

Sprawozdanie z projektów PJP

Eliza Petrycka, Dominika Bujnarowska

styczeń 2026

Streszczenie

Sprawozdanie opisuje dwa projekty z zakresu przetwarzania języka naturalnego: (1) statystyczną analizę korpusu języka portugalskiego wraz z wizualizacjami oraz (2) generator zdań w modelu SVO dla języka angielskiego. Przedstawiono cele, architekturę, główne funkcje, technologię oraz sposób uruchomienia.

1 Wprowadzenie

Celem prac było zaprojektowanie i implementacja rozwiązań PJP o różnym charakterze: analitycznym (statystyka korpusu i grafy językowe) oraz generatywnym (składanie zdań). Projekty realizują odpowiednio zadania 1 i 3 (analizy języka i grafy) oraz zadanie 2 (generator zdań).

2 Projekt 1 — statystyczna analiza języka portugalskiego

2.1 Cel i zakres

Projekt analizuje korpus języka portugalskiego oraz udostępnia wyniki w aplikacji webowej. Zakres obejmuje obliczenia statystyczne, prawo Zipfa oraz analizy grafowe relacji leksykalnych.

2.2 Główne funkcje

- tabela częstości wyrazów z filtrowaniem i sortowaniem,
- analiza prawa Zipfa z metrykami dopasowania,
- graf rdzenia języka (najśilniejsze powiązania słów),
- zestawienie 50 najczęstszych rzeczowników z tłumaczeniami,
- grafy dwudzielne: przymiotnik–rzeczownik oraz czasownik–rzeczownik.

2.3 Architektura i technologie

Rozwiązanie podzielono na backend i frontend:

- **Backend** — pipeline w Pythonie generujący pliki JSON z wynikami analiz.
- **Frontend** — aplikacja React (Vite) z Tailwind i DaisyUI do wizualizacji danych.

Dane wynikowe trafiają do katalogu `apps/backend/data/processed`, a następnie są kopiowane do `apps/frontend/src/data`, skąd korzysta UI.

2.4 Uruchomienie

Pipeline backendu uruchamiany jest poleceniem:

```
uv run python run_pipeline.py
```

Frontend uruchamiany jest standardowo poprzez `npm install` oraz `npm run dev`.

2.5 Rezultaty

Projekt dostarcza spójny zestaw analiz ilościowych i grafowych, pozwalając na eksplorację częstotliwości, zależności leksykalnych i struktury „rdzenia języka”. Wyniki są przedstawione w interaktywnym interfejsie webowym.

3 Projekt 2 — generator zdań (SVO)

3.1 Cel i zakres

Celem projektu jest generowanie poprawnych zdań w języku angielskim w schemacie Subject–Verb–Object. System uwzględnia liczbę, czas, tryb oraz dobór rodzajnika.

3.2 Główne funkcje

- składanie zdania na podstawie wybranych elementów S, V i O,
- obsługa czasów (m.in. present, past, future, perfect, continuous),
- dobór rodzajników i odmiany czasowników.

3.3 Architektura i technologie

Aplikacja została zbudowana w oparciu o Flask:

- `app.py` — routing i warstwa API,
- `engine.py` — logika składania zdań i odmiany,
- `data.py` — zasoby leksykalne i dane wejściowe,
- `templates/` oraz `static/` — interfejs webowy.

3.4 Uruchomienie

Najprostszy start dostępny jest przez Makefile:

```
make install
```

```
make run
```

Aplikacja działa pod adresem `http://127.0.0.1:5000`.

3.5 Przykład API

Endpoint POST `/generate` zwraca zdanie w języku angielskim na podstawie parametrów wejściowych, np. dla danych: `subject=cat`, `verb=eat`, `object=apple`, `tense=present`, `mode=affirmative` otrzymujemy: *The cat eats an apple*.

4 Podsumowanie

Oba projekty realizują komplementarne cele PJN: analiza statystyczna i grafowa korpusu oraz generowanie zdań według modelu składniowego. Pierwszy projekt koncentruje się na eksploracji i wizualizacji danych językowych, drugi — na regułowej generacji poprawnych zdań w języku angielskim. Razem stanowią spójny zestaw narzędzi do badań nad językiem naturalnym oraz prezentacji wyników w formie webowej.