

Závěrečná zpráva

Filip

Pražák

Masarykova Univerzita, Fakulta informatiky

Výpočet pravděpodobností

Výpočet pravděpodobností

Jako svůj hlavní úkol jsem si vybral výpočet pravděpodobností pro rozdělení vět a pro šipky propojující jednotlivá slova. Nejdůležitější a přitom nejpracnější částí bylo vymyslet uložení získaných dat do takové podoby, aby s nimi kolega Šimon, který je poté dále zpracovával, měl co nejjednodušší práci. Jelikož data můžou narůstat, musely být uloženy do dynamické datové struktury, proto byl vytvořen list listů dvojrozměrných polí relation, ve kterém se nachází všechna potřebná data k propojení slov.

Třída Evaluation

Třída Evaluation

Pro zpřehlednění kódu a jednodušší dodržení OOP jsem vytvořil třídu Evaluation.java. Abych spočítal pravděpodobnost, rozhodl jsem se vytvořit tři metody, a to compareSentence(), compareRelation(), fillArray(). Metoda compareSentence() porovná rozložení jednotlivých částí vět od všech hodnotitelů. Pro každého hodnotitele se předtím vytvořil objekt ze třídy Sentence.java. Třída Sentence.java obsahuje atribut String[] sentence ve kterém je uloženo rozdělení věty podle daného ohodnotitele. Každá věta, která se nebude shodovat s předchozími rozděleními je vytvořena jako nová položka v listu listSentence. Tyto data budou sloužit pro vykreslení SVG obrázků. Metoda compareRelation() naplní list relation větami. Každá věta obsahuje list jednotlivých částí věty. Každá část věty má přiděleno dvojrozměrné pole, které se vyplní pomocí funkce fillArray() následovně: Pole má šířku 2 a výšku podle počtu ohodnotitelů. Hodnoty ve sloupci na pozici 0 určují, kam bude ukazovat šipka, a hodnoty ve sloupci na pozici 1 představují četnost výskytu jednotlivých šipek.

Závěr

Závěr

Společný projekt byl pro mě velkým přínosem. Posunul mě velmi dopředu a to v tom smyslu, že jsem se naučil pracovat s Gitem, což je skvělý nástroj pro práci v týmu a nabízí více možností, než například Subversion, který už mi teď nepříjde tolik efektivní. Dále jsem se zdokonalil v tom, že program, který nepíšu pouze pro sebe a tudíž data, se kterými bude dále někdo pracovat musí být poskytnuty v takové tvaru, aby je kolega, který tyto data chce začít využívat, rychle pochopil a byly pro něj v dostupném tvaru. Proto jsou vytvořeny například gettery, které práci zjednoduší a navíc ostatním zpřístupní vytvořená data.