Analýza internetových vyhledávačů metodami formální konceptuální analýzy

Analysis of search engine using formal concept analysis methods

Bc. Josef Řeha

Diplomová práce
2010

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2009/2010

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Josef ŘEHA**
Osobní číslo:	**A08821**
Studijní program:	**N 3902 Inženýrská informatika**
Studijní obor:	**Počítačové a komunikační systémy**
Téma práce:	**Analýza internetových vyhledávačů metodami formální konceptuální analýzy**

Zásady pro vypracování:

1. **V teoretické části zpacujte základní pojmy a tvrzení teorie uspořádaných množin a teorie svazů.**
2. **Zejména uveďte vlastnosti uzávěrových operátorů a větu o pevném bodě v úplných svazech. Tvrzení uvádějte bez důkazů, jen s odkazem na odbornou literaturu.**
3. **V praktické části zpracujte základy formální konceptuální analýzy a formulujte základní rezeprezentační větu pomocí Galoisových konexí. Uveďte konkrétní příklad kontextu a jeho konceptuálního svazu (z dané oblasti).**
4. **Uveďte přehled používaných internetových vyhledávačů a popište základní principy jejich činnosti.**
5. **Metodami formální konceptuální analýzy proveďte rozbor používaných internetových vyhledávačů a poskytovaných služeb z hlediska vytěžování informací.**

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **HLAVENKA, J. Mistrovství ve vyhledávání na Internetu, Brno, Computer Press, 2004, ISBN 80-7226-759-0 .**
2. **ISKRA, J. Google- Vyhledávání, Gmail, Google Talk a další služby, Brno, Computer Press, a.s., 2006, ISBN 80-251-1043-5.**
3. **ISKRA, J. Google- Tipy a návody pro vyhledávač, Gmail, YouTube, Earth a další aplikace, Brno, Computer Press, a.s., 2008, ISBN 978-80-251-1833-7.**
4. **CALISHAIN, T, DORNFEST, R. 100 způsobů jak vyzrát na Google, Polsko, HELION S.A., 2004, ISBN 83-7361-565-2.**
5. **SCHRÖDER, B.S.W. Ordered Sets: an introduction,Birkhäuser Boston, 2003, 391 p., ISBN 0-8176-41289.**
6. **HARZHEIM, E. Ordered Sets, Springer, 2005, 386 p., ISBN 0-387-24219.8.**
7. **KUČERA, R. Základy teorie svazů [online]. Dostupný z WWW: http://www.math.muni.cz/ákucera/texty/Svazy2003.pdf**
8. **BĚLOHLÁVEK, R. Konceptuální svazy a formální konceptuální analýza [online]. Dostupný z WWW: http://oldwww.inf.upol.cz/belohlavek/rb_teacz.htm**

Vedoucí diplomové práce: **RNDr. Jiří Klimeš, CSc.**
Ústav matematiky

Datum zadání diplomové práce: **19. února 2010**

Termín odevzdání diplomové práce: **7. června 2010**

Ve Zlíně dne 19. února 2010

UNIVERZITA TOMÁŠE BATI VE ZLÍNĚ
Fakulta aplikované informatiky
L.S.

prof. Ing. Vladimír Vašek, CSc.
děkan

prof. Ing. Karel Vlček, CSc.
ředitel ústavu

ABSTRAKT

Diplomová práce se zabývá analýzou internetových vyhledávačů metodami formální konceptuální analýzy. V teoretické části jsou uvedeny základní poznatky teorie svazů, zejména pak uzávěrové operátory a jejich vlastnosti. V praktické části jsou zpracovány matematické základy formální konceptuální analýzy včetně její grafické interpretace. Následuje rozbor internetových vyhledávačů z hlediska jejich zaměření, principu činnosti a metod používaných při vyhledávání a řazení informací.

Klíčová slova:

Teorie svazů, formální konceptuální analýza, formální objekt, formální atribut, formální kontext, formální koncept, konceptuální svaz, internetový vyhledávač

ABSTRACT

Master thesis deals with analyzing search engines with method of formal concept analysis in information systems. In the theoretical part there are introduced the basic concepts of the lattice theory, particular closure operators and their properities. In the practical part there are presented the basic concepts and assertions of a formal concept analysis which are demonstrated by the appropriate graphical examples. It follows An analysis of Internet search engines in terms of their focus, operating principles and methods used to search and sort information.

Keywords:

Lattice theory, formal concept analysis, formal object, formal attribute, formal context, formal concept, concept lattice, information retrieval, e-mail processing, search engine

Rád bych poděkoval vedoucímu mé diplomové práce RNDr. Jiřímu Klimešovi, CSc. za jeho podmětné připomínky, rady, profesionální vedení při tvorbě diplomové práce a za odborné konzultace.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně …….

podpis diplomanta

OBSAH

ÚVOD

V diplomové práci se budu zabývat analýzou internetových vyhledávačů pomocí metod formální konceptuální analýzy. Formální konceptuální analýza (FCA) poskytuje detailně strukturovaný pohled na data. Metoda formální konceptuální analýzy vznikla v roce 1980 na univerzitě v Darmstadtu jako způsob zobrazení vztahů mezi daty a zobrazení těchto vztahů v hierarchické struktuře. Metoda spočívá v nalezení souvislostí mezi daty. FCA porovnává konkrétní i abstraktní objekty na základě jejich unikátních atributů. Vzájemná vazba množiny formálních objektů a atributů lze zaznamenat do podoby formálního kontextu prostřednictvím relace incidence a výslednou hierarchickou strukturu (hierarchické uspořádání) je možno posléze vizualizovat příslušným konceptuálním svazem tzv. Hasseovým diagramem.

Jelikož uplatnění FCA v mnoha odvětvích lidské činnosti je obrovské, rozhodl jsem se v praktické části zabývat aplikací FCA na internetové vyhledávače. Nejdříve provedu jejich základní rozdělení a poté uvedu některé jejich aplikace. Dále popíšu algoritmy, které používají jednotlivé vyhledávače k řazení relevantnosti výsledku vyhledávání. Konkrétně budu popisovat algoritmus JyxoRank, který využívá vyhledávač Jyxo a PageRank, který pro řazení výsledků používá Google. Kromě vyhledávačů, které využívají pro nezávislé ohodnocení kvality webových stránek svůj vlastní systém (algoritmus) popíšu i vyhledávače pracující na principu automatické kategorizace textových informací do výrazných, smysluplných, hierarchicky utříděných složek a kategorií. V závěru práce uvedu vyhledávače, využívající pro hledání informací klasické shlukové techniky kombinované s FCA. Výsledkem kombinací těchto dvou vyhledávací technik je konceptuální shluková technika.

I. TEORETICKÁ ČÁST

1 ZÁKLADY TEORIE SVAZŮ

Teorie svazů je oblast algebry, která se zabývá uspořádanými množinami, v nichž ke každým dvěma prvkům existuje supremum i infimum.

1.1 Polosvazy

Prvek x grupoidu $(G,\cdot)$ *se nazývá idempotentní, jestliže* $x \cdot x = x$.

Komutativní pologrupa, jejíž každý prvek je idempotentní, se nazývá polosvaz.

Podle předchozí definice tedy budeme i prázdný grupoid, který je samozřejmě komutativní i asociativní, považovat za polosvaz.

Příklad: Pro libovolnou množinu X budeme symbolem 2^X označovat množinu všech podmnožin množiny X. Pak $(2^X,\cap)$ a $(2^X,\cup)$ jsou polosvazy.

Příklad: Množina všech přirozených čísel $\mathbb{N}$ spolu s operací největší společný dělitel (resp. nejmenší společný násobek) tvoří polosvaz.

V následující větě použijeme právě učiněnou změnu definice grupoidu: grupoidem rozumíme i grupoid na prázdné množině, proto prázdná množina je podgrupoidem libovolného grupoidu. Protože existují komutativní pologrupy, v nichž žádný prvek není idempotentní (například $(\mathbb{N},+)$), museli bychom bez této změny následující větu formulovat takto:

„Nechť $(G,\cdot)$ je komutativní pologrupa. Pak množina všech idempotentních prvku, je-li neprázdná, tvoří podgrupoid pologrupy $(G,\cdot)$, který je polosvazem.“

Věta 1.1: *Nechť* $(G,\cdot)$ *je komutativní pologrupa. Pak množina všech idempotentních prvku tvoří podgrupoid pologrupy* $(G,\cdot)$*, který je polosvazem.*

Věta 1.2: *Nechť* $(G,\leq)$ *je uspořádaná množina, v níž k libovolným dvěma prvkům* $a,b \in G$ *existuje supremum* $a \vee b$. *Pak* $(G,\leq)$ *je polosvaz. Navíc pro každé* $a,b \in G$ *platí:*

$$a \leq b \Leftrightarrow a \vee b = b.$$

Věta 1.3: Nechť $(G,\cdot)$ je polosvaz. Potom relace $\leq$ daná vztahem

$$a \leq b \Leftrightarrow a \cdot b = b$$

pro každé $a,b \in G$ je uspořádání na G, ve kterém pro každé $a,b \in G$ je $a \cdot b$ supremum množiny $\{a,b\}$ v $(G,\leq)$.

Z uvedených vět vyplývá následující důsledek:

Polosvazy jsou totéž co uspořádané množiny, kde ke každým dvěma prvkům existuje supremum.

Princip duality: Nechť $(G, \leq)$ je uspořádaná množina. Definujeme-li na G novou relaci $\preccurlyeq$ takto: pro libovolné prvky $a, b \in G$ klademe

$$a \preccurlyeq b \Leftrightarrow b \leq a,$$

pak je $(G, \preccurlyeq)$ opět uspořádaná množina, přičemž supremum v $(G, \leq)$ se stane infimem v $(G, \preccurlyeq)$ a naopak.

Polosvazy jsou totéž co uspořádané množiny, kde ke každým dvěma prvkům existuje infimum [1], [2].

1.2 Svazy

Uspořádaná množina, v níž ke každým dvěma prvkům existuje supremum i infimum, se nazývá svaz.

Příklad: Každý řetězec (neboli lineárně uspořádaná množina, tj. uspořádaná množina, v níž jsou každé dva prvky srovnatelné) je svaz.

Příklad: Pro libovolnou množinu X je $(2^X, \subseteq)$ svaz.

Věta 2.1: *Nechť $(G, \leq)$ je svaz. Pro libovolné prvky $a, b \in G$ označme jejich supremum symbolem $a \vee b$ a jejich infimum symbolem $a \wedge b$. Pak $(G, \vee)$ a $(G, \wedge)$ jsou polosvazy a obě operace jsou spolu svázány tzv. absorpčními zákony: pro každé prvky $a, b \in G$ platí*

$$a \vee (b \wedge a) = a \wedge (b \vee a) = a.$$

Kromě toho pro každé prvky $a, b \in G$ platí

$$a \wedge b = a \Leftrightarrow a \leq b \Leftrightarrow a \vee b = b.$$

Věta 2.2: *Nechť $(G, \vee, \wedge)$ je množina se dvěma idempotentními, asociativními a komutativními operacemi, které jsou spolu svázány absorpčními zákony. Pak platí*

1. *pro každé prvky $a, b \in G$ platí $a \wedge b = a \Leftrightarrow a \vee b = b$,*
2. *definujeme-li na G relaci $\trianglelefteq$ takto: pro libovolné prvky $a, b \in G$ klademe*

$$a \trianglelefteq b \Leftrightarrow a \vee b = b,$$

pak je $\leq$ uspořádání na G takové, že $(G, \leq)$ je svaz, v němž pro libovolné prvky $a, b \in G$ je prvek $a \vee b$ jejich supremum a prvek $a \wedge b$ jejich infimum.

Z uvedených vět vyplývá, že svazy jsou totéž co algebraické struktury $(G,\vee,\wedge)$ se dvěma idempotentními, asociativními a komutativními operacemi, svázanými spolu absorpčními zákony. Proto i tyto struktury $(G,\vee,\wedge)$ budeme nazývat svazy.

Princip duality: Je-li $(G,\vee,\wedge)$ svaz, pak i $(G,\wedge,\vee)$ je svaz. Obecně, jestliže v nějakém platném tvrzení o svazech systematicky zaměníme supremum $\leftrightarrow$ infimum, $\vee\leftrightarrow\wedge$, $\leq\leftrightarrow\geq$ dostaneme opět platné tvrzení o svazech.

Protože není nutné zdůrazňovat, zda máme na mysli svaz jako uspořádanou množinu nebo jako algebraickou strukturu se dvěma operacemi, nebudeme v dalším textu, nebude-li to z určitého důvodu vhodné nebo dokonce nevyhnutelné, uspořádání či operace vyznačovat. Budeme tedy místo o svazu $(G, \leq)$ ci svazu $(G,\vee,\wedge)$ jednoduše psát o svazu G.

Věta 2.3: *V libovolném svazu G pro každou trojici prvku $a, b, c \in G$ platí tzv. distributivní nerovnosti*

$$(a \vee b) \wedge (a \vee c) \geq a \vee (b \wedge c),$$

$$(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c).$$

Je-li navíc $c \leq a$, platí tzv. modulární nerovnost

$$(a \wedge b) \vee c \leq a \wedge (b \vee c).$$

Věta 2.4: *Nechť G je svaz, $n \in \mathbb{N}$. Pro libovolné prvky $a_1, \ldots, a_n \in G$ platí, že $a_1 \vee \ldots \vee a_n$ je supremum množiny $\{a_1, \ldots, a_n\}$ a $a_1 \wedge \ldots \wedge a_n$ je infimum množiny $\{a_1, \ldots, a_n\}$.* [1], []

1.3 Podsvazy, ideály, filtry, homomorfismy

Nechť $(G,\vee,\wedge)$ je svaz, A podmnožina jeho nosné množiny G. Řekneme, že A je podsvaz svazu $(G,\vee,\wedge)$, jestliže je A podgrupoidem grupoidu $(G,\wedge)$ a současně podgrupoidem grupoidu $(G,\vee)$.

Je tedy $A \subseteq G$ podsvazem svazu G, právě když pro každé $a, b \in A$ platí $a \vee b \in A$ a $a \wedge b \in A$.

Příklad: Každá jednoprvková podmnožina svazu je jeho podsvazem, prázdná množina je podsvazem libovolného svazu, každý svaz je svým podsvazem.

Nechť G je svaz, $A \subseteq G$ podmnožina. Řekneme, že A je ideál svazu G, jestliže je A podsvazem svazu G, který navíc splňuje podmínku: pro každé $a \in A$ a každé $x \in G$ platí

$$x \leq a \Rightarrow x \in A.$$

Duálně, řekneme, že A je filtr svazu G, jestliže je A podsvazem svazu G, který navíc splňuje podmínku: pro každé $a \in A$ a každé $x \in G$ platí

$$x \geq a \Rightarrow x \in A.$$

Ideál svazu je tedy podsvaz, který s každým svým prvkem a obsahuje i všechny prvky svazu menší než a, filtr svazu je podsvaz, který s každým svým prvkem a obsahuje i všechny prvky svazu vetší než a.

Příklad: Každý svaz je svým ideálem i filtrem. Prázdná množina je ideálem i filtrem libovolného svazu.

Věta 3.1: *Průnik libovolného neprázdného systému podsvazů (resp. ideálů, resp. filtrů) daného svazu je opět podsvaz (resp. ideál, resp. filtr) tohoto svazu.*

Nechť G je svaz, $A \subseteq G$ podmnožina. Díky předchozí větě můžeme nyní definovat ideál $A\downarrow$ svazu G generovaný množinou A jako průnik všech ideálů tohoto svazu obsahujících množinu A. Duálně, filtr $A\uparrow$ svazu G generovaný množinou A je průnik všech filtrů tohoto svazu obsahujících množinu A. Je-li $A = \{a\}$, píšeme stručně $a\downarrow$ místo $\{a\}\downarrow$, resp. $a\uparrow$ místo $\{a\}\uparrow$, a hovoříme o hlavním ideálu, resp. o hlavním filtru, generovaném prvkem a.

Pro svaz G a podmnožinu $A \subseteq G$ je ideál $A\downarrow$ generovaný množinou A tím nejmenším (vzhledem k množinové inkluzi) ideálem svazu G ze všech ideálů obsahujících množinu A. Duálně filtr $A\uparrow$ generovaný množinou A je tím nejmenším (vzhledem k množinové inkluzi) filtrem svazu G ze všech filtrů obsahujících množinu A.

Je zřejmé, že podmnožina $A \subseteq G$ je ideálem svazu G, právě když $A\downarrow = A$, a je filtrem svazu G, právě když $A\uparrow = A$.

Věta 3.2: *Nechť G je svaz, $A \subseteq G$ podmnožina. Pro ideál $A\downarrow$ generovaný množinou A platí*

$$A\downarrow = \{x \in G; \exists n \in \mathbb{N}\ \exists a_1, \ldots, a_n \in A: x \leq a_1 \vee \ldots \vee a_n\}.$$

Duálně, pro filtr $A\uparrow$ generovaný množinou A platí

$$A\uparrow = \{x \in G; \exists n \in \mathbb{N}\ \exists a_1, \dots, a_n \in A: x \geq a_1 \wedge \dots \wedge a_n\}.$$

Nechť $(G, \leq)$, $(H, \preccurlyeq)$ *jsou uspořádané množiny* $f: G \to H$ *zobrazení. Řekneme, že je* f *izotonní zobrazení, jestliže pro každé* $a, b \in G$ *platí implikace*

$$a \leq b \Rightarrow f(a) \preccurlyeq f(b).$$

Řekneme, že f *je izomorfismus uspořádaných množin, je-li* f *bijekce a obě zobrazení* f *i* f^{-1} *jsou izotonní.*

Nechť G *a* H *jsou svazy,* $f: G \to H$ *zobrazení. Řekneme, že je* f *svazový homomorfismus, jestliže pro každé* $a, b \in G$ *platí*

$$f(a \wedge b) = f(a) \wedge f(b), \quad f(a \vee b) = f(a) \vee f(b).$$

Řekneme, že f *je svazový izomorfismus (neboli izomorfismus svazů), je-li* f *bijektivní homomorfismus.*

Protože každý svaz je také uspořádaná množina, má smysl se ptát, zda svazový homomorfismus je též izotonní zobrazení.

Věta 3.3: *Nechť* G *a* H *jsou svazy,* $f: G \to H$ *zobrazení.*

1. *Je-li* f *svazový homomorfismus, pak* f *je izotonní zobrazení a homomorfní obraz*
$$f(G) = \{f(a); a \in G\}$$
je podsvaz svazu H.
2. *Zobrazení* f *je svazový izomorfismus, právě když* f *je izomorfismus uspořádaných množin* [1],[2].

1.4 Úplné svazy

Podle věty Věta **2.4:** v libovolném svazu má každá neprázdná konečná podmnožina $\{a_1, \dots, a_n\}$ supremum $a_1 \vee \dots \vee a_n$ a infimum $a_1 \wedge \dots \wedge a_n$. Nekonečná podmnožina však supremum či infimum obecně mít nemusí.

Uspořádaná množina, v níž pro každou podmnožinu existuje supremum i infimum, se nazývá úplný svaz.

Každý úplný svaz G má nejmenší prvek (infimum množiny G ve svazu G) a nejvetší prvek (supremum množiny G ve svazu G).

Promysleme si, co znamená infimum, resp. supremum, prázdné podmnožiny svazu G. Je-li $A \subseteq G$, pak infimum množiny A ve svazu G je největší dolní závora množiny A ve svazu G. Dolní závora množiny A ve svazu G je prvek $x \in G$ takový, že pro každé $a \in A$ platí $x \leq a$. V případě $A = \emptyset$ je tato podmínka splněna pro každé $x \in G$, a tedy odtud plyne, že každý prvek svazu G je v G dolní závorou prázdné množiny. Proto infimem prázdné množiny ve svazu G je největší prvek svazu G. Duálně: supremem prázdné množiny ve svazu G je nejmenší prvek svazu G.

Příklad: Zřejmě platí, že každý úplný svaz je svazem a podle věty Věta **2.4:** je každý neprázdný konečný svaz úplným svazem.

Příklad: Prázdný svaz není úplný, neboť pro jeho (jedinou) prázdnou podmnožinu neexistuje infimum ani supremum. Jinými slovy: prázdný svaz nemá nejmenší prvek ani největší prvek, protože nemá žádný prvek.

Příklad: Pro libovolnou množinu X je $(2^X, \subseteq)$ úplný svaz.

Příklad: Pro libovolnou nekonečnou množinu X tvoří množina všech konečných podmnožin množiny X spolu s inkluzí $\subseteq$ svaz, který není úplným svazem.

Příklad: Nekonečný řetězec nemusí být úplný svaz (například $(\mathbb{N}, \leq)$ není úplný svaz, neboť neexistuje supremum celé množiny $\mathbb{N}$).

Věta 4.1: *Nechť $(G, \leq)$ je uspořádaná množina. Následující podmínky jsou ekvivalentní:*

1. *$(G, \leq)$ je úplný svaz.*
2. *$(G, \leq)$ má nejmenší prvek a každá neprázdná podmnožina množiny G má v uspořádané množině $(G, \leq)$ supremum.*
3. *$(G, \leq)$ má největší prvek a každá neprázdná podmnožina množiny G má v uspořádané množině $(G, \leq)$ infimum.*

Vzhledem k předchozí poznámce víme, že podmínku 2 lze formulovat stručněji takto: každá podmnožina množiny G má v uspořádané množině $(G, \leq)$ supremum. Analogicky pro podmínku 3: každá podmnožina množiny G má v uspořádané množině $(G, \leq)$ infimum.

Příklad: Svaz všech podgrup dané grupy G je dle předchozí věty úplný svaz, neboť má největší prvek (celou grupu G) a každá neprázdná množina podgrup má v tomto svazu infimum, kterým je průnik těchto podgrup. Rovněž svaz všech podsvazů (popřípadě svaz

ideálů nebo svaz filtrů) daného svazu je úplný svaz. Díky analogickým větám o průnicích neprázdných systému určitých podstruktur lze totéž říci i o svazu všech podokruhů daného okruhu nebo o svazu jeho ideálu, o svazu všech podtěles daného tělesa nebo o svazu všech podprostorů daného vektorového prostoru.

Příklad: $(\mathbb{N} \cup \{\infty\}, \leq)$ je dle předchozí věty úplný svaz, neboť má největší prvek ∞ a každá neprázdná podmnožina množiny $\mathbb{N} \cup \{\infty\}$ má v $(\mathbb{N} \cup \{\infty\}, \leq)$ infimum (plyne z dobré uspořádanosti).

Příklad: Ze svazu $(\mathbb{N}, |)$, který není úplný, lze doplněním nuly (která se stane jeho největším prvkem) utvořit úplný svaz $(\mathbb{N} \cup \{0\}, |)$.

Jak ukazuje následující věta, předchozí případy nebyly nijak výjimečné: vždy existuje způsob, jak doplnit svaz tak, aby se stal úplným.

Věta 4.2: *Nechť G je svaz. Pak existuje úplný svaz U, který obsahuje podsvaz H, jenž je izomorfní se svazem G.*

2 UZÁVĚROVÉ OPERÁTORY A VĚTA O PEVNÉM BODĚ

Věta 4.3 (Tarski [1], [2] **):** Nechť G je úplný svaz, $\varphi: G \to G$ je izotonní zobrazení. Pak existuje prvek $a \in G$ tak, že $\varphi(a) = a$ (tj. a je pevný bod zobrazení φ).

Tarského věta o pevném bodě má široké aplikace v computer science, zejména se používá při definicích sémantik programovacích jazyků. Dále platí, že množina pevných bodů každého izotonního zobrazení daného úplného svazu do sebe tvoří také úplný svaz. Další zajímavou vlastností Tarského věty o pevném bodě je skutečnost, že existenci pevného bodu každého izotonního zobrazení lze použít pro charakterizaci úplnosti svazů. Platí následující obrácení Tarského věty [1]:

Jestliže každé izotonní zobrazení daného svazu do sebe má pevný bod, pak je daný svaz úplný.

Odtud plyne zajímavá charakterizace úplnosti pro svazy:

Svaz G je úplný právě tehdy, když každé izotonní zobrazení svazu G do sebe má alespoň jeden pevný bod.

Definice 4.4 (Uzávěrový operátor [2] **):** Zobrazení $\varphi: G \to G$ uspořádané množiny G do sebe se nazývá uzávěrový operátor, jestliže pro každé $x, y \in G$ platí:

1. $x \leq \varphi(x)$,

2. $x \leq y \; implikuje \; \varphi(x) \leq \varphi(y)$,

3. $\varphi(\mathrm{x}) = \varphi(\varphi(x))$.

Věta 4.5 (Charakterizace uzávěrových operátorů [[2] **):** Libovolné zobrazení φ uspořádané množiny G do sebe je uzávěrový operátor právě tehdy, když pro všechna $x, y \in G$ platí ekvivalence

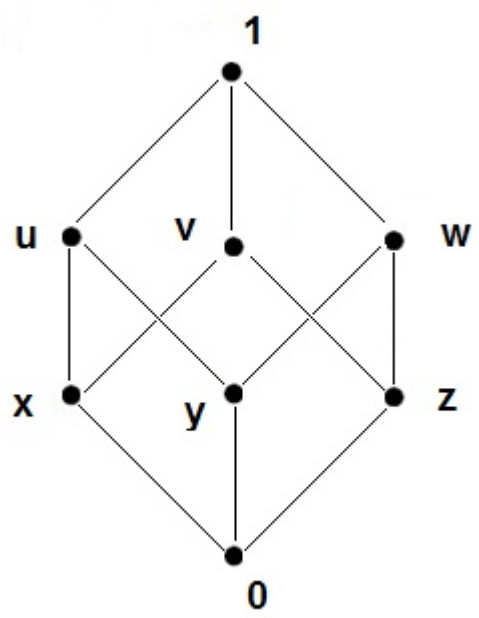
$$x \leq \varphi(y) \Leftrightarrow \varphi(x) \leq \varphi(y).$$

Teorie uzávěrových operátorů se využívá ve všech oblastech matematiky, zejména v topologii a je na ní založena definice konceptů ve formální konceptuální analýze. Pro úplné svazy platí následující věta.

Věta 4.6: Je-li φ uzávěrový operátor na úplném svazu G, pak množina všech pevných bodů $Fix(\varphi) = \{x \in G: x = \varphi(x)\}$ tvoří úplný svaz, ve kterém největší prvek je roven 1.

Na této větě je založena hlavní věta (**Věta 2 - hlavní věta o konceptuálních svazech**) formální konceptuální analýzy, která tvrdí, že množina všech konceptů každého kontextu tvoří, vzhledem k uspořádání množinovou inkluzí, úplný svaz, tzv. konceptuální svaz.

Příklad: (Uzávěrové operátory) Uvedeme konkrétní příklady uzávěrových operátorů na svazu $G = \{0, x, y, z, u, v, w, 1\}$, jehož Hasseův diagram je znázorněn na Obr.1



Obrázek 1 Hasseův diagram svazu G

Uzávěrové operátory f, g na svazu G jsou definovány předpisem v Tab. 1. Množiny pevných bodů těchto uzávěrových operátorů tvoří úplné svazy a platí pro ně:

$$Fix(f) = \{0, x, y, u, 1\},$$

$$Fix(g) = \{0, z, v, w, 1\}.$$

Dále jsou v Tab. 1 definována složená zobrazení $g \circ f$ a $f \circ g$ z těchto dvou uzávěrových operátorů. Složení $f \circ g$ je uzávěrový operátor, protože splňuje všechny tři podmínky z definice 4.4, a pro množinu pevných bodů tohoto složení platí:

$$Fix(f \circ g) = \{0,1\}.$$

Naopak složení $g \circ f$ není uzávěrový operátor na G, protože nesplňuje 3. podmínku z definice 4.4. Například pro prvek $x \in G$ platí:

$$(g \circ f)(x) = v,$$

$$(g \circ f) \circ (g \circ f)(x) = 1.$$

Odtud plyne, že složení dvou uzávěrových operátorů není obecně uzávěrový operátor,

I když množina pevných bodů tohoto složení

$$Fix(g \circ f) = \{0,1\}$$

tvoří úplný svaz.

	0	x	y	z	u	v	w	1
f	0	x	y	1	u	1	1	1
g	0	v	w	z	1	v	w	1
$g \circ f$	0	v	w	1	1	1	1	1
$f \circ g$	0	1	1	1	1	1	1	1

Tabulka 1 Definice uzávěrových operátorů f, g

2.1 Součin svazů

Podobně jako lze součinem grup $(G,\cdot)$, $(H,\cdot)$ získat grupu $(G \times H,\cdot)$ na kartézském součinu nosičů obou grup, můžeme součinem svazu získat nový svaz. Konstrukce bude naprosto stejná: operace na uspořádaných dvojicích se provedou nezávisle v každé složce.

Nechť $(G,\vee,\wedge)$, $(H,\vee,\wedge)$ *jsou svazy. Na kartézském součinu* $G \times H$ *definujme nové operace* $\vee$ *a* $\wedge$ *takto: pro každé* $g_1, g_2 \in G$, $h_1, h_2 \in H$ *klademe*

$$(g_1, h_1) \vee (g_2, h_2) = (g_1 \vee g_2,\ h_1 \vee h_2),$$

$$(g_1, h_1) \wedge (g_2, h_2) = (g_1 \wedge g_2,\ h_1 \wedge h_2).$$

Věta 5.1: *Za předpokladů učiněných v předchozí definici tvoří* $(G \times H,\vee,\wedge)$ *svaz.*

V součinu svazu platí všechny rovnosti platné v obou svazech. Vlastnosti, které se však nedají vyjádřit jako konjunkce rovností, už součin svazu zdědit nemusí. Například vlastnost být řetězec můžeme zachytit takto: pro každé dva prvky x, y platí $x \leq y$ nebo $x \geq y$, což pomocí svazových operací lze zapsat podmínkou $x \wedge y = x$ nebo $x \wedge y = y$. To ale není konjunkce rovností, ale disjunkce. A skutečně, tato vlastnost se součinem nedědí: součinem dvou dvouprvkových řetězců je čtyřprvkový svaz, který není řetězec.

Podobně jako součin dvou svazů jsme mohli definovat i součin n svazů pro libovolné $n \in \mathbb{N}$: na kartézském součinu nosných množin daných svazů se nové operace $\vee$ a $\wedge$ definují po složkách [1],[2].

2.2 Modulární svazy

Viděli jsme ve větě **Věta 2.3:** *V libovolném svazu G pro každou trojici prvku $a, b, c \in G$ platí tzv. distributivní nerovnosti*, že v libovolném svazu G pro každou trojici prvku $a, b, c \in G$ takových, že $c \leq a$, platí modulární nerovnost

$$(a \wedge b) \vee c \leq a \wedge (b \vee c).$$

Svaz G se nazývá modulární, jestliže pro každou trojici prvků $a, b, c \in G$ takových, že $c \leq a$, platí modulární rovnost

$$(a \wedge b) \vee c = a \wedge (b \vee c).$$

Příklad: Příklady modulárních svazů jsou svaz $(2^X, \cup, \cap)$ všech podmnožin nějaké množiny X nebo libovolný řetězec.

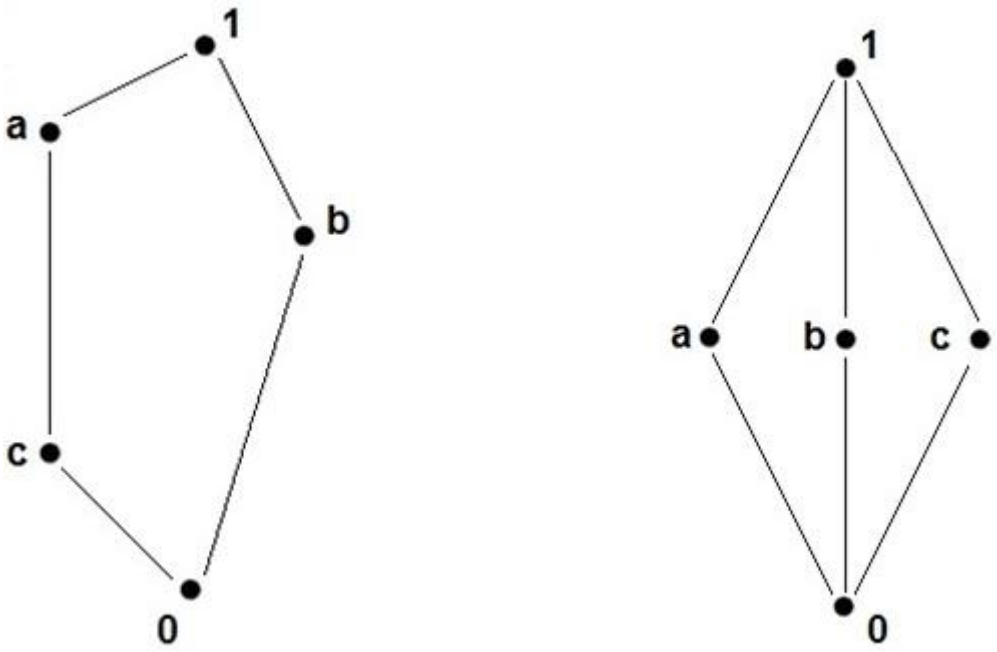
Příklad: Ukážeme, že svaz N_5, zvaný též petiúhelník, není modulární, kdežto svaz M_5, zvaný též diamant, modulární je (viz následující obrázky). Označme $0 < c < a < 1$ ony čtyři prvky, které jsou v Hasseově diagramu svazu N_5 nakresleny nad sebou vlevo, a b jeho pátý prvek. Pak nerovnost

$$(a \wedge b) \vee c = 0 \vee c = c < a = a \wedge 1 = a \wedge (b \vee c)$$

ukazuje, že svaz N_5 není modulární.

Nyní probírkou všech možností dokažme, že svaz M_5 je modulární. Označme 0 nejmenší a 1 největší prvek tohoto svazu. Nechť tedy $a, b, c \in M_5$ jsou libovolné takové, že $c \leq a$.

Jestliže $a = c$, plyne modulární rovnost z absorpčních zákonů. Jestliže $c < a$, pak na Hasseově diagramu svazu M_5 vidíme, že buď $c = 0$ nebo $a = 1$. V obou případech je modulární rovnost zřejmá.



Obrázek 2 V*levo:* Svaz N_5 Pentagon, *Vpravo:* Svaz M_5 diamant

Věta 6.1: *Svaz všech normálních podgrup dané grupy je modulární.*

Věta 6.2: *Podsvaz modulárního svazu je modulární svaz.*

Příklad: Svaz všech podprostorů daného vektorového prostoru V nad tělesem T je podle předchozí věty modulární. Je totiž podsvazem modulárního svazu všech podgrup grupy vektoru V, k tomu si stačí uvědomit, že každý podprostor je podgrupou, a ověřit, že infima i suprema se ve svazu všech podprostorů počítají stejně jako ve svazu podgrup: infimem je množinový průnik a supremem součet podprostorů.

Věta 6.3: *Svaz G je modulární, právě když pro každou trojici prvku $a, b, c \in G$ platí*

$$(a \wedge b) \vee (a \wedge c) = a \wedge \big(b \vee (a \wedge c)\big).$$

Součin modulárních svazů je modulární svaz. Homomorfní obraz modulárního svazu je modulární svaz.

Věta 6.4: *Svaz G je modulární, právě když pro každou trojici prvku $a, b, c \in G$ platí implikace*

$$a \geq c, a \wedge b = c \wedge b, a \vee b = c \vee b \Rightarrow a = c.$$

Následující věta ukazuje, že modularitu je možné charakterizovat pomocí svazu N_5 (tj. pětiúhelníku).

Věta 6.5: *Svaz G je modulární, právě když neobsahuje podsvaz izomorfní se svazem N_5.*

Duální svaz k modulárnímu svazu je opět modulární. [1], []

2.3 Distributivní svazy

Podle věty 2.3 platí: v libovolném svazu G pro každou trojici prvků $a, b, c \in G$ platí distributivní nerovnosti

$$(a \vee b) \wedge (a \vee c) \geq a \vee (b \wedge c),$$

$$(a \wedge b) \vee (a \wedge c) \leq a \wedge (b \vee c).$$

Svaz G se nazývá distributivní, jestliže pro každou trojici prvku $a, b, c \in G$ platí distributivní rovnost

$$(a \wedge b) \vee (a \wedge c) = a \wedge (b \vee c).$$

Příklad: Příklady distributivních svazů jsou svaz všech podmnožin nějaké množiny nebo libovolný řetězec.

Věta 7.1: *Nechť G je distributivní svaz. Pak pro každou trojici prvku $a, b, c \in G$ platí i následující distributivní rovnost*

$$(a \vee b) \wedge (a \vee c) = a \vee (b \wedge c).$$

Duální tvrzení k předchozí větě znamená, že z podmínky z věty plyne podmínka z definice. Je tedy lhostejné, kterou z obou distributivních rovností užijeme v definici, mohli jsme užít i obě najednou.

Duální svaz k distributivnímu svazu je opět distributivní.

Věta 7.2: *Každý distributivní svaz je modulární.*

Věta 7.3: *Podsvaz distributivního svazu je distributivní svaz.*

Věta 7.4: *Součin distributivních svazů je distributivní svaz. Homomorfní obraz distributivního svazu je distributivní svaz.*

Věta 7.5: *Svaz G je distributivní, právě když pro každou trojici prvku $a, b, c \in G$ platí implikace*

$$a \wedge b = c \wedge b, a \vee b = c \vee b \Rightarrow a = c.$$

Pro distributivní svazy platí analogie věty **6.5**: Svaz G je distributivní, právě když neobsahuje ani podsvaz izomorfní se svazem M_5 (diamant) ani podsvaz izomorfní se svazem N_5 (pentagon).

Věta 7.6: *Modulární svaz G je distributivní, právě když neobsahuje podsvaz izomorfní se svazem M_5.*

Na závěr kapitoly o distributivních svazech si uvedeme charakterizaci konečných distributivních svazů.

Prvek a svazu G se nazývá $\vee$ - nedosažitelný, jestliže pro každé $b, c \in G$ takové, že $a = b \vee c$, platí $a = b$ nebo $a = c$.

Prvek a svazu G je tedy $\vee$ - nedosažitelný, jestliže není supremem žádných dvou prvků ostře menších než on, tj. neexistují $b, c \in G$ splňující $b < a, c < a, a = b \vee c$. Ekvivalentně lze tuto podmínku vyjádřit také takto: prvek a svazu G je $\vee$ - nedosažitelný, jestliže pro každé $b, c \in G$ takové, že $b < a$ a současně $c < a$, platí $b \vee c < a$. Odtud se snadno dokáže indukcí, že takový prvek není supremem ani žádné neprázdné konečné množiny prvku ostře menších než on.

Množinu všech $\vee$ - nedosažitelných prvků svazu G označíme $J(G)$.

Věta 7.7: *V konečném distributivním svazu G je libovolný prvek a roven supremu množiny všech - nedosažitelných prvků, které neostře převyšuje, tj.*

$$a = \bigvee_{b \in J(G), b \leq a} b = \bigvee (a \downarrow \cap J(G)).$$

Nechť $(A, \leq)$ je uspořádaná množina. Množina $B \subseteq A$ se nazývá (dolů) dědičná, pokud pro každý prvek $b \in B$ a každý $a \in A$, $a \leq b$, platí $a \in B$.

Množina $B \subseteq A$ je tedy dědičná, jestliže s každým svým prvkem obsahuje všechny prvky množiny A, které jsou ještě menší. Pomocí této vlastnosti můžeme charakterizovat ideály svazu: jsou to právě dědičné podsvazy. Připomeňme, že na svazy se můžeme dívat jako na uspořádané množiny a že dva svazy jsou izomorfní, právě když jsou izomorfní jako uspořádané množiny.

Množinu všech neprázdných dědičných podmnožin uspořádané množiny A značíme $D(A)$.

Věta 7.8: *Pro konečný distributivní svaz G uvažme množinu $J(G)$ všech -nedosažitelných prvků svazu G spolu s uspořádáním, které na $J(G)$ indukuje uspořádání svazu G. Pak uspořádaná množina $\big(D\big(J(G)\big), \subseteq\big)$ je izomorfní se svazem G (chápaným jako uspořádaná množina).*

Věta mimo jiné říká, že je-li G konečný distributivní svaz, pak i $D\big(J(G)\big)$ je konečný distributivní svaz. Protože sjednocení i průnik dědičných množin je opět dědičná množina, jsou operacemi suprema a infima v $D\big(J(G)\big)$ právě množinový průnik a sjednocení. Je tedy $D\big(J(G)\big)$ podsvazem svazu všech podmnožin množiny $J(G)$.

Každý konečný distributivní svaz je izomorfní s některým podsvazem svazu všech podmnožin nějaké konečné množiny.

Podle předchozího důsledku každý konečný distributivní svaz můžeme chápat jako inkluzí uspořádaný systém množin, který je uzavřený na průniky a sjednocení. Protože naopak každý inkluzí uspořádaný systém množin, který je uzavřený na průniky a sjednocení, je zřejmě distributivním svazem, dostali jsme tak slíbenou charakterizaci konečných distributivních svazů [1],[2].

II. PRAKTICKÁ ČÁST

3 FORMÁLNÍ KONCEPTUÁLNÍ ANALÝZA

Základy formální konceptuální analýzy jsou založeny na tom, že určité objekty mají určité atributy. Základním principem je vztah mít mezi objekty a atributy: pro daný objekt a daný atribut platí, že objekt má daný atribut, nebo objekt nemá daný atribut, popř. objekt má daný atribut do jisté míry, či objekt má daný atribut s jistou hodnotou apod. Vztah *mít* mezi objekty a atributy bývá nejčastěji reprezentován tabulkou (maticí), ve které řádky odpovídají objektům, sloupce atributům a položka tabulky odpovídající objektu *x* a atributu *y* obsahuje informaci o tom, zda a popř. s jakou hodnotou má objekt *x* atribut *y*, viz Tabulka 2.

	y_1 ... y_j ... y_l
x_1	
⋮	⋮
x_i	... ... $I(x_i, y_j)$ ... ...
⋮	⋮
x_k	

Tabulka 2 Data s objekty x_i a atributy y_j

Formální konceptuální analýza (FCA) je jednou z metod analýzy tabulkových dat. Místo termínu „formální konceptuální analýza“ se také používá termín „metoda konceptuálních svazu“. Vstupem pro formální konceptuální analýzu jsou tabulková data. FCA je metodou explorativní (průzkumové) analýzy dat. Nabízí uživateli netriviální informace o vstupních datech, které mohou být využitelné přímo (jsou to nové poznatky o vstupních datech, které nejsou při pouhém pohledu na vstupní data zřejmé), popř. mohou být využitelné při dalším zpracování dat. FCA poskytuje dva základní výstupy: tzv. konceptuální svaz (což je hierarchicky uspořádaná množina jistých shluků, tzv. formálních konceptů, které jsou přítomny ve vstupní tabulce dat) a tzv. atributové implikace (které popisují jisté závislosti mezi atributy tabulky dat). V dalším textu diplomové práce předpokládáme, že atributy ve

vstupních datech jsou bivalentní (dvouhodnotové) logické atributy, tj. pro každý atribut *y* a každý uvažovaný objekt *x* platí, že *x* má *y* nebo *x* nemá *y*. Tabulka popisující takové atributy obsahuje v položce odpovídající *x* a *y* hodnotu 1 (*x* má *y*), nebo hodnotu 0 (*x* nemá *y*). Příklad takové tabulky je uveden v **Chyba! Nenalezen zdroj odkazů.**2.

Pojem lze chápat jako dvojici (A, B), kde A je množina objektů a B je množina atributů, které pod pojem patří. Ne každou dvojici (A, B) je však možné považovat za pojem. Aby tomu tak bylo, je nutné, aby A byla právě množinou všech objektů sdílejících všechny atributy z B a naopak, aby B byla právě množinou všech atributů společných všem objektům z A. Pojem ve smyslu FCA (tj. dvojici (A, B) splňující zmíněné požadavky) budeme v dalším textu nazývat koncept, popř. formální koncept. Poznamenejme, že koncepty vzájemně jednoznačně odpovídají v tabulkových datech maximálním obdélníkům vyplněným jedničkami.

Pojmy používané člověkem jsou hierarchicky uspořádány vztahem podpojem - nadpojem, daný pojem může být méně nebo více obecný než jiné pojmy. Tento vztah je v FCA modelován následovně. Řekneme, že koncept (A_1, B_1) je podpojmem konceptu (A_2, B_2) (tj. první koncept je nejvýše tak obecný jako druhý; duálně, druhý je nadpojmem prvního, popř. aspoň tak obecný jako první), pokud platí, že každý objekt z A_1 patří do A_2 nebo, což je ekvivalentní, že každý atribut z B_2 patří do B_1. Tato podmínka, kterou značíme $(A_1, B_1) \leq (A_2, B_2)$, odpovídá intuici. Vztah podpojem-nadpojem umožňuje množinu všech konceptů uspořádat podle jejich obecnosti. Takto uspořádaná množina všech konceptů se nazývá konceptuální svaz.

Atributové závislosti jsou v FCA vyjadřovány pomocí implikací tvaru *atributy* $y_1, \ldots, z_1$ *implikují atributy* $y_2, \ldots, z_2$, což se formálně zapisuje $\{y_1, \ldots, z_1\} \Rightarrow \{y_2, \ldots, z_2\}$. Význam takové implikace je ten, že každý formální koncept, který (ve svém obsahu) obsahuje $y_1, \ldots, z_1$, obsahuje i $y_2, \ldots, z_2$ (lze ukázat, že to platí, právě když každý objekt, který má všechny atributy z $y_1, \ldots, z_1$, má také všechny atributy z $y_2, \ldots, z_2$). V tomto smyslu implikace platí ve vstupních datech. Ve vstupních datech však platí velké množství implikací, řada z nich je triviálních. Proto je užitečné hledat nějakou neredundantní podmnožinu všech platných implikací, ze které popř. všechny ostatní platné implikace logicky vyplývají [1],[2],[3].

3.1 Základní pojmy a definice FCA

V této podkapitole jsou uvedeny základní teoretické pojmy a definice, které se týkají formální konceptuální analýzy. Pro podrobnější diskuzi, zdůvodnění a další informace lze odkázat na citovanou literaturu, zejména na [1].

3.1.1 Formální kontext a indukované Galoisovy konexe

(Formální) kontext je trojice $\langle X, Y, I\rangle$, kde I je binární relace mezi množinami X a Y.

Prvky množiny X, resp. Y, se nazývají objekty, resp. atributy. Fakt $\langle x, y\rangle \in I$ interpretujeme tak, že objekt x má atribut y. Formální kontext tedy reprezentuje výše zmíněná tabulková objekt-atributová data.

Každý kontext $\langle X, Y, I\rangle$ indukuje zobrazení $': 2^X \to 2^Y$ a $': 2^Y \to 2^X$ předpisem

$$A^{'} = \{y \in Y | \forall x \in A: \langle x, y\rangle \in I\}$$

pro $A \subseteq X$ a

$$B^{'} = \{x \in X | \forall y \in B: \langle x, y\rangle \in I\}$$

pro $B \subseteq Y$.

$A^{'}$ je tedy množina všech atributů společných všem objektům z A; $B^{'}$ je množina všech objektů, které sdílejí všechny atributy z B.

Zobrazení $f: 2^X \to 2^Y$ a $g: 2^Y \to 2^X$ tvoří tzv. Galoisovu konexi mezi množinami X a Y, pokud pro $A, A_1, A_2 \subseteq X$ a $B, B_1, B_2 \subseteq Y$ platí $A_1 \subseteq A_2$ implikuje $f(A_2) \subseteq f(A_1)$; $B_1 \subseteq B_2$ implikuje $g(B_2) \subseteq g(B_2)$; $A \subseteq g\big(f(A)\big)$; $B \subseteq f\big(g(B)\big)$.

Věta 1: *Pro binární relaci $I \subseteq X \times Y$ tvoří indukovaná zobrazení $'$ a $'$ Galoisovu konexi mezi X a Y. Naopak, tvoří-li f a g Galoisovu konexi mezi X a Y, existuje binární relace $I \subseteq X \times Y$ tak, že $f = '$ a $g = '$. Tím je dán vzájemně jednoznačný vztah mezi Galoisovými konexemi mezi X a Y a binárními relacemi mezi X a Y* [1],[3].

3.1.2 Formální koncepty a konceptuální svazy

(Formální) koncept v kontextu $\langle X, Y, I\rangle$ je dvojice (A, B), kde $A \subseteq X$ a $B \subseteq Y$ jsou takové, že $A' = B$ a $B' = A$.

Formální koncept je tedy dvojice sestávající z množiny A objektů i množiny B atributů takových, že B jsou právě všechny atributy společné objektům z A a A jsou právě všechny

objekty sdílející atributy z B. Z matematického pohledu je koncept právě pevným bodem Galoisovy konexe dané dvojicí zobrazení $f = '$ a $g = '$.

Množinu všech formálních konceptů v $\langle X, Y, I\rangle$ značíme $\mathcal{B}(X, Y, I)$, tj.

$$\mathcal{B}(X, Y, I) = \{(A, B) | A \subseteq X, B \subseteq Y, A^{'} = B, B^{'} = A\}.$$

Konceptuální svaz je množina $\mathcal{B}(X, Y, I)$ *spolu s relací* $\leq$ *definovanou na* $\mathcal{B}(X, Y, I)$ *předpisem* $(A_1, B_1) \leq (A_2, B_2)$ *právě když* $A_1 \subseteq A_2$ *(nebo ekvivalentně,* $B_2 \subseteq B_1$*).*

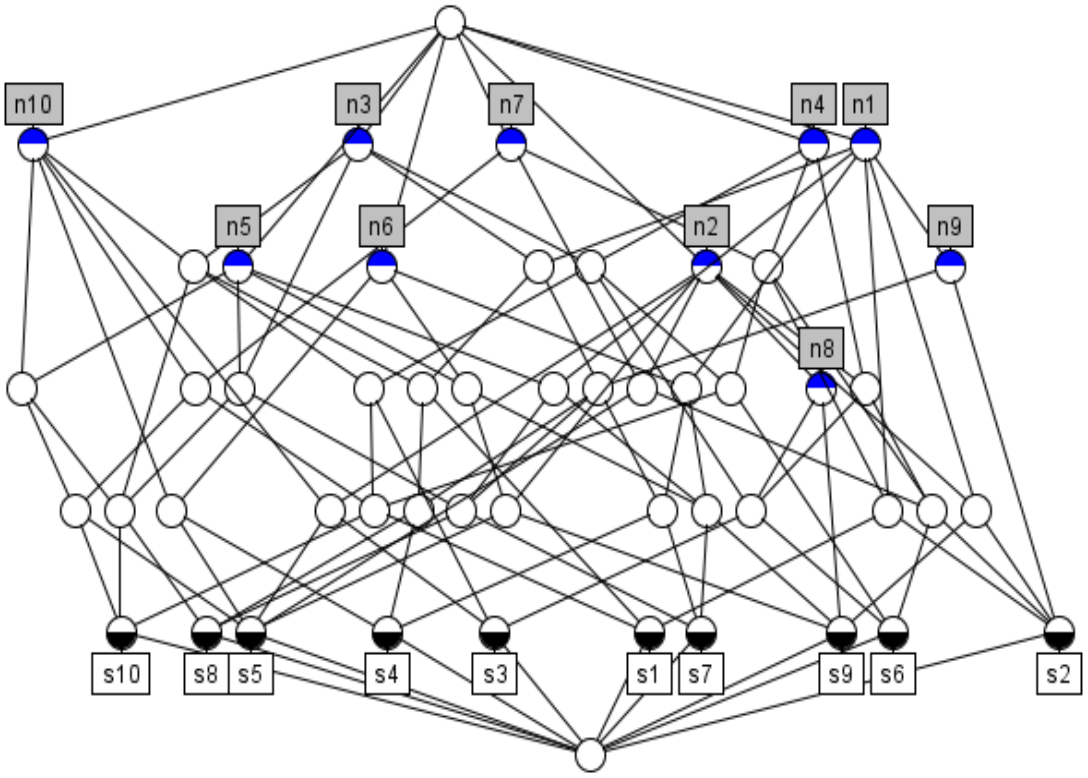
Pro další účely označíme $Int(I) = \{B \subseteq Y | \langle A, B\rangle \in \mathcal{B}(X, Y, I)\}$ pro nějakou $A \subseteq X$, tj. $Int(I)$ je množina obsahů všech konceptů z $\mathcal{B}(X, Y, I)$. Platí, že $B \subseteq Y$ je obsahem nějakého konceptu z $\mathcal{B}(X, Y, I)$. Podobně značíme $Ext(I)$ rozsahy konceptů z $\mathcal{B}(X, Y, I)$.

Relace $\leq$ je tedy relací podpojem-nadpojem. Následující věta, tzv. hlavní věta o konceptuálních svazech, popisuje strukturu $\mathcal{B}(X, Y, I)$. Mimo jiné zdůvodňuje název konceptuální svaz.

Příklad: V následujícím je uveden konkrétní příklad kontextu $\langle X, Y, I\rangle$, kde množina objektů X je tvořena deseti prvky *{s1, s2, s3, s4, s5, s6, s7, s8, s9, s10}*, množina atributů Y je tvořena také deseti prvky *{n1, n2, n3, n4, n5, n6, n7, n8, n9, n10}*, binární relace I je znázorněna v Tab. 3

	n1	n2	n3	n4	n5	n6	n7	n8	n9	n10
s1	X		X	X			X			X
s2	X	X		X			X		X	
s3		X	X	X				X		X
s4	X		X			X			X	X
s5		X			X	X	X			X
s6			X	X	X			X	X	
s7	X		X		X	X			X	
s8	X		X		X				X	X
s9	X	X			X	X		X		
s10			X	X	X		X			X

Tabulka 3 Binární relace mezi objekty X a atributy Y



Obrázek 3 Konceptuální svaz kontextu $\langle X, Y, I \rangle$

ConceptID	Extent	Intent
c(0)	{s1; s2; s3; s4; s5; s6; s7; s8; s9; s10}	{}
c(1)	{s1; s3; s4; s5; s8; s10}	{n10}
c(2)	{s3; s7; s9}	{n8}
c(3)	{s1; s2; s5; s6; s10}	{n7}
c(4)	{s1; s5; s10}	{n7; n10}
c(5)	{s4; s5; s7; s9}	{n6}
c(6)	{s4; s5}	{n6; n10}
c(7)	{s5; s7; s8; s9; s10}	{n5}
c(8)	{s5; s8; s10}	{n5; n10}
c(9)	{s5; s10}	{n5; n7; n10}
c(10)	{s5; s7; s9}	{n5; n6}
c(11)	{s1; s2; s3; s6; s10}	{n4}
c(12)	{s1; s2; s6; s10}	{n4; n7}
c(13)	{s1; s3; s4; s6; s7; s8; s10}	{n3}
c(14)	{s1; s3; s4; s8; s10}	{n3; n10}
c(15)	{s3; s7}	{n3; n8}
c(16)	{s7; s8; s10}	{n3; n5}
c(17)	{s8; s10}	{n3; n5; n10}
c(18)	{s1; s3; s6; s10}	{n3; n4}
c(19)	{s1; s3; s10}	{n3; n4; n10}
c(20)	{s1; s6; s10}	{n3; n4; n7}
c(21)	{s1; s10}	{n3; n4; n7; n10}
c(22)	{s10}	{n3; n4; n5; n7; n10}
c(23)	{s2; s3; s5; s6; s9}	{n2}
c(24)	{s3; s5}	{n2; n10}
c(25)	{s3; s9}	{n2; n8}
c(26)	{s2; s5; s6}	{n2; n7}
c(27)	{s5; s9}	{n2; n5; n6}
c(28)	{s5}	{n2; n5; n6; n7; n10}
c(29)	{s2; s3; s6}	{n2; n4}
c(30)	{s2; s6}	{n2; n4; n7}
c(31)	{s3; s6}	{n2; n3; n4}
c(32)	{s3}	{n2; n3; n4; n8; n10}
c(33)	{s6}	{n2; n3; n4; n7}
c(34)	{s1; s2; s4; s7; s8; s9}	{n1}
c(35)	{s2; s4; s7; s8}	{n1; n9}

Obrázek 4 Seznam konceptů

Věta 2 (hlavní věta o konceptuálních svazech): *Mějme formální kontext* $\langle X, Y, I \rangle$.

1. $\mathcal{B}(X, Y, I)$ *je vzhledem k* $\leq$ *úplný svaz, ve kterém jsou infima a suprema dána předpisy*

$$\bigwedge_{j \in J} \langle A_j, B_j \rangle = \langle \bigcap_{j \in J} A_j, \left(\bigcap_{j \in J} A_j \right)' \rangle = \langle \bigcap_{j \in J} A_j, \left(\bigcup_{j \in J} B_j \right)'' \rangle,$$

$$\bigvee_{j \in J} \langle A_j, B_j \rangle = \langle \left(\bigcap_{j \in J} B_j \right)', \bigcap_{j \in J} B_j \rangle = \langle \left(\bigcup_{j \in J} A_j \right)'', \bigcap_{j \in J} B_j \rangle.$$

2. *Daný úplný svaz* $\boldsymbol{V} = \langle V, \sqsubseteq \rangle$ *je izomorfní s* $\mathcal{B}(X, Y, I)$, *právě když existují zobrazení* $\gamma: X \to V$, $\mu: Y \to V$, *pro která je* $\gamma(X)$ *supremálně hustá ve* $\boldsymbol{V}$, $\mu(Y)$ *infimálně hustá ve* $\boldsymbol{V}$ *a* $\langle x, y \rangle \in I$ *platí právě když* $\gamma(x) \leq \mu(y)$ *(pro každé* $x \in X, y \in Y$*).*

Říkáme, že množina $K \subseteq V$ je supremálně hustá ve $\boldsymbol{V}$, právě když pro každý $v \in V$ existuje $K_v \subseteq K$ tak, že v je supremem množiny K_v; podobně pro infimální hustotu. [1], [], **Chyba! Nenalezen zdroj odkazů.**]

3.1.3 Atributové implikace

(Atributová) implikace (nad množinou Y atributů) je výraz tvaru $A \Rightarrow B$, kde $A, B \subseteq Y$.

Pro implikaci $A \Rightarrow B$ *a množinu* $C \subseteq Y$ *říkáme, že* $A \Rightarrow B$ *platí v* C, *popř. že* C *je modelem* $A \Rightarrow B$, *jestliže platí, že pokud* $A \subseteq C$, *pak i* $B \subseteq C$. *Obecněji, pro množinu* $\mathcal{M} \subseteq 2^Y$ *množin atributů a množinu* $T = \{A_j \Rightarrow B_j \,|\, j \in J\}$ *implikací říkáme, že* T *platí v* $\mathcal{M}$, *popř. že* $\mathcal{M}$ *je modelem* T, *jestliže* $A_j \Rightarrow B_j$ *platí v* C *pro každé* $C \in \mathcal{M}$ *a* $A_j \Rightarrow B_j \in T$.

Říkáme, že implikace platí v kontextu $\langle X, Y, I \rangle$ (popř. že je to implikace kontextu $\langle X, Y, I \rangle$), jestliže platí v systému $\mathcal{M} = \{\{x\}' \,|\, x \in X\}$ obsahů všech objekt-konceptů (tj. obsahů konceptů tvaru $\langle \{x\}'', \{x\}' \rangle$). Dále říkáme, že implikace platí v konceptuálním svazu $\mathcal{B}(X, Y, I)$, jestliže platí v systému $Int(I)$ všech obsahů.

Věta 3: *Atributová implikace platí v* $\langle X, Y, I \rangle$, *právě když platí v* $\mathcal{B}(X, Y, I)$.

Implikace $A \Rightarrow B$ *(sémanticky) plyne z množiny* T *implikací (zapisujeme* $T \vdash A \Rightarrow B$*), jestliže* $A \Rightarrow B$ *platí v každé* $C \subseteq Y$, *ve které platí* T. *Množina* T *implikací se nazývá*

- *uzavřená, jestliže obsahuje každou implikaci, která z ní plyne;*
- *neredundantní, jestliže žádná implikace z T neplyne z ostatních (tj. neplatí $T - \{A \Rightarrow B\} \vdash A \Rightarrow B$).*

Množina T implikací kontextu $\langle X, Y, I \rangle$ se nazývá úplná, jestliže z ní plyne každá implikace kontextu $\langle X, Y, I \rangle$. Báze je úplná a neredundantní množina implikací daného kontextu.

Význam předchozích pojmu je následující. Zajímají-li nás implikace, které ve vstupních datech (tj. v kontextu) platí, nezajímají nás implikace všechny. Zejména nás nezajímají triviální implikace, např. $A \Rightarrow B$, kde $B \subseteq A$, ty můžeme vynechat. Dále je přirozené vynechat ty implikace, které v nějakém přirozeném smyslu plynou z ostatních (proto pojem vyplývání). Při vynechávání bychom měli kontrolovat, zda aktuální množina je stále úplná (tj. všechny implikace z kontextu z ní plynou) a snažit se, aby nebyla redundantní. Následující tvrzení je důsledkem známého výsledku z teorie relačních databází.

Věta 4: *Množina T implikací je uzavřená, právě když, pro každé $A, B, C, D \subseteq Y$ platí*

1. $A \Rightarrow A \in T$;
2. *pokud* $A \Rightarrow B \in T$, *pak* $A \cup C \Rightarrow B \in T$;
3. *pokud* $A \Rightarrow B \in T$ *a* $B \cup C \Rightarrow D \in T$, *pak* $A \cup C \Rightarrow D \in T$ [1],[3].

3.1.4 Vícehodnotové kontexty a konceptuální škálování

Vícehodnotové kontexty (many-valued contexts) jsou rozšířením formálních kontextu, které umožňuje reprezentovat vstupní data i s jinými atributy než jen s bivalentními logickými atributy.

Vícehodnotový kontext je čtveřice $\langle X, Y, W, I \rangle$, kde $I \subseteq X \times Y \times W$ je ternární relace taková, že pokud $\langle x, y, v \rangle \in I$ a $\langle x, y, w \rangle \in I$, pak $v = w$.

Prvky množin X, Y a W se nazývají objekty, (vícehodnotové) atributy a hodnoty atributů. Fakt $\langle x, y, w \rangle \in I$ znamená, že objekt x má atribut y s hodnotou w, píšeme také $y(x) = w$. Vícehodnotové kontexty zřejmým způsobem rozšiřují základní kontexty. FCA přistupuje k analýze vícehodnotových kontextů následovně. Vícehodnotový kontext je prostřednictvím vhodného tzv. konceptuálního škálování (conceptual scaling) převeden na základní kontext, který je poté analyzován.

Škála (scale) pro atribut y vícehodnotového kontextu je kontext $S_y = \langle X_y, Y_y, I_y \rangle$*, pro který* $y(X) \subseteq X_y$ *(kde* $y(X) = \{y(x) | x \in X\}$*). Prvky množin* X_y *a* Y_y *se nazývají škálové hodnoty a škálové atributy.*

Jako škálu pro daný atribut vícehodnotového kontextu můžeme použít libovolný kontext splňující podmínky definice. Nicméně škála by měla odrážet význam daného atributu. Pro atributy, které se ve vícehodnotových kontextech běžně vyskytují, je k dispozici řada standardních škál (např. nominální, ordinální, interordinální, biordinální, dichotomická, atd.

Nyní popíšeme tzv. jednoduché škálování (plain scaling), které je základní procedurou převedení vícehodnotového kontextu na základní kontext.

Je-li $\langle X, Y, W, I \rangle$ *vícehodnotový kontext a jsou-li* $S_y (y \in Y)$ *škály, pak kontext odvozený jednoduchým škálováním je kontext* $\langle X, Z, J \rangle$*, kde*

- $N = \cup_{y \in Y} \dot{Y}_y$ *(*$\dot{Y}_y = \{y\} \times Y_y$*);*
- $\langle x, \langle y, z \rangle \rangle \in J$ *právě když* $y(x) = w$ *a* $\langle w, z \rangle \in I_y$.

Objekty odvozeného kontextu jsou tedy shodné s objekty vícehodnotového kontextu a množina atributů odvozeného kontextu je disjunktním sjednocením atributů jednotlivých škál. Operaci jednoduchého škálování je možné popsat následovně: v tabulce se označení řádků nemění, místo sloupce s označením y vložíme $|Y_y|$ sloupců označených atributy z Y_y a každou hodnotu $y(x)$ z vícehodnotového kontextu nahradíme řádkem škály S_y příslušným objektu x [1],[3].

3.2 Praktická ukázka aplikace FCA

Jako praktický příklad pro využití FCA jsem zvolil vzájemné porovnávání internetových vyhledávačů. Vybrané vyhledávače jsem porovnal podle jejich zaměření (katalogový nebo fulltextový) podle škály služeb, které nabízejí. Z dalších možností jsem zohlednil varování před malwarem, možnost hledání určitých typů souborů (video, PDF) a schopnost prezentace výsledů (třídění a nahléd).

Množina objektů:

$$X = \{Google, Yahoo, Ask, LiveSearch, Seekport, Seznam, AltaVista, Excite\}.$$

Množina atributů:

$$Y = \begin{Bmatrix} hledání\ videa, Náhledy\ výsledků, Malware\ varování, Pdf\ dokumenty, \\ Třídění\ výsledků, fulltextový, katalogový \end{Bmatrix}$$

I	*Hledání videa*	*Náhledy výsledků*	*Malware varování*	*Pdf dokumenty*	*Třídění výsledků*	*fulltextový*	*katalogový*
Google	**1**	0	**1**	**1**	**1**	**1**	0
Yahoo	**1**	0	0	**1**	**1**	**1**	0
Ask	**1**	**1**	0	0	**1**	**1**	0
LiveSearch	**1**	**1**	0	0	**1**	0	**1**
Seekport	0	**1**	0	0	0	**1**	0
Seznam	0	**1**	**1**	0	0	**1**	0
AltaVista	0	0	**1**	**1**	**1**	**1**	0
Excite	**1**	0	**1**	0	0	0	**1**

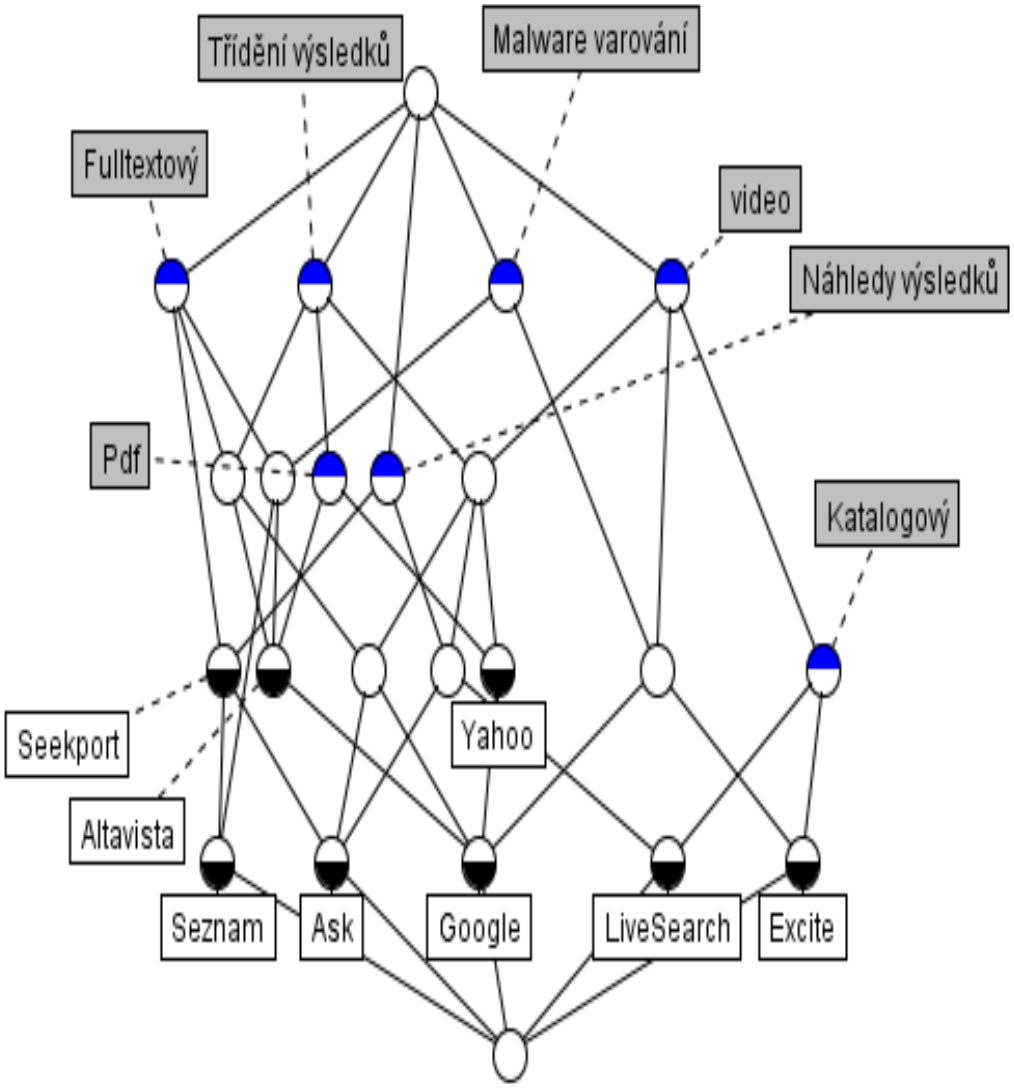
Tabulka 4 Formální koncept internetových vyhledávačů

Seznam konceptů:

{Google; Yahoo; Ask; LiveSearch; Seekport; Seznam; A	{}
{Ask; LiveSearch; Seekport; Seznam; Altavista; Excite}	{Fulltext}
{Google; Yahoo; Ask; LiveSearch; Altavista}	{Třídění výsledků}
{Ask; LiveSearch; Altavista}	{Třídění výsledků; Fulltext}
{Google; Yahoo; Altavista}	{Pdf; Třídění výsledků}
{Google; Seznam; Altavista; Excite}	{Malware}
{Seznam; Altavista; Excite}	{Malware; Fulltext}
{Google; Altavista}	{Malware; Pdf; Třídění výsledků}
{Altavista}	{Malware; Pdf; Třídění výsledků; Fulltext}
{Ask; LiveSearch; Seekport; Seznam}	{Náhledy výsledků; Fulltext}
{Seznam}	{Náhledy výsledků; Malware; Fulltext}
{Google; Yahoo; Ask; LiveSearch; Excite}	{Video}
{Ask; LiveSearch; Excite}	{Video; Fulltext}
{LiveSearch; Excite}	{Video; Fulltext; Katalog}
{Google; Yahoo; Ask; LiveSearch}	{Video; Třídění výsledků}
{Google; Yahoo}	{Video; Pdf; Třídění výsledků}
{Google; Excite}	{Video; Malware}
{Excite}	{Video; Malware; Fulltext; Katalog}
{Google}	{Video; Malware; Pdf; Třídění výsledků}
{Ask; LiveSearch}	{Video; Náhledy výsledků; Třídění výsledků; Fulltext}
{LiveSearch}	{Video; Náhledy výsledků; Třídění výsledků; Fulltext; Katalog}
{}	{Video; Náhledy výsledků; Malware; Pdf; Třídění výsledků; Fulltext; Katalog}

Obrázek 5 Seznam konceptů

Konceptuální svaz:



Obrázek 6 Konceptuální svaz

4 ROZDĚLENÍ INTERNETOVÝCH VYHLEDÁVAČŮ

Informace na internetu lze hledat v podstatě dvěma způsoby - za použití katalogů (katalogové vyhledávače) nebo vyhledávačů (fulltextové vyhledávače). V praxi sice bývají katalogy s vyhledávači částečně propojeny, způsob jejich práce je ale v principu odlišný [4].

4.1 Katalogové vyhledávače

Katalogové vyhledávače představují weby, na kterých jsou shromažďovány odkazy na jiné webové stránky a portály spolu se stručnými popisy. Odkazy jsou tematicky setříděny a uspořádány do kategorií, které umožňují procházení jednotlivých sekcí a vyhledávání jednoduchých dotazů. Setřídění je provedeno pomocí hierarchické struktury. Nejvyšší úroveň reprezentují obecné kategorie (např. Cestování, Sport, Kultura), podkategorie jsou naopak méně obecné a jejich odkazy vedou na další nižší úrovně. Princip hledání v katalogu spočívá v procházení všech kategorií, od obecných, přes méně obecné až ke konkrétním požadovaným výsledkům.

Katalogové vyhledávače lze přirovnat ke kartotékám v knihovně. Jsou zde zařazeny seznamy míst, které byly katalogu předloženy. Je známo co se v konkrétním seznamu nachází, a které položky se v konkrétní kategorii podle určitých slov mají vyhledat. Uživatel tak může procházet kategoriemi katalogu bez nutnosti vyhledávání podle vyhledávacích slov [4],[5].

Zápis do katalogu se obvykle provádí ručně registrací do příslušné sekce. Při registraci je potřeba zadat URL (adresu stránky), popisek a zvolit kategorii zařazení stránky. Návrh zápisu provede správce katalogu (webmaster), který podle určitých pravidel posoudí vhodnost pro danou kategorii. Zápis je poté zkontrolován editorem, upraven a následně schválen. Vzhledem k tomu, že jsou do katalogu zařazeny pouze stránky splňující určité požadavky týkající se kvality, stávají se katalogy výběrovými. Díky tomu dosahují vyšší relevance ve srovnání s fulltextovými vyhledávači.[6], [4].

4.2 Fulltextové vyhledávače

Používání fulltextových vyhledávačů je nejrozšířenější způsob jak vyhledávat na internetu požadované informace. V dnešní době, kdy se datový objem informací na internetu odhaduje přibližně na [DOPLN], představují vyhledávače sofistikovaný způsob, jak uživatel najde to, co potřebuje najít. Jelikož informace, které uživatele zajímají, se nacházejí především v textu webových stránek, neprochází vyhledávače pouze URL adresy či titulky. Vyhledávače naopak procházejí celý text webových stránek, odtud je odvozen i jejich název - fulltextové vyhledávače. Obecně řečeno jsou vyhledávače systémy, které na základě klíčového slova formulovaného uživatelem hledají ve své databázi nebo v indexu. Vyhledávač poté obratem vypíše seznam odkazů na stránky, které hledané informace obsahují. Princip fulltextové vyhledávače se skládá ze 2 částí, z robota a z vyhledávacího programu.

Robot (spider, crawler) je program, který prochází sítí po hypertextových odkazech, navštěvuje webovské stránky a všechna slova v nich obsažená předává indexu. Robot se po webu pohybuje jako po velké pavučině, leze po jejích vláknech, odkazech, které spojují jednotlivé webové stránky mezi sebou. Proto se pro označení robota fulltextových vyhledávačů používá také výraz spider.

Robota je možno rozdělit na další 2 části. Na robota nazývaného getter, který soubory stahuje a na robota nazývaného indexer, který má za úkol stránky zpracovat, vyhodnotit a uložit do databáze na serveru vyhledávače. Hledání tedy probíhá v databázi na serveru vyhledávače, nikoliv přímo v realném čase na internetu. Tento způsob zajistí, že uživatel obdrží výsledek hledání téměř okamžitě po zadání dotazu.

Všechno co robot getter najde je uloženo indexovacím robotem (indexer) do databáze a označeno výrazem index. Index obsahuje každé slovo z navštívených stránek a informace o jeho výskytu na stránce pokud jde o frekvenci a umístění a další údaje. Práce robota je cyklická, robot se po určitém čase na stránky vrací a zjišťuje jejich případné změny.

Změní-li se webová stránka, je index na stránce aktualizován [3]. Druhou částí fulltextového vyhledávače je webové rozhraní, kam uživatel zadává svůj dotaz pro hledání. Webové rozhraní je reprezentováno vyhledávacím programem, který po zadání dotazu prochází index a hledá slova shodná s klíčovými slovy, které jsou uvedeny v dotazu. Po jejich nalezení předloží uživateli soupis webovských stránek, které obsahují požadovaná klíčová slova [6],[7].

4.3 Metavyhledávače

Speciální vyhledávací nástroje, využívající větší počet katalogových vyhledávačů a fulltextových vyhledávačů zapojených do synchronního vyhledávání se nazývají metavyhledávače. Dotaz od uživatele je metavyhledávačem převeden do podoby, které vybrané nástroje rozumějí. Přeložený dotaz je poté metavyhledávačem odeslán předem vybraným nástrojům. Ty porovnají své databáze s dotazem a výsledky hledání odešlou zpět metavyhledávači. Ten je vyhodnotí tak, že vyřadí duplicity a zbývající záznamy uspořádá do soupisu podle volitelných kritérií, zpravidla podle relevance nebo podle jednotlivých použitých vyhledávacích nástrojů. Výsledek vyhledávání je mimo jiné sestaven na základě kvality dílčích dotázaných zdrojů a také podle úspěšnosti nalezených výsledků z předchozích hledání uživatelů [8],[9].

4.4 Google

Autory Googlu jsou pánové Sergey Brin a Lawrence Page. Slovo Google znamená pro většinu lidí internetové stránky stejnojmenné společnosti, přesněji řečeno internetový vyhledávač, pomocí kterého lze na Internetu hledat prakticky cokoliv. Vyhledávání je velice jednoduché a intuitivní, odpovídá totiž poslání společnosti Google, jehož cílem je uspořádat informace z celého světa tak, aby byly všeobecně přístupné a užitečné. Není proto divu, že se vyhledávač rychle rozšířil mezi uživateli a výraz „googlovat“ ve smyslu hledat na internetu zlidověl.

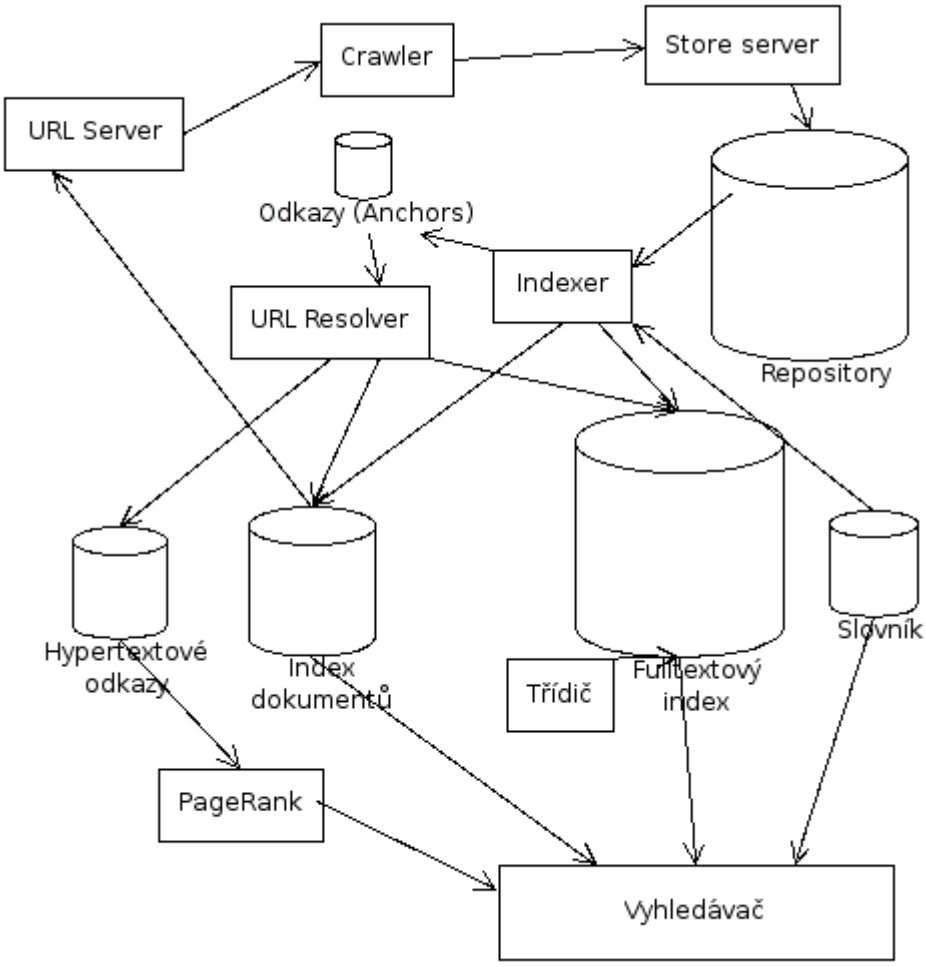
Vyhledávač Google je v současnosti považován za největší a nejpoužívanější na světě – jde o snadno použitelnou bezplatnou službu, která zpravidla zobrazí nejrelevantnější výsledky ve zlomcích sekundy. Právě díky své rychlosti, obsáhlosti, vysoké přesnosti výsledků a absenci reklamy se stal Google tak oblíbeným. Jeho technologii využívá několik dalších velkých vyhledávačů, např. AOL.com, Earthlink.com, AT&T, stejně jako stovky dalších malých vyhledávačů. Celkový podíl vyhledávání technologií Google tak činí 72%. Obliba Google roste i mezi českými uživateli, kdy jeho služeb pro hledání využívá cca 27% uživatelů [10].

Gooogle už dávno není jen světový vyhledávač, zdarma totiž poskytuje spoustu užitečných služeb a aplikací, mezi které například patří emailová schránka (Gmail), kancelářské

aplikace (Google Docs), překladač (Google Translator), webové album (Google Picasa) a spousta dalších [11],[12],[13].

4.4.1 Struktura Google

Crawler stahuje ze sítě dokumenty, které mu určí URL Server. Google využívá několik crawlerů, kteří běží paralelně. Každý najednou udržuje stovky otevřených spojení k webovým serverům, aby se nezdržovaly čekáním na jejich odpovědi. Dokumenty, které crawler stáhne, jsou uloženy do uložiště (depozitáře). V depozitáři se shromáždí všechny dokumenty, jsou kompresovány a obdrží své identifikátory (ID). Dalším modulem je indexer, který prochází uložiště a parsuje dokumenty a pro každé slovo vytváří hit. Hit tvoří výskyt slova v dokumentu, jeho pozice a relativní velikost písma. Hity jsou poté uloženy do zásobníků (barrels), které tvoří částečně setříděný index. Kromě toho Indexer vytváří seznam odkazů ve tvaru (z URL, na URL a text odkazu) a ukládá je do zvláštního souboru, který následně zpracuje URL resolver a zajistí převod relativních URL na absolutní a na ID těchto dokumentů. Texty odkazů jsou zahrnuty v indexu k danému dokumentu, na nějž odkazují. Tyto informace o vzájemných odkazech slouží pro výpočet PageRanku. Modul třídič přetřiďuje index do zpětného indexu. Vyhledávač (searcher) běží na webovém serveru a s pomocí slovníku, zpětného indexu a PageRanků odpovídá na dotazy [14], [15].



Obrázek 7 Architektura vyhledávače Google

4.4.2 Analýza kvality stránek – Google PageRank

Technologii Google PageRank vyvinuli zakladatelé Google – Larry Page a Sergey Brin na Stanford University jako algoritmus pro ohodnocení kvality a důležitosti webových stránek. Výsledkem algoritmu PageRank je číselná hodnota, která vyjadřuje důležitost webové stránky, tedy významnost stránek z hlediska vyhledávačů. Jedním z kritérií pro výpočet PageRanku je počet odkazů, které na danou stránku vedou. Kromě zpětných odkazů se také zohledňuje hodnocení odkazujících stránek. Přesný vzorec na výpočet PageRanku je přísně utajen, zná jej pouze několik zaměstnanců Google. Přesný Vzorec pro výpočet hodnoty PageRank uvedl Google před několika lety (tady zkusím zjistit rok). Tento vzorec se dnes již nepoužívá, ale výpočet je s velkou pravděpodobností podobný [16].

$$PR(A) = (1 - d) + d(PR(T1)/C(T1) + \ \ldots + PR(Tn)/C(Tn))$$

Parametr $PR(A)$ označuje PageRank stránky A, $PR(T1)$ $až$ $PR(Tn)$ je PageRank stránek, které odkazují na stránku A. Parametr d je tlumící faktor, jehož hodnota se pohybuje mezi 0 a 1 (většinou se uvádí 0,85) a parametr $C(Tn)$označuje počet odkazu na stránkách, které odkazují na stránku A.

Zjednodušeně lze celý algoritmus popsat takto: Web A odkazující na web B předává část svého PageRanku. Čím má stránka A vyšší PageRank (čím je kvalitnější), tím větší PageRank předává stránce B. Čím více se na odkazující stránce A nachází dalších odkazů, tím méně PageRanku předává dál.

Čím je hodnota PageRanku vyšší, tím je stránka důvěryhodnější a tím výše se ve vyhledávačích nachází, to ale nemusí platit ve všech případech. Hodnota PageRanku je pouze jeden ze stovky faktorů, které Google používá pro řazení stránek ve výsledcích vyhledávání [17].

Hodnotu PageRanku lze zjistit pomocí lišty Google Toolbar, která se nainstaluje do internetového prohlížeče. Údaj, který lišta ukazuje, se nazývá Google Toolbar PageRank (GTPR) a vyjadřuje zaokrouhlenou (přibližnou) hodnotu skutečného PageRanku. Tato hodnota se pohybuje ve škále od 1 do 10, přičemž nabývá pouze celých čísel. Hodnoty GTPR se aktualizují najednou přibližně každé tři nebo čtyři měsíce, zatímco k aktualizacím skutečného PageRanku dochází častěji [18],[19].

Protože většina běžně používaných vyhledávačů (viz. např. Google nebo CiteSeer) používá pro konkretní dotaz jen statistické metody (např. četnost a histogram), je účelné rozšířit možnosti vyhledávačů pomocí metod teorie formální konceptualní analýzy. Metody formální konceptualní analýzy umožní odstranit hlavní problém současných vyhledáčů, které nezobrazí relevantní odkazy na původní dotaz, protože jsou popsány jinými výrazy. Tento problém řeší FCA tím, že vytvoří větší konceptualní svaz z bohatšího kontexu, který dostaly vyhledávače na základě původního dotazu.

4.4.3 Google aplikace - Google Schoolar

V listopadu 2004 představil Google svůj nový produkt, vyhledávač vědecké literatury Google Scholar. Služba Google Scholar představuje jednoduchý způsob uceleného vyhledávání odborné literatury. Z jednoho místa je tak možné vyhledávat informace z mnoha oborů a zdrojů. Orientace vyhledávače je opravdu široká, recenzované články, dizertační práce, knihy, abstrakty a články, od akademických nakladatelství, odborných společností, archivů preprintů a dalších odborných organizací. Cílem vývoje bylo ulehčit uživateli vyhledávání vědecké a odborné literatury a nezahrnout ho tak množstvím neodborných informací, které svět internetu nabízí [20].

Funkce služby Google Scholar:

- Vyhledávání rozmanitých zdrojů z jednoho místa
- Hledání článků, abstraktů a citací
- Nalezení celého článku v knihovně či na webu
- Informace o klíčových článcích v jakémkoli vědním oboru

Vyhledávat je umožněno dvěma způsoby. K dispozici je buď klasické jednoduché vyhledávací okénko pro zadání „klíčových" slov nebo vyhledávací formulář určený pro rozšířené vyhledávání. Rozšířené vyhledávání umožňuje nastavení dalších atributů, které zvyšují přesnost nalezení cílového dokumentu. Mezi atributy, které je možno nastavit patří například omezení hledání na jméno autora či datum vydání publikace. Výsledek vyhledávání je reprezentován jako seznam prací, přičemž každá práce je tvořena jedním či několika různými projevy téhož článku, který je umístěn například na webu autora nebo je publikován v časopise. Google Scholar články automatizovaně setřídí a seskupí všechny projevy daného článku dohromady do jedné vědecké práce. Nejrelevantnější výsledky hledání jsou zobrazeny vždy na první stránce. Pro každý z nalezených výsledků je k dispozici řada funkcí

- Cited by (Počet citací tohoto článku): uvádí práce, které citovaly danou práci a jsou obsaženy vdatabázi GS;
- Related articles (Související články): hledá další práce, které jsou tematicky podobné článkům dané práce;

- Web Search (Hledání na webu): vyhledává informace o dané práci na webu, přes klasický webový vyhledávač Google;
- All x versions (Všechny verze - počet: x): uvádí, z kolika projevů se skládá daná práce a umožňuje všechny projevy práce zobrazit [21],[22].

4.5 Jyxo

Jyxo je technologie pro zpracování rozsáhlého množství dat: sběr, analýzu, vyhledávání. Značka Jyxo především výkonný český fulltextový vyhledavač, jehož kvality docení zejména zkušenější a náročnější uživatelé internetu. Díky tomu, že vyhledávač obsahuje lingvistický modul, který umí ohýbat česká slova, nabízí české skloňování a časování. Touto vlastností nedisponuje žádný zahraniční vyhledávač. Další předností tohoto vyhledávače je častá aktualizace informací, jejichž databáze je aktualizována každý den, oproti konkurenčním českým vyhledávačům. Unikátní vlastností je členění dokumentů do tematických skupin dle jejich typu a příslušnosti. Uživatel tak může snadno upřesnit své vyhledávání klepnutím na název skupiny [23].

Jyxo dále nabízí různé speciální volby pro vyhledávání – umožňuje např. hledat pouze v článcích nebo diskusích, dovede hledat obrázky, video soubory, zboží v internetových obchodech nebo například zájezdy. Kromě standardního vyhledávání internetových stránek v jazyce HTML a již zmíněných speciálních předvoleb zvládne Jyxo i vyhledávání v netradičních formátech PDF nebo MS WORD [24].

4.5.1 Analýza kvality stránek Jyxo - JyxoRank

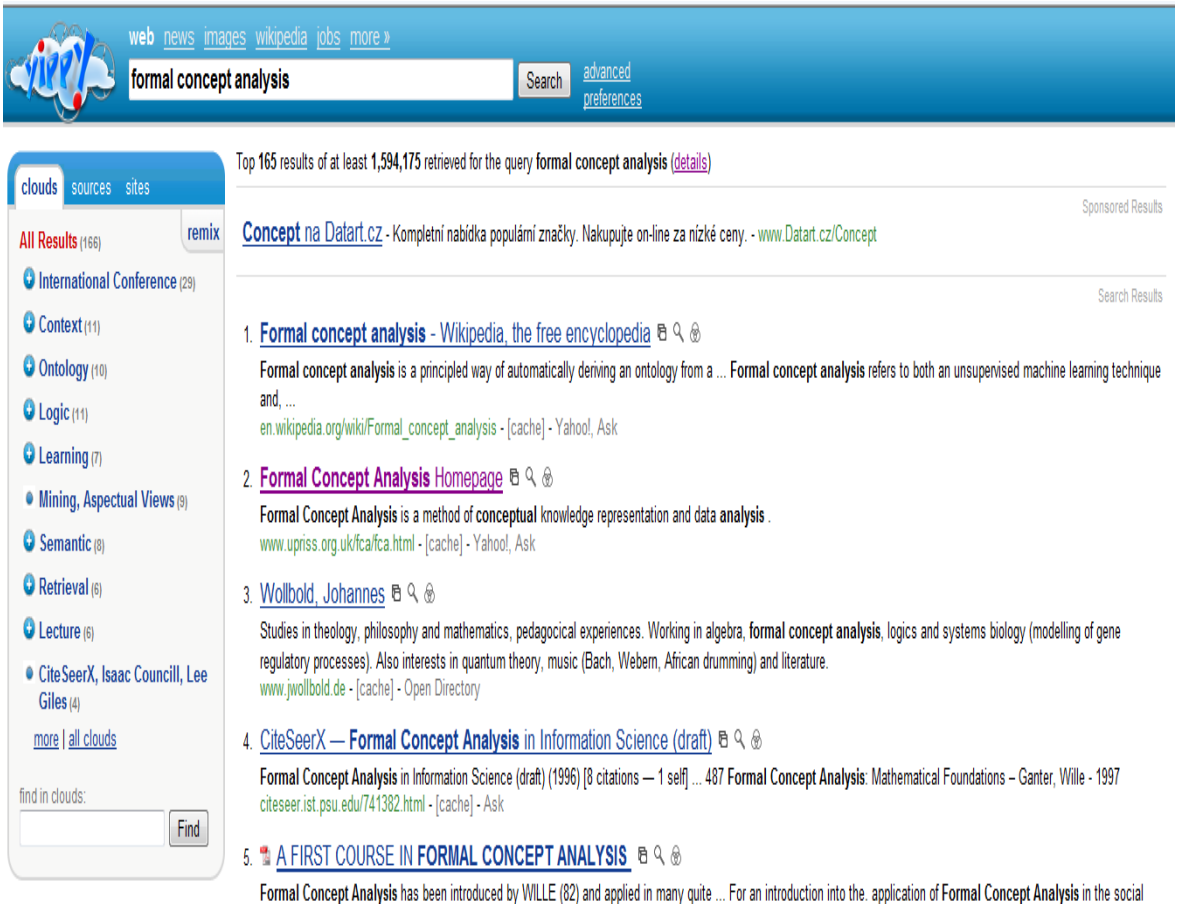
Také fulltextový vyhledávač Jyxo využívá pro nezávislé ohodnocení kvality webových stránek svůj systém (algoritmus) nazvaný JyxoRank. Princip funkce hodnocení stránek podle zpětných odkazů tohoto algoritmu má podobné rysy jako PageRank od Google nebo ostatní používané PageRanky. Hodnota JyxoRanku je počítána na základě všech odkazů v databázi. Výpočet hodnoty probíhá iterativně (několika průchody). Na rozdíl od algoritmů ostatních vyhledávačů nehodnotí JyxoRank pouze zpětné odkazy, ale vnáší do algoritmu nový element. Navíc sleduje i to, kdo se na danou stránku odkazuje, přesněji řečeno z jakých domén a z jakých IP adres je na stránku odkazováno. Pokud by se

na webu nacházely 3 odkazy na ostatní cílové stránky algoritmus od Google by výsledný PageRank spočítal jako součet zlomků z Pageranků odkazujících stránek, zatímco JyxoRank prozkoumá domény a IP adresy stránek [25]. Prozkoumá, co mají stránky společného a co rozdílného a podle toho stránky ohodnotí několika nezávislými zdroji. Z tohoto pohledu je výpočet algoritmu JyxoRank složitější z hlediska naprogramování i na hardware serveru, kde se neukládají pouze údaje o stránce, ale i to kdo stránku doporučil. Právě tento element přináší spravedlivější ohodnocení dokumentů a z toho plynoucí větší přesnost vyhledávání. Největším přínosem je však odolnost vůči zneužití, nedochází totiž k tomu, že se do výpočtu výsledného (Rank) hodnocení webu započítají stránky, které na sebe odkazují navzájem, jak k tomu dochází u algoritmu PageRank. Milión stránek, které odkazují samy na sebe navzájem či na nějakou cílovou stránku si na webu muže zřídit kdokoliv, zatímco pořídit si milion domén a IP adres není snadné a ekonomicky výhodné. Jyxo Rank nabývá hodnot v rozmezí 0 – 220, kdy platí pravidlo, že čím vyšší, tím lepší [26]. Výsledné hodnocení se počítá pro každou webovou stránku (URL adresu) samostatně, podobně jako např. S-rank. Hodnotu JyxoRanku lze vyčíst z přídavné lišty (JyxoToolbaru). Její aktualizace probíhá denně, přičemž při procesu přepočítávání je hodnota nulová nebo se nezobrazuje vůbec žádná hodnota. Tento stav způsobuje časová prodleva na centrálním serveru Jyxo.

4.6 Yippy

I když jsou internetové vyhledávače velmi užitečné, jejich plné využití je často omezené. Nalezených informací je zpravidla velmi mnoho a jen jejich malá část je skutečně použitelná. Právě z tohoto důvodu byl firmou Vivisimo vyvinut metavyhledávač Yippy. Původně byl metavyhledávač znám pod jménem Clusty, ale v roce 2010 ho odkoupila floridská společnost Yippy, po které byl také pojmenován [27],[28]. Yippy nabízí rozumnější formu prohledávání internetu a automatické třídění nalezených odkazů do kategorií dle obsahu. Proto je Yippy někdy označován jako vyhledavač nové generace. Funguje na principu takzvaného shlukování dokumentů (automatické kategorizace), takže dokáže nabídnout vyhledané odkazy seřazené do několika tematických okruhů. Díky této technologii dochází k automatickému rozdělování (kategorizaci) textových informací do výrazných, smysluplných, hierarchicky utříděných složek kategorií, které je možné procházet ve formě stromu. Tím lze výrazně zkrátit dobu hledání správného dokumentu.

Princip funkce systému lze přirovnat k super rychlému knihovníkovi, který dokáže chaotický obsah obrovské knihovny uspořádat během okamžiku do kategorií způsobem, který má smysl [29]. Algoritmus, který Yippy používá je založen na kombinaci metod lingvistiky a statistiky, nepoužívá předdefinované kategorie – vytváří je až během zpracování podle obsahu nalezených informací. V levém panelu máme na výběr 3 záložky, clouds, sources a sites. Záložkou clouds můžeme vybrat požadovanou kategorii, záložka sources zobrazuje výsledky, které nabízí ten který vyhledávač. Poslední záložka sites dovoluje roztřídit výsledky hledání podle národních domén.



Obrázek 8 Rozhraní vyhledávače Yippy

Klasické shlukové (clusteringové) techniky používané pro vyhledávání informací je možné kombinovat s FCA. Výsledkem je konceptuální shluková technika.

4.6.1 Konceptuální shluková technika

Konceptuální technika shlukování, přináší oproti klasickým shlukovým algoritmům následující výhody:

a) poskytuje intenzionální popis každého shluku (clusteru), shlukování je tím pádem lépe vyhodnotitelné,

b) organizace shluku netvoří svaz, což umožňuje znovu opakovat dotaz, ale zkoumání hierarchie poskytuje bohatší a flexibilnější způsob procházení dokumentů než klasické hierarchické shlukování.

Myšlenka použití FCA jen na malou podmnožinu dokumentů (v našem případě výsledky vyhledávání) odstraňuje některé problémy spojené s používáním FCA ve vyhledávání informací:

- FCA je výpočetně dražší než standardní shlukování, ale obě 2 techniky se mohou stejným způsobem (rovnoměrně) uplatnit na malých sadách dokumentů (v rozsahu 50 až 500), což je dostatečně efektivní pro on-line aplikace
- Svazy generované FCA mohou být velké a složité, tím pádem jsou těžko použitelné pro účely praktického hledání. K tomuto problému dochází při aplikaci na velké sady dokumentů, kdy jsou produkovány nezvladatelné struktury. Problém je eliminován, pokud je soubor dokumentů omezen velikostí a tématem z předchozího hledání

U konceptuálních shlukových technik existují dvě různé metriky vztahující se k uživatelskému dotazu při hledání relevantní informace. Lattice distillation factor (LDF), který měří, jak dobře shluky dokumentů ve svazu brání uživateli v přístupu k irelevantním dokumentům (v porovnání s původním řazeným seznamem, který vrací vyhledávač). Druhou technikou je lattice browning complexity (složitost procházení svazu), která měří

počet popisu uzlu, které musí byt posouzeny k dosažení relevantních informaci. Optimální svaz bude mít vysoký faktor destilace a nízkou složitost procházení**.**

Co chceme měřit, je to, jestli struktura svazu efektivně „destiluje“ příslušné dokumenty dohromady a umožňuje tak uživateli nalézt relevantní informace rychleji a lépe než v seřazeném seznamu dokumentů.

Laticce distillation faktor (LDF) je soubor opatření, která se opírají o teorii procházení minimální oblasti.

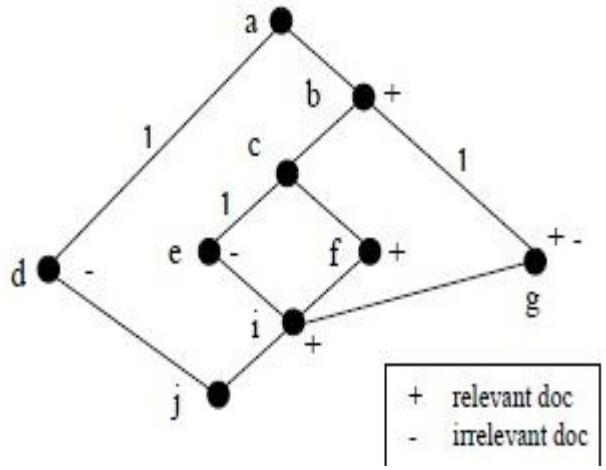
Lattice Destillatin Factor – destilační faktor svazu

Nechť C je množina uzlů v konceptuálním svazu, kde jsou všechny tyto dokumenty označeny jako relevantní nebo non-relevantní pro daný (položený) dotaz. Předpokládejme, že při návštěvě uzlu, uživatel vidí dokumenty, u nichž uzel představuje předmět jejich zájmu.

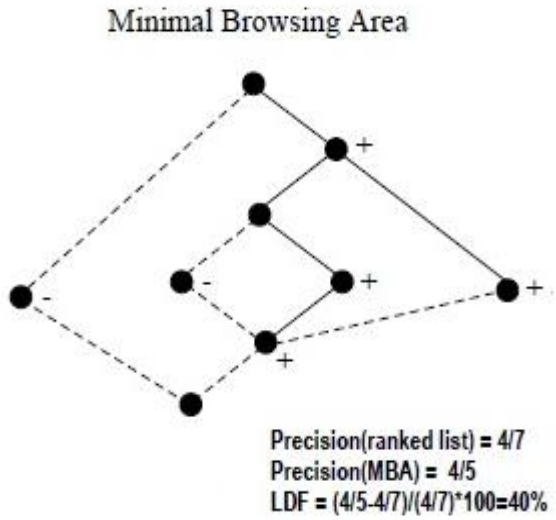
Pojem relevantní koncept budeme používat k označení konceptů, které tvoří alespoň jeden relevantní dokument, a pojem irelevantní koncept k označení konceptů, který tvoří pouze jeden nebo více irelevantních dokumentů.

Definujeme $C_{REL} \subseteq C$ jako podmnožinu relevantních konceptů v svazu. Za účelem nalezení všech relevantních dokumentů zobrazených ve svazu, uživatel musí zkoumat, alespoň obsah všech konceptů v C_{REL}.

Definujeme minimální prohledávací prostor (MBA) jako minimální část svazu, který by měl uživatel prohledávat. Začíná se od vrcholu uzlu směrem ke všem relevantním konceptům v C_{REL}, tak se minimalizuje počet irelevantních dokumentů, které musí být kontrolovány za účelem získání všech relevantních informací.



Obrázek 9 Počáteční koncepce svazu



Obrázek 10 Redukovaný konceptuální svaz

Přesnost MBA je vyjádřena jako poměr mezi relevantními dokumenty a všemi dokumenty v MBA. Horní mez udává kapacitu svazu destilovat relevantní informace z výsledku hledání. Dolní mez je přesnost (precision) originálního seznamu: uživatel musí

proscanovat všechny získané dokumenty, aby se ujistil, že žádné relevantní dokumenty nejsou ze seznamu vynechány.

Faktor destilace svazu (LDF) může být pak definován jako potenciální dosažený stupeň přesnosti mezi svazem a pořadím seznamu.tj. jako procentuální stupeň přesnosti mezi procházením minimální oblasti a původním pořadí seznamu

$$\mathrm{LDF} = \frac{\mathrm{Precision}_{\mathrm{MBA}} - \mathrm{Precision}_{\mathrm{RL}}}{\mathrm{Precision}_{\mathrm{RL}}} . 100$$

Minimální procházení oblasti a faktor destilace lze použít stejně na hierarchické shlukování nebo jakékoliv jiné grafické seskupení výsledků hledání [30].

5 DATAMINING

Datamining je informační technologie, která se zabývá získáváním znalostí z již existujících naplněných databází informací. Často se o datamininguu mluví také jako o technice dolování znalostí. Datamining je velmi rozsáhlé odvětví, v němž se využívá strojové učení a další obory umělé inteligence. Ke klíčovým problémům patří nalezení zajímavého vzorku (clustering), popis vzorku (clasification) a hledání závislostí. Často se říká, že klasické technologie slouží k vyhledávání toho, co víme, že nevíme. Pomocí nástrojů datamining se můžeme posunout směrem k nalézání toho, co nevíme, že nevíme.

Vyhledávač Google, používá k řazení pořadí výsledné množiny stránek již zmíněný algoritmus Pagerank, který určuje důležitost hodnocené stránky. Formální konceptuální analýza (FCA) se aplikuje až na seznam stránek seřazených tímto algoritmem. Ze seřazeného seznamu stránek je poté vytvořen kontext, který dokáže zefektívnit celý proces vyhledávání dokumentů či webových stránek.

Další z popsaných vyhledávačů Jyxo využívá pro sběr dat technologii, která má modulární strukturu, k analýze kvality webových stránek či dokumentů je hodnocena rovněž vlastním algoritmem s názvem JyxoRank.

Existují ovšem i internetové vyhledávače, využívají technik shlukování dokumentů spolu s aplikací formální konceptuální analýzy. Právě takovým případem je zmíněný vyhledávač Yippy z produkce firmy Vivisimo.

ZÁVĚR

Ve své diplomové práci jsem se zabýval analýzou internetových vyhledávačů pomocí metod formální konceptuální analýzy. V teoretické části jsem nejprve definoval základní pojmy teorie svazů. Dále jsem popsal uzávěrové operátory včetně věty o pevném bodě.

Praktická část je rozdělena na 3 kapitoly. Nejprve uvádím základní pojmy a tvrzení formální konceptuální analýzy, po níž následuje formulování základní reprezentační věty pomocí Galoisových konexí. Kapitola formální konceptuální analýzy je završena hierarchickým uspořádáním (tzv. Hasseovým diagramem), vztahujícím se k tématu práce. Dále je v praktické části uvedeno základní členění internetových vyhledávačů, podle jejich funkce. Základní popis činnosti fulltextového vyhledávače je demonstrován pomocí prototypu architektury, kterou k hledání využívá Google. Zmíněny jsou i některé další používané vyhledávače, přičemž jsou popsány algoritmy analyzující kvalitu webových stránek, tedy významnost stránek z hlediska vyhledávačů. Kromě vyhledávačů, které k řazení výsledku používají své vlastní algoritmy, jsou zmíněny i ty, které k zobrazení relevantních výsledků využívají shlukovací techniky, respektive konceptuální shlukovací techniky. Byla nastíněna myšlenka použití FCA na malou skupinu dokumentů. Jelikož se jedná o kombinaci klasických shlukovacích technik s metodami formální konceptuální analýzy, bylo nutné stanovit 2 základní metriky vztahující se k uživatelskému dotazu při hledání relevantní informace. Konkrétně byly zavedeny pojmy LDF (Lattice distillation factor) a lattice browsing commplexity (složitost procházení svazu). Dále byl stanoven pojem MBA jako (takzvaná minimální oblast, kterou bude uživatel procházet za účelem hledání relevantních dokumentů). Zavedením uvedených faktorů bylo zjištěno, že procházení minimální oblasti a faktor destilace lze použít jak na hierarchické seskupení nebo jakékoliv jiné grafické seskupení výsledků hledání.

V posledním bodě práce byl zaveden pojem vytěžování informací (data mining) a současně byly shrnuty metody řazení dokumentů již dříve popsaných vyhledávačů.

ZÁVĚR V ANGLIČTINĚ

I dealt in the analysis of internet search engines by with using formal concept analysis in my work. There are in the theoretical part introduced the basic concepts of the lattice theory and closure operators with fixed-point theorems.

The practical part is divided into 3 chapters. First, I present the basic concepts and conceptual analysis of the formal statement, followed by the formulation of basic sentences using representational Galois connections. Chapter formal conceptual analysis is completed by a hierarchical structure (called Hasseovým diagram), relating to the subject work. Chapter formal conceptual analysis is completed by graphical representation of concept lattices in the form of Hasse diagrams. Furthermore,The practical part concerns a comparison of existing search engine, according to their function. The basic description of a full-text search engine is demonstrated by the prototype architecture, which Google uses for search. There are described some other search engines and algorithms to analyze the quality of websites, thus the significance of sites in terms of search engines. In addition to search engines which use to sort results its own algorithms, which are discussed as well as those that display the relevant results of using the cluster technique, or conceptual grouping techniques. It was outlined the idea of using FCA to a small group of documents. Since it is a combination of classical clustering techniques with formal methods of conceptual analysis, it was necessary to provide two basic metrics related to a user query to find relevant information. Specifically, the concepts were introduced by LDF (Lattice distillation factor) and lattice complexity browsing (browsing the complexity of association). Furthermore, the concept was introduced as an MBA (the so-called minimum area that the user will browse to find relevant documents). It was found with Introduction of those factors that the minimal browsing area and a factor of distillation can be used as a hierarchical group or any other graphical grouping of search results.

SEZNAM POUŽITÉ LITERATURY

[1] WILLE, R., GANTER, B. *Formal Concept Analysis – Mathematical Foundations*. 1st ed. Springer, 1998. 284 s. ISBN 3-540-62771-5.

[2] KUČERA, Radan. *Základy teorie svazů* [online]. [cit. 2010-05-05]. Dostupný z WWW: <http://www.math.muni.cz/~kucera/texty/Svazy2003.pdf>.

[3] BĚLOHLÁVEK, Radim. *Konceptuální svazy a formální konceptuální analýza* [online]. [cit. 2010-05-05]. Dostupný z WWW: <http://belohlavek.inf.upol.cz/publications/Bel_Ksfka.pdf>.

[4] *Ataxo.cz* [online]. 2010 [cit. 2010-06-05]. Jak fungují katalogy. Dostupné z WWW: <http://www.ataxo.cz/info/vyhledavace/katalogy/>.

[5] *Voxcafe.cz* [online]. 3.12.2006 [cit. 2010-06-05]. Jak pracují vyhledávače. Dostupné z WWW: <http://www.voxcafe.cz/clanky/optimalizace-stranek/seo-versus-uzivatele-a-vyhledavace.html>.

[6] *volny.cz* [online]. 3.8.2007 [cit. 2010-06-05]. Katalogy (Directors). Dostupné z WWW: <http://www.volny.cz/ist987/priloha/page/textHTML.html>.

[6] *webprezent.cz* [online]. 3.8.2007 [cit. 2010-06-05]. Rozdíl mezi katalogem a vyhledávačem. Dostupné z WWW: <http://www.webprezent.cz/webdesign/diplomka/kapitoly/2-1-rozdil-mezi-katalogem-vyhledavacem>.

[7] *Ataxo.cz* [online]. 2010 [cit. 2010-06-05]. Roboti a crawleři aneb jak fungují fulltextové vyhledávače. Dostupné z WWW <http://www.ataxo.cz/info/vyhledavace/fulltextove-vyhledavace/>.

[8] *Vyhledavac.oblibena.net* [online]. 6.5.2010 [cit. 2010-06-05]. Metavyhledávač. Dostupné z WWW: <http://vyhledavac.oblibena.net/metavyhledavac.html>.

[9] *Infogram.cz* [online]. 2008 [cit. 2010-06-07]. Světové metavyhledávače. Dostupné z WWW: <http://www.infogram.cz/article.do?articleId=1763>.

[10] *Google.com* [online]. 2008 [cit. 2010-06-07]. About Google. Dostupné z WWW: <http://www.google.com/intl/en/about.html>.

[11] *pcmagazin.cz* [online]. 2008 [cit. 2010-06-07]. Google jen není jen vyhledávač. Dostupné z WWW:<http://pcmagazin.cz/internet/830-google-neni-jen-vyhledavac.html>.

[12] *ataxo.cz* [online]. 2008 [cit. 2010-06-07]. O Google. Dostupné z WWW: <http://www.ataxo.cz/info/vyhledavace/google/#zdroj-2>

[13] *swmag.cz* [online]. 2008 [cit. 2010-06-07]. Google vyhledávač. Dostupné z WWW: < http://www.swmag.cz/12/google-vyhledavac/ >.

[14] *Lupa.cz* [online]. 22.5.2002 [cit. 2010-06-07]. Jak vypadá Google uvnitř. Dostupné z WWW: <http://www.lupa.cz/clanky/jak-vypada-google-uvnitr/>.

[15] *Wikipedia.org* [online]. 23.11.2009 [cit. 2010-06-07]. Google (Vyhledávač). Dostupné z WWW: <http://cs.wikipedia.org/wiki/Google_(vyhled%C3%A1va%C4%8D)>.

[16] ROZEHNAL, Jan. *Artic Studio* [online]. 12. březen 2006 [cit. 2010-06-07]. Google PageRank. Dostupné z WWW: <http://www.artic-studio.net/clanky/google-pagerank/>.

[17]*Rank* [online]. [cit. 2010-06-07]. Stefajir. Dostupné z WWW: <http://www.stefajir.cz/?q=ranky>.

[18] JANOVSKÝ, Dušan . *Lupa.cz* [online]. 25 ledna 2005 [cit. 2010-06-07]. Záhadný Google Toolbar PageRank. Dostupné z WWW: <http://www.lupa.cz/clanky/zahadny-google-toolbar-pagerank/>.

[19] ILLICH, Michal . *Lupa.cz* [online]. 23 června 2003 [cit. 2010-06-07]. PageRank a jeho rozšíření. Dostupné z WWW: <http://www.lupa.cz/clanky/pagerank-a-jeho-rozsireni/>.

[20] *Google scholar* [online]. 1. ledna 2010 [cit. 2010-06-07]. Google scholar. Dostupné z WWW: <http://scholar.google.cz/intl/cs/scholar/about.html>.

[21] *Google scholar* [online]. 21. ledna 2010 [cit. 2010-06-07]. Google Scholar. Dostupné z WWW: <http://google-cz.blogspot.com/2008/01/google-scholar.html>.

[22] *Jyxo.cz* [online]. [cit. 2010-06-07]. Jyxo Dokumentace. Dostupné z WWW: <http://jyxo.cz/d/info>.

[23] *Ataxo* [online]. [cit. 2010-06-07]. Jyxo – www.jyxo.cz – český vyhledávač pro náročné. Dostupné z WWW: <http://www.ataxo.cz/info/vyhledavace/jyxo/>.

[24] *Lupa.cz* [online]. 23. 6. 2003 [cit. 2010-06-07]. PageRank a jeho rozšíření. Dostupné z WWW: <http://www.lupa.cz/clanky/pagerank-a-jeho-rozsireni/>.

[25] *Jyxo Rank Checker* [online]. 1.ledna 2007 [cit. 2010-06-07]. Jyxorank. Dostupné z WWW: <http://pagerank.jklir.net/?p=jyxorank>.

[26] *Phil.muni.cz* [online]. 1. ledna 2010 [cit. 2010-06-07]. Vyhledávač VIVÍSIMO - dobrý pomocník. Dostupné z WWW: <http://www.phil.muni.cz/~hoskova/Vivisimo.htm>.

[27] *Pandia.com* [online]. 25. května 2010 [cit. 2010-06-07]. Metasearch engine Clusty becomes Yippy. Dostupné z WWW: <http://www.pandia.com/sew/2911-metasearch-engine-clusty-becomes-yippy.html>.

[28] *Kryl.info* [online]. 30.9.2004 [cit. 2010-06-07]. Clusty-nový pohled na vyhledávání. Dostupné z WWW: <http://kryl.info/clanek/156-clusty-novy-pohled-na-vysledky-vyhledavani>.

[29] BRDIČKA, Bořivoj . *Učitelský spomocník* [online]. 27. září 2004 [cit. 2010-06-07]. Nové vyhledávací služby určené pro výuku. Dostupné z WWW: <http://www.spomocnik.cz/index.php?id_document=122>.

[30] Cigarran, Juan. *Browsing Search Results Via Formal Concept Analysis: Automatic Selection of Attributes* [online]. [cit. 2010-05-05]. Dostupný z WWW: <http://nlp.uned.es/pergamus/pubs/icfca2004.pdf>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

FCA Formal Concept Analysis

LDF Lattice distillation factor

MBA Minimal browning area

PR PageRank

SEZNAM OBRÁZKŮ

SEZNAM TABULEK