

Projekt nr 1

Cezary Wolszczak
271100 gr.3

8 marca 2015

Spis treści

1	Opis ogólny	2
1.1	Nazwa programu	2
1.2	Opis programu	2
2	Opis funkcjonalności	2
2.1	Jak korzystać z programu	2
2.2	Uruchamianie programu	2
2.3	Możliwości programu	2
3	Format danych i struktura plików	3
3.1	Struktura katalogów	3
3.2	Przechowywanie danych w programie	3
3.3	Dane wejściowe	3
3.4	Dane wyjściowe	3
4	Scenariusz działania	4
4.1	Scenariusz działania programu	4
4.2	Ekran działania programu	4
5	Testowanie	4
5.1	Ogólny przebieg testowania	4

1 Opis ogólny

1.1 Nazwa programu

ATG – Automatic Text Generator

1.2 Opis programu

Program ma na celu pobranie jednego lub więcej plików tekstowych (w formacie txt) które następnie może generować na sposób wybrany przez użytkownika. Następnie uprzednio wygenerowany tekst zapisuje do pliku tekstowego (w formacie txt), po czym zwraca na wyjściu dane statystyczne.

2 Opis funkcjonalności

2.1 Jak korzystać z programu

Program będzie można uruchomić za pomocą komend w Terminalu poleceń systemów operacyjnych z rodziny Unix.

2.2 Uruchamianie programu

```
./ATG -p [plik.txt] -d [plik.txt] -m [plik.txt] -g -s/-a [liczba słów/akapitów]  
-n [n-gramy] -l [ilość n-gramów]
```

Legenda:

-p parametr potrzebny do pobrania do programu plików z tekstem bazowym (w formacie txt);
-g generuje tekst;
-d dopisanie danych do pliku pośredniego
-m pobiera plik pośredni (w formacie txt)
-s parametr służący do generacji tekstu o zadanej przez użytkownika liczbie słów;
-a parametr służący do generacji tekstu o zadanej przez użytkownika liczbie akapitów;
-n rząd n-gramu;
-l liczba wypisanych n-gramów;

2.3 Możliwości programu

- wczytywanie do programu kilku tekstów bazowych (teksty w formacie txt),
- generację tekstu na podstawie jednego lub kilku tekstów bazowych,

- generację tekstu o zadanej przez użytkownika liczbie słów, akapitów,
- generację tekstu w oparciu o n-gramy dowolnego rzędu,
- zapis wygenerowanego tekstu do pliku,
- generację statystyki tekstów wejściowych oraz tekstu wyjściowego (najczęściej wykorzystywane słowa, najczęściej powtarzające się n-gramy, PMI itd.),
- opracowanie mechanizmu przechowywania danych pośrednich (inicjacja bazy, import plików, oddzielne narzędzie/tryb do generowania tekstu o określonym rozmiarze).

3 Format danych i struktura plików

3.1 Struktura katalogów

Katalog zawierający program składa się z czterech podkatalogów:
 bin – zawiera się w nim plik wykonywalny,
 dane – przechowuje teksty bazowe,
 src – zawiera pliki z programem,
 gen – zawiera wygenerowane pliki tekstowe.

3.2 Przechowywanie danych w programie

Dane będą przechowywane w tablicy struktur. Każda ze struktur będzie przechowywała jedno słowo.

3.3 Dane wejściowe

Program będzie pobierał plik tekstowy z folderu "dane" za pomocą linii komend.

3.4 Dane wyjściowe

Program po pierwsze będzie zapisywał w folderze "gen" pliki tekstowe przegenerowane w wyniku działania programu. Następnie będzie wypisywał dane statystyczne w formacie .txt (najczęściej wykorzystywane słowa, najczęściej powtarzające się n-gramy, PMI (Pointwise Mutual Information)).

4 Scenariusz działania

4.1 Scenariusz działania programu

- Uruchomienie programu z linii poleceń. Sprawdzenie parametrów wywoływania, oraz w zależności od parametrów dalsza część działania programu (generacja nowego tekstu, tworzenie n-grafów, generacja poszczególnych zdań lub akapitów).
- Analiza tekstu bazowego.
- Generowanie tekstu wyjściowego.
- Tworzenie danych statystycznych.
- Zapis wygenerowanego tekstu oraz wypisanie danych statystycznych na wyjście.

4.2 Ekran działania programu

Program będzie uruchamiany i wykonywany tekstowo z poziomu Terminala.

5 Testowanie

5.1 Ogólny przebieg testowania

Program będzie testowany ręcznie przy użyciu danych tekstowych oraz przy użyciu debuggera "GNU debugger" (GDB).