situace* THEN *akce*
- IF *podmínka* THEN *závěr* AND *akce*
- IF *podmínka* THEN *důsledek1* ELSE *důsledek2*

Předpokladová část je obvykle nazývána *antecedent* a důsledková část *konsekvent*.

Pravidlové systémy jsou schopny *nemonotónního uvažování* a *zpracování neurčitosti*. Tyto pojmy jsou blíže vysvětleny v rámci následujících prezentací.

Pro pravidlové systémy se používají dvě základní struktury:

- **Inferenční síť**

Inferenční síť je reprezentována grafem a používá se pro domény s omezeným počtem řešení. Graf je vhodný pro řešení klasifikačních a diagnostických problémů.

- **Systém porovnávání se vzory**

Systém porovnání se vzory za běhu porovnává fakta se vzory a stanovuje vztahy mezi fakty a pravidly. Díky tomu nabízí větší flexibilitu a používá se v doménách s vysokým počtem možných řešení. To může být navrhování, plánování nebo syntéza. Nevýhodou je ale horší podpora při zpracování neurčitosti.

Výhodou pravidlových systémů je přirozenost a uniformita. Nevýhodou pak nebezpečí nekonečného řetězení, neefektivnost a neprůhlednost. Hrozí také nebezpečí nekonečného řetězení a možnost přidání rozporné vlastnosti.

4.2.3 Nepravidlové znalostní systémy

Třetí možností jsou nepravidlové systémy. Zde je představeno celkem šest způsobů reprezentace.

Sémantické sítě

Prvním uvedeným nepravidlovým systémem jsou sémantické sítě [DVOŘÁK(2004)].

Ty mají podobu ohodnoceného orientovaného grafu, který se sestává z *uzlů* (ty jsou ohodnoceny termy) a *hran* (ty jsou pojmenovány druhem relací mezi uzly) [DELIYANNI(1979)].

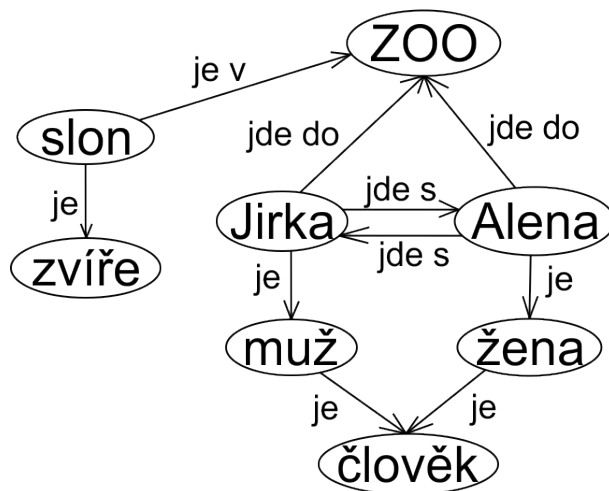
Sémantické sítě poskytují vyšší úroveň porozumění akcím, příčinám a událostem v odpovídající doméně. Díky tomu umožňují úplnější usuzování znalostního systému z této domény.

Sémantické sítě jsou výhodné pro reprezentaci fyzikálních, kauzálních a taxonomických vztahů. Podporují rovněž dědičnost a tranzitivitu. Jako příklady vztahů lze vyjmenovat například vztahy *je*, *má* nebo *je-součástí*.

V případě vztahu *je* je ovšem potřeba dbát na správnou interpretaci, neboť může nabývat více významů (je instancí, je prvkem, je podmnožinou, atd.).

Výhodou sémantických sítí je jasné vyjádření a redukce doby hledání. Nevýhodou je nebezpečí chybné interpolace a neexistence interpolačních standardů.

Jednoduchou sémantickou sít' si lze prohlédnout na obrázku 4.2.



Obrázek 4.2: Příklad sémantické sítě.

Rámce

Druhým často využívaným typem nepravidlových systémů jsou rámce [DVOŘÁK(2004)].

Jedná se o struktury pro reprezentaci stereotypních situací a stereotypních činností. Každý rámec je tvořen *jménem* a množinou *atributů*. Tyto atributy pak dále obsahují položky, jako je například *aktuální hodnota*, *implicitní hodnota* nebo *rozsah možných hodnot*. Rámce umožňují i implementaci speciálních procedur, které jsou automaticky aktivovány, pokud nastanou dané

podmínky. Těmito situacemi může být například změna (*if-changed*) nebo přidání (*if-added*) atributu.

Rámce jsou preferovaným schématem reprezentace v modelovém a případovém usuzování a využívají se v jazycích *KRYPTON*, *FRL* nebo *KSL*.

Je potřeba zmínit, že i mezi rámci může existovat vztah dědičnosti, kdy lze distribuovat informace bez nutnosti jejich zdvojení. Rámec pak může být specializací jiného (obecnějšího) rámce.

Výhodou rámci je možnost strukturované organizace znalostí, samořízení (rámce mohou určit svou aplikovatelnost v dané situaci) a možnost uchování dynamických hodnot. Nevýhodou je obtížné přizpůsobení se novým situacím a obtížný popis detailních heuristických znalostí.

Objekty

Dalším z nepravidlovým systémů jsou objekty [DVOŘÁK(2004)].

Jedná se o programové struktury, ne nepodobné tomu, co již studenti znají z *objektově orientovaného programování*. Každý objekt je instancí své třídy a obsahuje svá data a své metody, které s daty pracují. Data jsou pak přístupná pouze přes tyto metody (princip *zapouzdření*).

Známý by již měl být vztah dědičnosti, kdy lze z jedné třídy odvodit třídu jinou (ta pak může přidávat vlastní metody nebo je překrývat).

Pro znázornění je v prezentaci uvedeno několik obrázků znázorňujících vztah mezi třídou a objektem a fungování dědičnosti.

Samotná komunikace mezi objekty se může odlišovat. Při využití *časné vazby* je příjemce zprávy určen v okamžiku kompilace (tedy ještě před spuštěním programu). Při *pozdní vazbě* je příjemce určen až za běhu programu (princip *polymorfismu*).

Výhodou objektové reprezentace je možnost abstrakce, zapouzdření, dědičnosti a polymorfismu. Problémy mohou vznikat při zohlednění dosud neuvažované situace.

Petriho síť

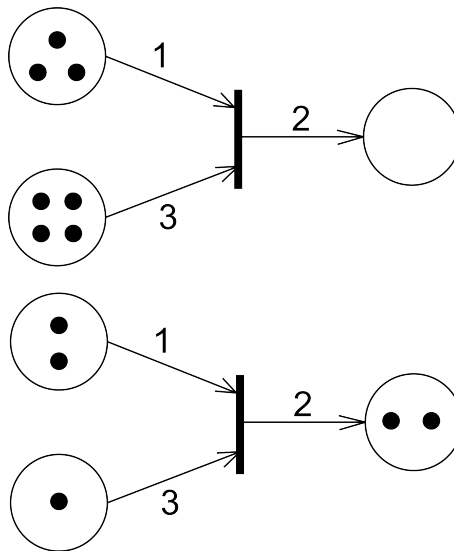
Předposledním příkladem nepravidlových systémů jsou Petriho síť.

Jedná se o orientovaný bipartitní graf s ohodnocením. Ten se skládá z uzlů, které mají podobu *míst* (*places*) nebo *přechodů* (*transitions*), a *hran* [DESEL(1995)].

Každé *místo* obsahuje libovolný počet *tokenů* (*teček*). Jejich rozložení v síti se nazývá *značení*. Hraný pak slouží ke spojování míst s přechody. Podle toho, jestli hrana vede do přechodu nebo z přechodu, rozlišujeme *vstupní místa* a *výstupní místa*. Přechod pak slouží pro *odpalování* tokenů ze vstupního místa do místa výstupního.

Výpočet v Petriho sítích je nedeterministický, což znamená, že více přechodů může být *uschopněno* současně a může pálit libovolný z nich. Na druhou stranu nemusí být odpálen žádný z nich (odpalování proběhne v čase od nuly do ∞).

Ukázka Petriho sítě je k nahlédnutí na obrázku 4.3.



Obrázek 4.3: Petriho síť před a po odpálení.

Rozhodovací stromy

Posledním příkladem nepravidlových systémů jsou rozhodovací stromy [JIROUŠEK(1995)].

Jak už z názvu vyplývá, jedná se o strom tvořený *uzly* a *hranami*. Uzly představují fáze rozhodovacího procesu a dělíme je na dva druhy. Prvním druhem je *rozhodovací uzel*, který je deterministický a závislý na uživateli. Druhým případem jsou *situační uzly*, které představují náhodné alternativy s danou pravděpodobností [KIRKWOOD(2013)].

Optimální strategie je pak určena vyhodnocením uzlů stromu od konce.

4.3 Hybridní systémy

Zatímco dřívější jazyky (první generace) byly omezeny použitím jen jednoho způsobu reprezentace znalostí, jazyky druhé generace obvykle používají hybridní reprezentaci [DVOŘÁK(2004)].

Výhodou hybridní reprezentace je možnost zkombinování pravidlově, rámcově i objektově orientovaných technik. Díky tomu je umožněn modelový přístup ke tvorbě systému.

4.4 Získávání znalostí

Poslední část prezentace je ryze teoretická a zabývá se získáváním znalostí od expertů (*Knowledge Acquisition*) a s tím spojených problémů [JONES(1989)]. Tomuto tématu jsou věnovány poslední čtyři snímky prezentace.

Získávání znalostí je klíčovou operací vývoje každého znalostního systému, která jej obvykle provází od začátku do konce. Díky tomu se jedná o nejdelsí a nejpracnější část a jeden z hlavních problémů tvorby znalostního systému.

Proces získávání znalostí je obvykle dělen na tři fáze. Při první dochází k hrubému seznámení se s problémem a získáním základních znalostí od experta. V dalších dvou fázích pak probíhá postupné zpřesňování získaných

znalostí přes znalosti obecné až po specifické.

Rozdílná je i práce s jedním a se skupinou expertů. V prvním případě obvykle dochází k použití nestrukturovaných a strukturovaných interview, myšlení nahlas nebo použití *repertoárových tabulek*. Tento způsob práce je obecně jednodušší na přípravu, ale náchylnější vůči chybně provedené expertíze. Naopak u skupiny expertů je riziko chybné expertízy nižší a probíhá většinou formou *brainstormingu*. Nevýhodou je náročnější příprava a průběh a rovněž nebezpečí konfliktu expertů.

Zajímavostí je *Paradox znalostního inženýrství*. Ten nám říká, že čím více se experti stávají kompetentními, tím méně jsou schopni popsat znalosti, které pro řešení problémů používají.

5 Inferenční metody

5.1 Inference

Inferencí rozumíme proces používaný k vyvozování nových informací z informací známých.

5.1.1 Metody inference

Existuje několik metod, jak může inference probíhat [MÜLLER(2002)].

Dedukce

Zřejmě nejznámější metodou je dedukce. Zde dochází k vyvozování neznámých závěrů ze známých předpokladů (*premis*). Tento způsob odpovídá pravidlu *modus ponens*, o němž bude řeč dále.

Jako příklad si uveďme předpoklady "Sókratés je člověk" a "Všichni lidé jsou smrtelní". Pomocí dedukce (a pravidla *modus ponens*) získáváme závěr, který zní "Sókratés je smrtelný".

Indukce

Další metodou je indukce. Zde vytváříme obecný závěr ze znalosti dílčích poznatků.

Například vidíme několik hnědých medvědů. Pomocí indukce usuzujeme, že všichni medvědi mají hnědou barvu.

Indukci můžeme dělit na dva druhy. V případě *úplné indukce* zohledníme všechny předpoklady. Příkladem tedy může být *matematická indukce*. Druhý případ se nazývá *neúplná indukce* a v potaz bereme jen část možných předpokladů. Příkladem může být anketa.

Abdukce

V případě abdukce známe pravdivé závěry a usuzujeme předpoklady.

Například máme závěr "*Pan Wágner jezdí autem*" a zároveň víme "*Každý, kdo má řidičský průkaz, jezdí autem*". Pomocí abdukce pak usuzujeme "*Pan Wágner má řidičský průkaz*".

Je potřeba si ale uvědomit, že tento náš vyvozený předpoklad nemusí být správný.

Heuristika

V případě heuristiky dochází k uplatnění pravidel *selského rozumu*. Namísto opírání se o exaktní logiku se tak opíráme spíše o *dobré mínění*.

V důsledku tak nemáme zaručeno, že nalezneme správné řešení, ale víme, že se hledání bude ubírat *správným* směrem.

Jako příklad heuristické znalosti lze uvést tvrzení "*Volný řemen větráku způsobuje hluk*".

Analogie

V případě analogie vytváříme model, který můžeme využít pro popis podobného objektu.

Jako příklad lze uvést vztah mezi tygrem a lvem. Obě šelmy spolu sdílejí společné vlastnosti, jako je například zařazení (šelma), potrava (masožravec) nebo počet nohou. Tyto vlastnosti, nadefinované jen u jednoho zvířete, pak lze pomocí analogie použít a popsat jimi i druhé zvíře.

Nemonotónní usuzování

V případě nemonotónního usuzování můžeme revidovat předcházející fakta na základě nových znalostí. Podstatou je, že předcházející fakta mohou přestat platit, dozvíme-li se nové informace.

Uvedme si příklad vyjádření výjimky z pravidla. Mějme fakta "Ptáci létají" a "Tweety je pták". Závěr tedy zní "Tweety létá".

Následně získáme nová fakta "Tučňáci jsou ptáci", "Tučňáci nelétají" a "Tweety je tučňák". S těmito novými fakty můžeme upravit náš závěr na "Tweety nelétá".

Generování a testování

V tomto případě jsou generovány nové závěry a je testována jejich správnost.

Jedna se tedy o metodu *pokus - omyl*.

Defaultní inference

Jednoduchá metoda inference, při které v případě neznalosti předpokladu použijeme defaultní hodnotu.

Intuice

Posledním a nejobtížnějším případem inference je intuice. Ta prozatím nebyla ve znalostních systémech implementována, do jisté míry se jí může blížit usuzování v rámci *neuronových sítí*.

5.1.2 Pravidla inference

Při použití inference používáme několik základních pravidel.

Modus ponens

Modus ponens, neboli *přímého usuzování*, nám říká "Jestliže platí předpoklad E a pravidlo $E \rightarrow H$, pak platí závěr H ".

Uvedme si matematický zápis $\frac{E, E \Rightarrow H}{H}$ nebo též $[E \wedge (E \Rightarrow H)] \Rightarrow H$.

Pro ilustraci pak předpokládejme následující příklad:

$E \Rightarrow H$: "Jestliže prší, je mokro"

E : "Prší"

Pomocí pravidla modus ponens pak získáme závěr H : "Je mokro"

Modus tollens

Druhou možností je použití pravidla modus tollens, tedy *zpětného usuzování*. Jestliže platí pravidlo $E \rightarrow H$ a zároveň platí $\neg H$, pak platí také $\neg E$.

Matematický zápis $\frac{\neg H, E \Rightarrow H}{\neg E}$ nebo $[(E \Rightarrow H) \wedge \neg H] \Rightarrow \neg E$.

Opět si pro ilustraci uveďme jednoduchý příklad:

$E \Rightarrow H$: "Jestliže prší, je mokro"

$\neg H$: "Není mokro"

Pak pomocí pravidla modus tollens usuzujeme $\neg E$: "Neprší"

Rezoluce

Třetí možností je použití rezoluce. Pomocí ní dokazujeme nespílitelnost formulí.

Platí zde vztah (5.1).

$$(A \vee B) \wedge (\neg B \vee C) \Rightarrow A \vee C \quad (5.1)$$

Znovu si uveďme příklad: Necht' platí $E1$; $E1 \rightarrow E2$ a $E2 \rightarrow E3$. Naším cílem je dokázat, že $E3$ je pravdivé.

1. Provedeme negaci závěru.

$$E1; E1 \Rightarrow E2; E2 \Rightarrow E3; \neg E3$$

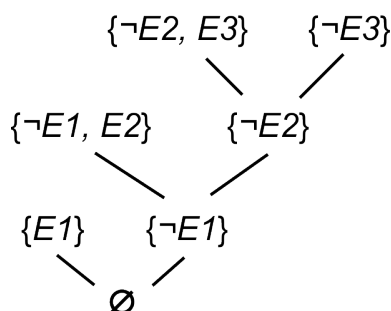
2. Nahradíme implikace disjunkcemi.

$$E1; \neg E1 \vee E2; \neg E2 \vee E3; \neg E3$$

3. Přepíšeme disjunkce do nového tvaru.

$$\{ E1 \}; \{ \neg E1, E2 \}; \{ \neg E2, E3 \}; \{ \neg E3 \}$$

4. Vytvoříme strom rezolučního vyvrácení tak, jak je zobrazen na obrázku 5.1.



Obrázek 5.1: Strom rezolučního vyvrácení.

5. Pomocí důkazu sporem bylo $E3$ dokázáno.

5.2 Inference v pravidlových systémech

V pravidlových systémech je inference založena na pravidlu modus ponens a nepřímé (zpětné) usuzování je dáno pravidlem modus tollens.

Řešení problému pak znamená nalézt řadu inferencí, které tvoří cestu od definice problému až k jeho řešení.

5.3 Inferenční síť

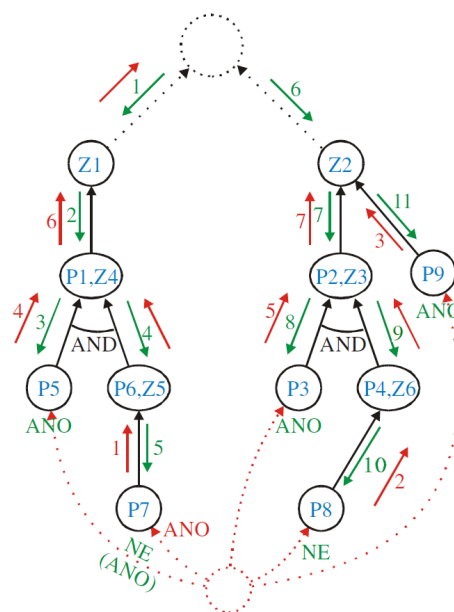
Inferenční síť bývá reprezentována jako graf, jehož uzly jsou fakty a orientované hrany jsou pravidla [MÜLLER(2002)]. Své uplatnění nalézá v doménách s omezeným počtem možných řešení, což se týká klasifikačních a diagnostických problémů. Příklad, jak může inferenční síť vypadat, je k nahlédnutí na obrázku 5.2.

Servírovat bílé či červené?

1. IF objednáno maso
THEN červené víno
2. IF objednána ryba
THEN bílé víno
3. IF mají rybu
AND máme dost peněz na rybu
THEN objednáme rybu
4. IF mají maso
AND máme dost peněz na maso
THEN objednáme maso
5. IF máme ≥ 300 Kč
THEN máme dost peněz na maso
6. IF máme ≥ 150 Kč
THEN máme dost peněz na rybu
7. IF skutečně chceme bílé víno
THEN bílé víno

P1
Z1
P2
Z2
P3
P4
Z3
P5
P6
Z4
P7
Z5
P8
Z6
P9
Z2

ZŘ ... prohledávání do hloubky
DŘ ... prohledávání do šířky



Obrázek 5.2: Příklad inferenční sítě a možnosti prohledávání, převzato z [MÜLLER(2002)].

5.3.1 Prohledávání do hloubky

V tomto případě je v inferenční síti využito klasické prohledávání do hloubky.

Výhodou tohoto řešení je interakce s uživatelem v logickém sledu a je zaručeno, že řešení bude nalezeno. Toto řešení je nejvhodnější v případě *zpětného řetězení* (bude diskutováno dále).

Nevýhodou je, že se jedná o slepé prohledávání, nevhodné pro rozsáhlé stavové prostory. Není také vhodné pro *dopředné řetězení*.

Prohledávání do hloubky je uvedeno na obrázku 5.2.

5.3.2 Prohledávání do šířky

I v tomto případě se jedná o jednoduché prohledávání, nyní do šířky.

Tento způsob zaručuje nalezení řešení. Prohledávání je rychlé v případech, kdy je výhodné použít dopředné řetězení. Lze jím také nalézt minimální cestu.

Prohledávání do šířky je slepé a tedy nevhodné pro rozsáhlé stavové prostory. Špatná je také interakce s uživatelem, kdy mohou být dotazy kladeny v nelogickém sledu. Není vhodné ho použít v případě zpětného řetězení.

Náhled prohledávání do šířky je na obrázku 5.2.

5.3.3 Heuristické prohledávání

Poslední diskutovanou možností je heuristické prohledávání.

To je obecně rychlé a není slepé, hodí se tedy pro rozsáhlejší stavové prostory. Nevýhodou je, že nemusí nalézt řešení, byť existuje.

5.4 Strategie procesu usuzování

Existují dvě strategie usuzování - *dopředné* a *zpětné* řetězení.

5.4.1 Dopředné řetězení

Nejprve proberme dopředné řetězení, čili řetězení řízené daty. V tomto případě známe data a podle nich rozhodujeme.

Základní myšlenkou je porovnávat položky z báze faktů s levými stranami pravidel z báze znalostí a následně přidávat důsledky těchto pravidel do báze faktů. To probíhá tak dlouho, dokud není odvozen cílový fakt, nebo do chvíle, kdy již nelze vykonat žádné pravidlo.

Uved'me obecný algoritmus [DVOŘÁK(2004)]:

1. **Porovnání** (matching)

Pravidla ze znalostní databáze jsou nejprve porovnána se známými fakty a zjistí se, u kterých pravidel jsou splněny předpoklady.

2. Řešení konfliktu (conflict resolution)

Z množiny pravidel se splněnými pravidly z prvního kroku se vybere pravidlo podle priority. Pokud je priorita shodná u více pravidel, provede se řešení konfliktu podle zvolené strategie.

3. Provedení (execution)

Posledním krokem je provedení vybraného pravidla. Důsledkem pak může být přidání či odstranění faktu z báze faktů, přidání pravidla do báze znalostí apod.

Obvykle je stanovena podmínka, že pravidlo může být použito pouze jednou na danou množinu faktů.

Zmínili jsme strategie řešení konfliktu. Těmito strategiemi může být:

- **Prohledávání do hloubky** (depth strategy) – v tomto případě se použije pravidlo s novějšími daty.
- **Prohledávání do šířky** (breadth strategy) – zde se naopak použije pravidlo se staršími daty.
- **Strategie složitosti / specifčnosti** (complexity strategy) – u této strategie jsou preferována složitější pravidla s více podmínkami.
- **Strategie jednoduchosti** (simplicity strategy) – u tohoto řešení jsou použita jednodušší pravidla (méně podmínek).

Příklad

Nyní si pro ilustraci uveďme příklad dopředného řetězení. Tento příklad je převzat z [ŠTÝBNAROVÁ(2002)].

Mějme bázi znalostí a faktů:

- **Báze znalostí** obsahuje následující pravidla:
 1. Jestliže (**křeček1, pohlaví, samička*) & (**křeček2, pohlaví, sameček*), pak (**křeček1, *křeček2, pár*).

2. Jestliže (**křeček, věk, *číslo*) & (**číslo > 2*), pak (**křeček, plodnýVěk, ano*).
3. Jestliže (**křeček1, *křeček2, pár*) & (**křeček1, plodnýVěk, ano*), pak (**křeček1, plodnost, ano*).
4. Jestliže (**křeček, velkéBřicho, ano*) & (**křeček, plodnost, ano*), pak (**křeček, gravidita, ano*).

• **Báze faktů** obsahuje následující fakta:

- Žulina, pohlaví, samička
- Pascie, pohlaví, sameček
- Žulina, věk, 5
- Žulina, velkéBřicho, ano

Nakonec si stanovme cílový fakt: **Žulina, gravidita, ano**

Nyní provedeme dopředné řetězení:

1. Zjištění, jaká pravidla lze použít:

- (a) Jestliže (**křeček1, pohlaví, samička*) & (**křeček2, pohlaví, sameček*), pak (**křeček1, *křeček2, pár*).
- (b) Jestliže (**křeček, věk, *číslo*) & (**číslo > 2*), pak (**křeček, plodnýVěk, ano*).

Vidíme, že nastal konflikt pravidel, který je potřeba vyřešit.

2. Řešení konfliktu pomocí jedné z možných strategií.

3. Bylo vybráno jedno z pravidel, například druhé v pořadí.

Do báze faktů nyní přibude nový fakt: **Žulina, plodnýVěk, ano**

4. Opět zjistíme, která pravidla lze použít.

- (a) Jestliže (**křeček1, pohlaví, samička*) & (**křeček2, pohlaví, sameček*), pak (**křeček1, *křeček2, pár*).
- (b) Jestliže (**křeček, věk, *číslo*) & (**číslo > 2*), pak (**křeček, plodnýVěk, ano*).

Lze použít opět stejná pravidla. Druhé pravidlo již bylo použito, použijeme tedy první.

Nyní do báze faktů přibude fakt: **Žulina, Pascie, pár**

5. Opět zjistíme, která pravidla lze použít:

- (a) Jestliže (**křeček1, pohlaví, samička*) & (**křeček2, pohlaví, sameček*), pak (**křeček1, *křeček2, pár*).
- (b) Jestliže (**křeček, věk, *číslo*) & (**číslo > 2*), pak (**křeček, plodnýVěk, ano*).
- (c) Jestliže (**křeček1, *křeček2, pár*) & (**křeček1, plodnýVěk, ano*), pak (**křeček1, plodnost, ano*).

První dvě pravidla již použita byla, použijeme tedy pravidlo třetí.

Do báze faktů přibude: **Žulina, plodnost, ano**

6. Ještě jednou zjistíme, která pravidla lze použít:

- (a) Jestliže (**křeček1, pohlaví, samička*) & (**křeček2, pohlaví, sameček*), pak (**křeček1, *křeček2, pár*).
- (b) Jestliže (**křeček, věk, *číslo*) & (**číslo > 2*), pak (**křeček, plodnýVěk, ano*).
- (c) Jestliže (**křeček1, *křeček2, pár*) & (**křeček1, plodnýVěk, ano*), pak (**křeček1, plodnost, ano*).
- (d) Jestliže (**křeček, velkéBřicho, ano*) & (**křeček, plodnost, ano*), pak (**křeček, gravidita, ano*).

Nyní lze použít všechna pravidla. První tři ale již byla použita, použijeme tedy pravidlo čtvrté.

Do báze faktů přibude fakt: **Žulina, gravidita, ano**

7. Byl odvozen cílový fakt

Odvodili jsme cílový fakt a odvozování končí.

Tímto jsme si představili dopředné řetězení.

Mezi jeho výhody patří skutečnost, že i z malého množství informací dokáže odvodit velké množství nových faktů. Je proto vhodné pro sběr informací s následným vyhodnocením a pro plánování.

Nevýhodou je, že nedokáže určit, jak důležité vstupní informace jsou a odvozuje tedy vše. Také může klást otázky v nelogickém sledu. Nevýhodné je tak jeho použití v případě, kdy chceme zodpovědět jen malé množství hypotéz, ale máme velké množství vstupních dat.

5.4.2 Zpětné řetězení

Druhu možností je zpětné řetězení, tedy řetězení řízené cíli. To zvolí možný závěr a snaží se dokázat jeho platnost. Je výhodné pro diagnostické problémy s malým počtem cílových hypotéz.

Základní myšlenkou je vybrat pravidlo podle jeho důsledkové části a zjistit, jestli je jeho předpokladová část splněna. Z předpokladové části se nyní stává *podcíl*, který má být splněn. Dochází tak k nacházení podcílů namísto faktů. Proces je ukončen ve chvíli, kdy hledané položky vložil do báze faktů sám uživatel.

Opět si uveďme obecný algoritmus [DVOŘÁK(2004)]:

1. Vytvoření zásobníku a jeho naplnění koncovými cíli.
2. Shromáždění pravidel, která mohou splnit cíl na vrcholu zásobníku. Pokud je zásobník prázdný, pak konec.
3. Postupně zkoumej všechna pravidla:
 - (a) Pokud jsou splněny předpoklady, odvod' závěr. Pokud se jednalo o koncový cíl, odstraň jej ze zásobníku a vrať se na druhý krok. Pokud šlo o podcíl, odstraň jej ze zásobníku a vrať se ke zpracování předchozího pravidla.
 - (b) Pokud fakta z báze faktů nesplňují předpoklady pravidla, ukonči zkoumání pravidla.
 - (c) Pokud pro některý parametr předpokladu chybí hodnota, zjisti, zda lze hodnotu odvodit z jiného pravidla. Pokud to lze, parametr se vloží do zásobníku jako podcíl a zkoumané pravidlo se dočasně odloží, přejde se na krok 2. Pokud to nelze, zjisti se hodnota od uživatele a pokračuje se krokem 3 a).
4. Pokud nelze pomocí žádného pravidla odvodit hodnotu důsledku, daný cíl zůstává neurčen, odstraní se ze zásobníku a pokračuje se druhým krokem.

Příklad

Pro představu si opět uveďme příklad [ŠTÝBNAROVÁ(2002)]:

Uvažujme bázi faktů a znalostí shodnou s minulým příkladem. Rovněž uvažujme i shodné cílové tvrzení: **Žulina, gravidita, ano**

Nyní probíhá zpětné řetězení:

1. Potvrzujeme tvrzení: **Žulina, plodnost, ano**

Lze použít pouze pravidlo: Jestliže (**křeček, velkéBřicho, ano*) & (**křeček, plodnost, ano*), pak (**křeček, gravidita, ano*)

Musíme tedy potvrdit platnost faktů:

- (a) Žulina, velkéBřicho, ano – již zadáno uživatelem, není potřeba dokazovat
- (b) Žulina, plodnost, ano – je potřeba dokázat

2. Potvrzujeme podcíl: **Žulina, plodnost, ano**

Nyní použijeme pravidlo: Jestliže (**křeček1, *křeček2, pár*) & (**křeček1, plodnýVěk, ano*), pak (**křeček1, plodnost, ano*)

Musíme potvrdit platnost faktů:

- (a) Žulina, *křeček2, pár – musíme dokázat
- (b) Žulina, plodnýVěk, ano – rovněž musíme dokázat

3. Potvrzujeme podcíl: **Žulina, *křeček2, pár**

Použijeme pravidlo: Jestliže (**křeček1, pohlaví, samička*) & (**křeček2, pohlaví, sameček*), pak (**křeček1, *křeček2, pár*)

Musíme tedy potvrdit platnost faktů:

- (a) Žulina, pohlaví samička – zadáno uživatelem
- (b) Pascie, pohlaví sameček – také zadáno uživatelem

4. Zbývá tedy potvrdit: **Žulina, plodnýVěk, ano**

Použijeme pravidlo: Jestliže (**křeček, věk, *číslo*) & (**číslo > 2*), pak (**křeček, plodnýVěk, ano*)

Potvrzujeme platnost faktů:

(a) Žulina, plodný Věk, >2 – i tento fakt již byl zadán uživatelem

5. Tvrzení bylo potvrzeno

Odvozování končí.

Výhodou je účinnost pro malé množství hypotéz. Otázky jsou kladeny v logickém sledu a jsou hledána jen fakta nutná pro splnění cíle. Své využití nachází zpětné řetězení v diagnostice.

Nevýhodou je slepé postupování od cíle dolů. Není tak vhodné pro řešení velkého množství hypotéz při malém počtu vstupních dat.

6 Usuzování za neurčitosti

6.1 Neurčitost

Stěžejním pojmem této kapitoly je neurčitost. Ta je charakteristickým rysem složitých systémů. Při neurčitosti nevíme, nakolik můžeme danému tvrzení či pravidlu důvěřovat.

6.1.1 Příčiny neurčitosti

Existují dvě základní příčiny neurčitosti [DVOŘÁK(2004)].

- **Problém s daty**

V tomto případě data chybějí, jsou nedostupná, nespolehlivá nebo nekonzistentní.

- **Nejisté znalosti**

Zde znalosti nemusí platit pro všechny případy nebo mohou obsahovat *vágní* pojmy.

6.1.2 Vyjádření neurčitosti

Neurčitost bývá obvykle vyjádřena *numerickými parametry* [DVOŘÁK(2004)]. Ty se označují jako *váhy*, *stupně důvěry* nebo *faktory jistoty*. Tyto parametry se pak tvrzením a pravidlům přidělují jako číslo, nebo, v novějších systémech, jako interval (dvojice hodnot).

6.2 Bayesovský přístup

Možností, jak neurčitost zpracovat, je několik. Patří sem například *Fuzzy logika*, *Dempster-Shaferova teorie*, *teorie určitosti* nebo *Bayesovský přístup*.

6.2.1 Bayesovy vzorce

Bayesovský přístup je nejstarší a nejlépe definovanou technikou [DVOŘÁK(2004)]. Znalost zde nabývá tvaru $E \rightarrow H$, kde E je předpokladem (*evidence*) a H závěrem (*hypothesis*). Toto pak čteme jako "Předpoklad E podporuje závěr H ".

Dále definujeme následující pravděpodobnosti:

- $P(E)$ – apriorní pravděpodobnost jevu E
- $P(H)$ – apriorní pravděpodobnost jevu H
- $P(H|E)$ – aposteriorní pravděpodobnost závěru (hypotézy) H
- $P(E|H)$ – podmíněná pravděpodobnost jevu E za podmínky, že nastal jev H

Nyní si můžeme představit **Bayesovy vzorce** [DVOŘÁK(2004)]:

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} \quad (6.1)$$

$$P(E) = P(E|H)P(H) + P(E|\neg H)P(\neg H) \quad (6.2)$$

$$P(H|E) = \frac{P(E|H)P(H)}{P(E|H)P(H) + P(E|\neg H)P(\neg H)} \quad (6.3)$$

Příklad

Pro ilustraci si představme pacienta s obtížemi, který je vyšetřen pomocí testu. Víme následující:

- Apriorní pravděpodobnost jevu, že pacient je nemocný, je $P(N+) = 0,1$.
- Podmíněná pravděpodobnost jevu, že test je u nemocného člověka pozitivní, je $P(T+|N+) = 0,9$.

- Podmíněná pravděpodobnost jevu, že test je u zdravého člověka negativní, je $P(T- | N-) = 0,95$.

Zajímá nás, jaká je pravděpodobnost, že člověk s pozitivním výsledkem je skutečně nemocný - $P(N+ | T+)$.

Z Bayesových vzorců (6.3) odvodíme vztah (6.4).

$$P(N+ | T+) = \frac{P(T+ | N+)P(N+)}{P(T+ | N+)P(N+) + P(T+ | N-)P(N-)} \quad (6.4)$$

Neznámé pravděpodobnosti zjistíme pomocí vztahů:

- $P(T+ | N-) = 1 - P(T- | N-) = 0,05$
- $P(N-) = 1 - P(N+) = 0,9$

Nyní tedy můžeme do vztahu dosadit hodnoty a příklad spočítat (6.5).

$$P(N+ | T+) = \frac{0,9 \cdot 0,1}{0,9 \cdot 0,1 + 0,05 \cdot 0,9} = \frac{2}{3} \quad (6.5)$$

Závěr pak zní, že dvě třetiny lidí s pozitivním výsledkem testu jsou skutečně nemocní.

6.2.2 Pravděpodobnostní šance

Šance [FARUZEL(2007)], na rozdíl od pravděpodobnosti, která nabývá hodnot z intervalu $< 0,1 >$, může nabývat hodnot z intervalu $< 0, \infty >$. Je definována jako poměr pravděpodobností, že daný jev nastane, vůči pravděpodobnosti, že nenastane. Lze ji vyjádřit vzorcem (6.6).

$$O(H) = \frac{P(H)}{P(-H)} = \frac{P(H)}{1 - P(H)} \quad (6.6)$$

Aposteriorní pravděpodobnostní šance má pak tvar(6.7).

$$O(H|E) = \frac{P(H|E)}{P(\neg H|E)} = \frac{P(H|E)}{1 - P(H|E)} \quad (6.7)$$

Pro vyjádření pravděpodobnosti ze šance pak využijeme vztahu (6.8).

$$P(H) = \frac{O(H)}{1 + O(H)} \quad (6.8)$$

6.2.3 Míry postačitelnosti a nezbytnosti

Míra postačitelnosti

Z Bayesových vzorců pro $P(H|E)$ a $P(\neg H|E)$ nyní můžeme zapsat vztah (6.9) [DVOŘÁK(2004)], [FARUZEL(2007)].

$$O(H|E) = L \cdot O(H), \text{ kde } L = \frac{O(H|E)}{O(H)} = \frac{P(E|H)}{P(E|\neg H)} \quad (6.9)$$

L (v některé literatuře také LS [MÜLLER(2002)]) se nazývá *mírou postačitelnosti*.

Říkáme "Když $E \rightarrow H$ s velkou pravděpodobností, pak L se blíží nekonečnu". To znamená, že velká hodnota L ($L \gg 1$) říká, že předpoklad E je *postačitelný* k dokázání hypotézy H .

Míra nezbytnosti

Obdobně můžeme zapsat vztah pro (6.10) [DVOŘÁK(2004)], [FARUZEL(2007)].

$$O(H|\neg E) = L^\wedge \cdot O(H), \text{ kde } L^\wedge = \frac{O(H|\neg E)}{O(H)} = \frac{P(\neg E|H)}{P(\neg E|\neg H)} \quad (6.10)$$

L^\wedge (v literatuře někdy jako LN [MÜLLER(2002)]) se nazývá *mírou nezbytnosti*.

Říkáme "Když $\neg E \rightarrow H$ s malou pravděpodobností, pak L^\wedge se blíží k 0". To znamená, že malá hodnota L^\wedge ($0 < L^\wedge \ll 1$) znamená, že E je *nezbytné* pro dokázání H .

6.2.4 Vztahy mezi mírou postačitelnosti a nezbytnosti

Vztahy, které platí mezi mírou postačitelnosti a nezbytnosti, lze nejlépe ilustrovat na tabulce 6.1 a na následujícím přehledu [MÜLLER(2002)]:

- $L > 1 \Leftrightarrow L^\wedge < 1$
- $L = 1 \Leftrightarrow L^\wedge = 1$
- $L < 1 \Leftrightarrow L^\wedge > 1$

L	L^\wedge
0 $\neg E$ je nezbytné pro H	∞ $\neg E$ je postačitelé pro H
1 E ani $\neg E$ nemá žádný vliv na H	1 E ani $\neg E$ nemá žádný vliv na H
∞ E je postačitelé pro H	0 E je nezbytné pro H

Tabulka 6.1: Tabulka vztahů mezi *mírou postačitelnosti* a *mírou nezbytnosti*.

Mezi mírou postačitelnosti a mírou nezbytnosti platí také vztah (6.11).

$$L^\wedge = \frac{1 - L \cdot P(E|\neg H)}{1 - P(E|\neg H)} \quad (6.11)$$

Toto však platí pouze **teoreticky** [MÜLLER(2002)]. V praxi expert zadává L i L^\wedge subjektivně tak, že tento vztah **splňovat nemusí**.

Pravidlo $E \rightarrow H$ pak můžeme chápat jako:

if E then H with *váha* L else H with *váha* L^\wedge

6.2.5 Kombinace pravidel

Představme si nyní, že máme více pravidel: $E_1 \rightarrow H, E_2 \rightarrow H, \dots, E_n \rightarrow H$.

Za předpokladu, že E_i jsou nezávislá, vypočteme aposteriorní pravděpodobnost podle vzorce (6.12) [DVOŘÁK(2004)].

$$O(H|E_1 \wedge \dots \wedge E_n) = L_1 \cdot \dots \cdot L_n \cdot O(H) \quad (6.12)$$

Pokud bychom neměli přesnou evidenci E_i a máme pouze pozorování E'_i , můžeme použít vztah (6.13).

$$O(H|E'_1 \wedge \dots \wedge E'_n) = L'_1 \cdot \dots \cdot L'_n \cdot O(H), \text{ kde } L'_i = \frac{O(H|E'_i)}{O(H)} \quad (6.13)$$

6.2.6 Kombinace předpokladů

Nyní předpokládejme existenci více předpokladů. Můžeme uvažovat jejich *disjunkci*, *konjunkci* nebo *negaci* [DVOŘÁK(2004)]. Uvedené vzorce vycházejí z *fuzzy* logiky.

1. **Disjunkce předpokladů** – $P(E_1 \vee E_2) = \max\{P(E_1), P(E_2)\}$
2. **Konjunkce předpokladů** – $P(E_1 \wedge E_2) = \min\{P(E_1), P(E_2)\}$
3. **Negace předpokladů** – $P(\neg E) = 1 - P(E)$

6.2.7 Výhody a nevýhody Bayesovského přístupu

Na závěr si uvedme výhody a nevýhody použití Bayesovského přístupu [DVOŘÁK(2004)].

Výhodou jsou dobré teoretické základy a dobře definovaná sémantika rozhodování.

Mezi nevýhody patří potřeba velkého množství pravděpodobnostních dat, nebezpečí neúplnosti či nekonzistence dat a obtížnost vysvětlování. Problémem je rovněž skutečnost, že uvedený předpoklad nezávislosti předpokladů E_i nebývá splněn. Hrozí také ztráta informace při popisu neurčitosti jen jedním číslem.

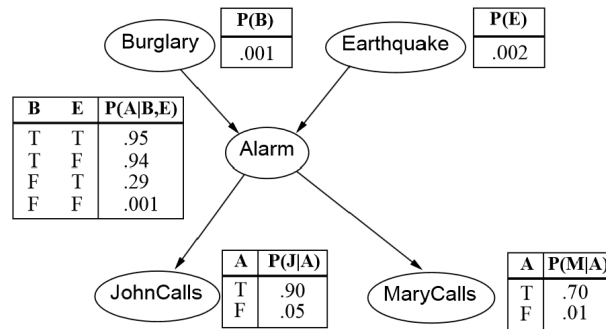
6.3 Bayesovská síť

Bayesovská síť je pravděpodobnostní model, který umožňuje reprezentovat pravděpodobnostní vztahy mezi jevy pomocí grafické reprezentace. Jedná se o orientovaný acyklický graf, který se skládá z uzlů a hran [DOMINGOS(2015)].

Uzly představují náhodné veličiny a hrany (vztahy) představují kauzální závislosti mezi proměnnými. Bayesovská síť umožňuje provádět prediktivní i diagnostické inference.

Namísto pojmu Bayesovská síť se používají rovněž pojmy *kauzální síť* nebo *influenční diagram*.

Jednoduchá Bayesovská síť je k vidění na obrázku 6.1.



Obrázek 6.1: Příklad Bayesovské sítě, převzato z [DOMINGOS(2015)].

6.3.1 Definici Bayesovské sítě

Pojmy pro definici Bayesovské sítě

Nejprve si představme pojmy potřebné k reprezentaci Bayesovské sítě.

Necht' $G = (V, E)$ je orientovaný acyklický graf a necht' $v \in V$. Nyní definujeme následující množiny:

- $C(v) = \{u \in V | (u, v) \in E\}$
- $D(v) = \{w \in V | \text{existuje cesta z } v \text{ do } w\}$
- $A(v) = \{x \in V | x \neq v \text{ a } x \notin C(v) \cup D(v)\}$

Množina $C(v)$ je množinou *bezprostředních předchůdců* uzlu v a $D(v)$ je množinou všech *následníků* uzlu v . Prvky množiny $C(v)$ pak nazýváme příčinami.

Definice Bayesovské sítě

Nyní můžeme Bayesovskou síť definovat.

- Necht' (Ω, P) je pravděpodobnostní prostor, kde $\Omega = \Omega_1 \times \dots \times \Omega_n$.
- Necht' $X_i (i = 1, \dots, n)$ je projekce Ω na Ω_i ($X_i : \Omega \rightarrow \Omega_i$ je *náhodná proměnná*).
- Necht' (V, E) je orientovaný acyklický graf, kde $V = \{X_1, \dots, X_n\}$.

Nyní můžeme říci, že (V, E, P) je Bayesovská síť, jestliže pro všechna $X_i \in V$ a všechna $W \subseteq A(X_i)$ a W je *podmíněně nezávislé* při daném $C(X_i)$.

Pokud platí $W = \{Y_1, \dots, Y_k\}$ a $C(X_i) = \{Z_1, \dots, Z_m\}$ a zároveň také platí $P(Y_1 \wedge \dots \wedge Y_k \wedge Z_1 \wedge \dots \wedge Z_m) \neq 0$, pak platí i (6.14).

$$P(X_i | Y_1 \wedge \dots \wedge Y_k \wedge Z_1 \wedge \dots \wedge Z_m) = P(X_i | Z_1 \wedge \dots \wedge Z_m) \quad (6.14)$$

Pokud známe příčiny X_i , pak pouze X_i nebo jeho *následovníci* nám mohou dát další informace o X_i .

6.3.2 Vlastnosti Bayesovské sítě

Pokud (V, E, P) je Bayesovská síť, pak platí vztah (6.15) [JIROUŠEK(1995)], [?].

$$P(V) = \prod_{\substack{X \in V \\ P(C(X)) \neq 0}} P(X | C(X)) \quad (6.15)$$

Necht' (V, E) , je orientovaný acyklický graf, kde $V = \{X_1, \dots, X_n\}$, přičemž X_i jsou proměnné s obory hodnot $O(X_i)$. Necht' $f(X | C(X))$ je nezáporná reálná funkce taková, že $\sum_{x \in O(X)} f(x | C(X)) = 1$ pro všechny kombinace hodnot proměnných z $C(X)$. Pak (6.16) definují pravděpodobnostní prostor, pro nějž (V, E, P) je Bayesovská síť. Přitom $P(X | C(X))$ je buď rovna 0 nebo $f(X | C(X))$.

$$\Omega = O(X_1) \times \dots \times O(X_n) \text{ a } P(V) = \prod_{X \in V} f(X|C(X)) \quad (6.16)$$

6.3.3 Konstrukce Bayesovské sítě

Konstrukci Bayesovské sítě lze rozdělit do tří kroků:

1. Specifikace veličin X_i, \dots, X_n a jejich oborů hodnot $O(X_i)$.
2. Konstrukce orientovaného acyklického grafu (V, E) , kde $V = \{X_1, \dots, X_n\}$, vyjadřujícího kauzální závislosti mezi veličinami.
3. Odhad pravděpodobnosti P tak, že odhadneme $P(X|C(X))$ pro všechna X , všechny hodnoty X a všechny kombinace hodnot proměnných z $C(X)$. Nezbytné je splnění pouze podmínek (6.17) a (6.18).

$$0 \leq P(X|C(X)) \leq 1 \quad (6.17)$$

$$\sum_{x \in O(X)} P(x|C(X)) = 1 \quad (6.18)$$

7 Podoba vypracovaných prezentací

7.1 Získávání a reprezentace znalostí

Tématem první prezentace je získávání a reprezentace znalostí. Prezentace se skládá ze 40 snímků, kde první tři snímky jsou věnovány názvu prezentace, seznamu probíraných částí a doporučené literatuře.

Následuje seznámení se s pojmem znalost a představení způsobů reprezentace znalostí. Poté jsou jednotlivé možnosti reprezentace postupně probrány. U každého způsobu je uvedena definice, možnosti použití a seznam výhod a nevýhod daného způsobu (viz obr. 7.1). Jednotlivé možnosti nepravidlové reprezentace pak pro ilustraci obsahují i obrázky.

6. Reprezentace znalostí v ZS

Výhody a nevýhody pravidlových systémů

- Výhody
 - Modularita
 - Uniformita
 - Přirozenost
- Možné nevýhody
 - Nebezpečí nekonečného řetězení
 - Přidání nové rozporné znalosti
 - Modifikace existujících pravidel
 - Neefektivnost
 - Neprůhlednost
 - Pokrytí domény (některé vyžadují příliš mnoho pravidel)

© Václav Matoušek, Josef Strolený
Úvod do znalostního inženýrství, ZS 2014/2015

6-14

Obrázek 7.1: Souhrn výhod a nevýhod pravidlových systémů.

Poté jsou uvedeny výhody hybridní reprezentace a následuje poslední část, která se zabývá získáváním znalostí. Této části jsou věnovány závěrečné čtyři stránky prezentace. Uveden je proces a techniky získávání znalostí od experta a možné problémy při práci s experty.

7.2 Inferenční metody

Druhá prezentace je zaměřena na inferenční metody. Prezentace má rozsah 37 snímků, kdy první tři jsou opět věnovány názvu prezentace, seznamu jednotlivých částí prezentace a doporučené literatuře.

Nejprve je uveden termín inference a jsou představeny metody inference, jako je např. dedukce či indukce (náhled na obrázku 7.2), a pravidla inference, jako např. pravidlo modus tollens.

7. Inferenční metody

Indukce

- Vytváření obecných závěrů z dílčích poznatků.
- *Vidíme tři hnědé medvědy.*
- Indukcí usuzujeme:
 - „Všichni medvědi jsou hnědí.“

Úplná indukce: zohledníme všechny premisy (matematická indukce)
Neúplná indukce: zohledněna jen část premis (anketa)

© Václav Matoušek, Josef Strolený
Úvod do znalostního inženýrství, ZS 2014/2015

7-7

Obrázek 7.2: Představení indukce.

Druhá část prezentace je věnována inferenční síti a možnostem jejího prohledávání (ukázka na obrázku 7.3). Opět jsou uvedeny hlavní výhody a nevýhody možných řešení. Nakonec jsou představeny strategie procesu usuzování, tedy usuzování řízené daty a usuzování řízené cíli.

7. Inferenční metody

Prohledávání do šířky

- Výhody
 - Zaručuje řešení
 - Rychlé v případech, kdy je výhodné použít dopředné řetězení
 - Může najít minimální cestu
- Nevýhody
 - Slepé
 - Nevhodné pro rozsáhlý stavový prostor
 - Nevhodné v případech, kdy je výhodné použít zpětné řetězení
 - Špatná interakce s uživatelem

© Václav Matoušek, Josef Strolený
Úvod do znalostního inženýrství, ZS 2014/2015 7-21

Obrázek 7.3: Prohledávání do šířky.

7.3 Usuzování za neurčitosti

Třetí a poslední prezentace je zaměřena na usuzování za neurčitosti. Prezentace má rozsah 34 snímků, první tři snímky jsou opět věnovány názvu prezentace, výčtu probíraných témat a doporučené literatuře.

Na úvod prezentace je představen pojem neurčitost a možnosti jeho vyjádření a zpracování. Prezentace se pak zaměřuje na Bayesovský přístup. Jsou tedy uvedeny Bayesovy vzorce, pravděpodobnostní šance a míry postačitelnosti a nezbytnosti (náhled na obrázku 7.4). Blíže jsou pak rozebrány vztahy mezi oběma měrami (viz obr. 7.5) a na závěr jsou uvedeny způsoby kombinace více pravidel a předpokladů.

Druhá část prezentace je věnována Baesovské síti. Je tedy uveden příklad Bayesovské sítě a její definice spolu s možností využití. Nakonec jsou řečeny vlastnosti Bayesovské sítě a způsob konstrukce.

8. Usuzování za neurčitosti

Míra postačitelosti

- Z Bayesových vzorců pro $P(H|E)$ a $P(\neg H|E)$ vyplývá:

$$O(H|E) = L \cdot O(H), \text{ kde:}$$

$$L = \frac{O(H|E)}{O(H)} = \frac{P(E|H)}{P(E|\neg H)}$$
- L (někdy též LS) se nazývá mírou postačitelosti.
 - Když $E \rightarrow H$ s velkou pravděpodobností, pak se L blíží nekonečnu, tzn. velká hodnota $L \gg 1$ říká, že předpoklad E je postačitelý k dokázání hypotézy H .

© Václav Matoušek, Josef Strolený
Úvod do znalostního inženýrství, ZS 2014/2015 8-13

Obrázek 7.4: Představení míry postačitelosti.

8. Usuzování za neurčitosti

Vztah L a L^\wedge

- Platí:
 - $L > 1 \Leftrightarrow L^\wedge < 1$
 - $L = 1 \Leftrightarrow L^\wedge = 1$
 - $L < 1 \Leftrightarrow L^\wedge > 1$
- Neboť platí:
 - $L^\wedge = \frac{1-L \cdot P(E|\neg H)}{1-P(E|\neg H)}$
- !Toto platí pouze teoreticky!**
 - Prakticky expert zadává L i L^\wedge , které vztah splňovat nemusí.

© Václav Matoušek, Josef Strolený
Úvod do znalostního inženýrství, ZS 2014/2015 8-18

Obrázek 7.5: Vztah mezi mírou postačitelosti a mírou nezbytnosti.

7.4 Internetové stránky předmětu

Vypracované prezentace jsou umístěny na stránkách předmětu, kde si je mohou studenti stáhnout. Náhled webových stránek je zobrazen na snímku 7.6.

Úvod do znalostního inženýrství
(KIV / UZI)

Organizační informace

- [Sylabus přednášek a pokyny pro cvičení](#)
- Fólie pro přednášky
 - [Literatura pro znalostní systémy](#)
 - [Literatura pro studium Prologu](#)
 - [Fólie "Programování v Prologu" Tomáše Ciglera \(2014\)](#)
 - [Fólie "Programování v Prologu" Tomáše Sládka \(2014\)](#)
 - [Fólie "Úvod do programování v Prologu"](#)
 - [Znalostní systémy - úvod](#)
 - [Metody reprezentace a zpracování znalostí v UI](#) - skriptum Prof. Radima Jirouška, VŠE Praha
 - [Reprezentace znalostí](#) - 4. kapitola učebního textu prof. Ing. Petra Berky, CSc., VŠE Praha
 - [Pravidlové znalostní systémy](#)
 - [Nepravidlové a hybridní znalostní systémy](#)
 - [Reprezentace znalostí](#)
 - [Rete algoritmus](#)
 - [Rete algorithm - úplný \(vyčerpávající\) popis algoritmu \(engl.\)](#)
 - [Systémy typu tabule \(blackboard\)](#)
 - [Zpracování neurčitosti - klasické přístupy](#)
 - [Zpracování neurčitosti - vybrané teoretické přístupy](#)
 - [Strojové učení](#)
 - [Návrh a realizace znalostních systémů](#)
 - [Obecné otázky znalostních systémů](#) (fólie Josefa Stroleného)
 - [Inferenční metody](#) (fólie Josefa Stroleného)
 - [Usuzování za neurčitosti](#) (fólie Josefa Stroleného)
 - [Reprezentace znalostí](#) (fólie Josefa Kazáka)
 - [Zpracování neurčitosti](#) (fólie Josefa Kazáka)
 - [Vybrané otázky zpracování neurčitosti ve znalostních systémech](#) (fólie Josefa Kazáka)
 - ...
- [Návod ke cvičením](#)

Zadání pro cvičení z předmětu UZI

- 1. týden - [Informace o průběhu cvičení a výběr cvičení \(zadání programů\) v Prologu](#)
- 2. týden - [Jednoduché programy v Prologu a jejich zpracování](#)
- 3. týden - [Jednoduchá úloha v Prologu - zadání faktů](#) a [zdrojový text zadání faktů](#)
- 4. týden - [Testování jednoduchých úloh v Prologu](#)
- 5. týden - [Dva "reálné" programy v Prologu - závěrečné cvičení z Prologu](#)
- 6. týden - Dokončení, předvedení a odevzdání programů z 3. - 5. cvičení
- 7. týden - [Zadání semestrálních prací](#)
- 8. týden - Konzultace zpracování semestrálních prací
- 9. týden - [Zadání tří teoretických úloh na procvičení základů znalostních systémů a reprezentace znalostí](#)

Obrázek 7.6: Internetové stránky předmětu.

8 Zhodnocení výsledků

Prvním bodem zadání bylo vytvoření webových stránek předmětu. Tyto stránky však byly vytvořeny ještě před započítím této práce, proto byl po domluvě s vedoucím práce tento bod vypuštěn.

Druhým požadavkem bylo zpracování dostupných podkladů a vytvoření prezentací pro výuku předmětu. Z celkem více než dvaceti materiálů dodaných vedoucím práce a dále z deseti internetových zdrojů z prostředí českých i zahraničních univerzit byla vytvořena trojice prezentací. V prezentacích byla zpracována všechna témata požadovaná vedoucím práce, a to v míře, která by měla vyhovovat danému rozsahu probírané látky.

Pro procvičení části z probíraných témat bylo navíc vytvořeno i několik příkladů. Ty by měly studentům umožnit procvičit si nejdůležitější probírané okruhy. Vytvořeny tak jsou příklady zabývající se vytvářením sémantických sítí, rezolučními stromy a nakonec počítáním pravděpodobností pomocí Bayesova vzorce. Příklady obsahují i správná řešení a měly by tedy posloužit jako vítaná pomůcka studentů pro procvičování. Vytvořit příklady i pro zbylá témata pak může být náplní dalších studentů předmětu.

Třetím bodem zadání bylo umístění vytvořených prezentací na webové stránky předmětu. Tento bod byl splněn a vytvořené prezentace jsou z webových stránek pro studenty dostupné.

9 Závěr

V rámci bakalářské práce byly zpracovány celkem tři prezentace, které mohou být použity pro výuku předmětu *Úvod do znalostního inženýrství (KIV/UZI)*.

Každá z prezentací zpracovává téma, které by mělo být odpřednášeno v rámci dvouhodinové přednášky. Tato témata se zabývají reprezentací znalostí, inferenčními metodami a usuzováním za neurčitosti. Při započtení titulních stránek obsahuje jedna prezentace průměrně 37 stran, což koresponduje s původním předpokladem 35 stran na prezentaci.

V prezentacích je kladen důraz zejména na úplnost, stručnost a na logickou posloupnost snímků. V rámci každého tématu je tak uvedena definice, specifické vlastnosti, možnost využití a nakonec zhodnocení a příklad nebo obrázek.

Jakým způsobem budou prezentace využity, a jaká bude jejich praktická použitelnost, se ukáže až v následujícím akademickém roce. Je tak možné, že se prezentace nevyhnou dalším úpravám.

Literatura

- [DELIYANNI(1979)] DELIYANNI, A. Semantic Networks. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.133.575&rep=rep1&type=pdf>, 1979. Online; naposledy navštíveno 6. 7. 2015.
- [DESEL(1995)] DESEL, J. Free Choice Petri Nets. <https://www7.in.tum.de/~esparza/fcbook-middle.pdf>, 1995. Online; naposledy navštíveno 6. 7. 2015.
- [DOMINGOS(2015)] DOMINGOS, P. Bayesian networks. <http://courses.cs.washington.edu/courses/cse515/09sp/slides/bnets.pdf>, 2015. Online; naposledy navštíveno 27. 4. 2015.
- [DVOŘÁK(2004)] DVOŘÁK, J. Expertní systémy. <http://www.zam.fme.vutbr.cz/~jdvorak/Opory/ExpertniSystemy.pdf>, 2004. Online; naposledy navštíveno 27. 4. 2015.
- [FARUZEL(2007)] FARUZEL, P. Webový průvodce světem expertních systémů. <http://faruzel.borec.cz/510.html>, 2007. Online; naposledy navštíveno 27. 4. 2015.
- [JIROUŠEK(1995)] JIROUŠEK, R. Metody reprezentace a zpracování znalostí v umělé inteligenci. <http://staff.utia.cas.cz/vomlel/r.pdf>, 1995. Online; naposledy navštíveno 27. 4. 2015.
- [JONES(1989)] JONES, P. H. Knowledge Acquisition. <https://engineering.purdue.edu/~engelb/abe565/knowacq.htm>, 1989. Online; naposledy navštíveno 6. 7. 2015.
- [KIRKWOOD(2013)] KIRKWOOD, C. W. Decision Trees. <http://www.public.asu.edu/~kirkwood/DASstuff/decisiontrees/DecisionTreePrimer-1.pdf>, 2013. Online; naposledy navštíveno 6. 7. 2015.

- [MÜLLER(2002)] MÜLLER, L. Znalostní systémy. https://courseware.zcu.cz/wps/PA_Courseware/DownloadDokumentu?id=46289, 2002. Online; naposledy navštíveno 27. 4. 2015.
- [ŠTÝBNAROVÁ(2002)] ŠTÝBNAROVÁ, L. Znalostní a expertní systémy. http://ui.fpf.slu.cz/diplomky/znalostni_a_expertni_systemy/def/Vyhody.html, 2002. Online; naposledy navštíveno 8. 11. 2014.