Určení smysluplnosti české věty
LS 2007/2008, MFF UK

Vladimír Rovenský
V.Rovensky@lit.cz

Specifikace ročníkového projektu

Vedoucí projektu: Barbora Vidová-Hladká
Autor: Vladimír Rovenský
Programovací jazyk: C++ (IDE MS Visual Studio)
Platforma: Unix, Windows

Zadání

Navrhnout a implementovat automatickou proceduru, která na základě tvaroslovné (morfologické) informace poskytnuté programem *tool_chain* rozhodne, zda je česká věta smysluplná či nikoli. Větou může být i otázka. Součásti práce je vhodná definice smysluplnosti věty vzhledem k implementaci.

Způsob řešení

Celá věta bude nejprve rozdělena na věty jednoduché. Souvětí je smysluplné, právě když jsou smysluplné všechny jeho věty jednoduché.

Smysluplnost jednoduché věty bude definována na základě vztahů mezi jednotlivými slovními jednotkami věty. Slovní jednotka (neboli token) je základní atom, na které větu rozdělí nástroj *tool_chain* a nejčastěji odpovídá jednomu slovu nebo jednomu interpunkčnímu znaménku. Ke každé slovní jednotce přidává *tool_chain* rovněž tvaroslovnou informaci. Tvaroslovná informace je reprezentována řetězcem pevné délky (a sice 15 znaků), ve kterém každá pozice jednoznačně odpovídá právě jedné tvaroslovné kategorii. Například na první pozici je informace o slovním druhu, na druhé o poddruhu, atd. Tyto řetězce jsou označovány jako morfologické značky, případně tagy.

Základem práce programu je nalezení smysluplných dvojic slovních jednotek ve větě. To jsou ty dvojice, jejichž morfologická informace si určitým způsobem odpovídá.

Konkrétně budou pro každou dvojici slovních druhů s_1,s_2 definovány podmínky, za kterých budou dvě slovní jednotky, z nichž jedna má slovní druh s_1 a druhá má slovní druh s_2 tvořit smysluplnou dvojici. Program projde všechny dvojice slovních jednotek ve větě a na základě jejich slovních druhů vybere příslušné podmínky, jejichž platnost následně ověřuje. Podmínek pro jednu dvojici slovních druhů může být více (nebo žádná). Pokud testovaná dvojice slovních jednotek splňuje alespoň jednu z nich, potom tvoří smysluplnou dvojici slovních jednotek.

Věta bude dále zpracovávána jako graf, jehož vrcholy budou tvořit slovní jednotky a hrany budou odpovídat smysluplným dvojicím slovních jednotek (tj. $\forall t_1,t_2$ slovní jednotky: $t_1,t_2 \in E(G) \Leftrightarrow$ existuje podmínka pro smysluplnou dvojici, které t_1,t_2 vyhovují). Celá věta bude vyhodnocena jako smysluplná, pokud bude vzniklý graf souvislý. Součástí projektu bude soubor takovýchto podmínek ve formátu definovaném v dokumentaci i nástroje pro jeho upravování a rozšiřování.

V podmínkách bude možné se odkázat např. na morfologické informace slovní jednotky, slovosled, vzdálenost slovních jednotek, maximální počet dvojic daného typu atp. Struktura programu bude navržena s ohledem na snadné přidávání a úpravu těchto pravidel.

Mimo výše uvedeného se bude dále testovat zapojení spojek ve větě (podobně jako u smysluplných dvojic, jen ternární vztah – spojka a dvě slovní jednotky) a další podmínky dle potřeby.

Příklad podmínky pro smysluplnou dvojici

```
rel     N     R
2       N     (R V)
5       3
ord     2     1
lemma             (k proti naproti oproti kvůli díky)
end
```

Výše je uvedena jedna z podmínek, které využívá program k určování smysluplných dvojic slovních jednotek. Tato popisuje vztah předložky a podstatného jména ve třetím pádě.

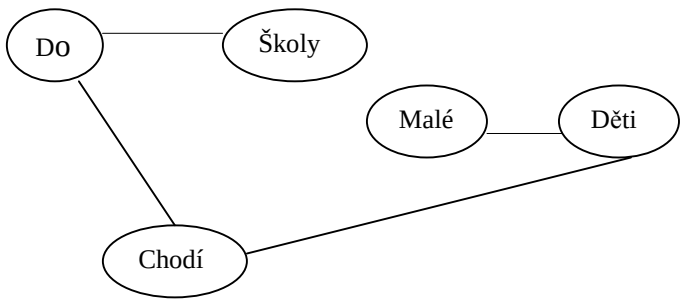
První řádek je uvozen klauzulí *rel* a tvoří „deklaraci“ podmínky, tedy jaké dvojice slovních druhů se týká. V tomto případě se jedná o popis dvojice předložka (R) podstatné jméno (N). To znamená, že kdykoli se v testované větě objeví dvojice slovních jednotek taková, že jedna z nich má slovní druh podstatné jméno a druhá má slovní druh předložka, bude testována tato podmínka (a všechny ostatní se stejnou deklarací).

Následuje výčet vlastností, které musí mít slovní jednotky s těmito slovními druhy, aby se jednalo o smysluplnou dvojici. Obecně je možné uvést řádek číslem morfologické značky a k oběma slovním druhům uvést požadované hodnoty. Zde druhý řádek vyžaduje přesné hodnoty slovních poddruhů (morfologická značka *číslo 2*), třetí říká, že podstatné jméno musí být ve třetím pádě (morfologická značka číslo 5). Na hodnotě morfologické značky číslo 5 u předložky nezáleží, proto ji není třeba udávat. Čtvrtý řádek určuje pořadí slovních jednotek (uvození klauzulí *ord*) s tím, že předložka musí být před podstatným jménem. Pátý se odkazuje na základní tvar předložky (uvození klauzulí *lemma*), který musí být roven jednomu z řetězců „k“, „proti“, „naproti“, „oproti“, „kvůli“, „díky“. Poslední řádek (klauzule *end*) ukončuje definici podmínky.

Této podmínce by vyhovovaly například dvojice „naproti škole“, “díky otci“, nebo „kvůli sestře“. Nevyhověly by například dvojice „naproti škola“ – nesouhlasí pád podstatného jména, „otci díky“ – nesouhlasí slovosled, nebo „před bratrovi“ – nesouhlasí lemma předložky.

Příklad rozboru jednoduché věty

Mějme větu: Do školy chodí malé děti.
Takto bude vypadat sestavený graf. Po kontrole souvislosti bude věta prohlášena za smysluplnou.



Každá hrana odpovídá jedné podmínce, které dvojice slov daná spojenými vrcholy dokázala vyhovět. Např. dvojice „do školy“ by vyhovovala podmínce podobné té v příkladu výše, jen pro podstatné jméno ve druhém pádě. Dvojici „malé děti“ – podstatné jméno, přídavné jméno - lze rozpoznat na základě shody v pádu, rodu, čísle atd.

Struktura programu

Aplikace bude využívat objektové vlastnosti C++, včetně sady knihoven STL – zejména budou využity kontejnery STL (map, vector, pair…). Mimo STL nejspíše další externí knihovny použity nebudou.
Základ aplikace bude tvořit jedna nadtřída - singleton, jejíž instance bude schopná větu zpracovat i analyzovat. Bude se starat o načtení a rozdělení vstupu, rozdělení souvětí na věty jednoduché, správu konfigurace a další úkoly globálního charakteru. Vlastní testování podmínek bude provádět samostatná třída. Tato načte a rozdělí soubor s podmínkami, uloží je do vhodných struktur (např. asociativní pole dvojice slovních druhů -> seznam podmínek pro tuto dvojici) a bude poskytovat metody pro kontrolu definovaných podmínek.

Literatura

- Popis nástroje *tool_chain*:
  Barbora Vidová Hladká a kol., Český akademický korpus 2.0, v tisku
  http://ufal.mff.cuni.cz/rest/CAC/cac_20.html
- Popis morfologických značek:
  http://ufal.mff.cuni.cz/rest/CAC/doc-cac20/cac-guide/cz/html/ch13.html