# Eksploracja tekstów
# Pracownia 5
# Zajęcia ostatnie i termin dodatkowy

**Uwaga 1**: Zadania dotyczące odpowiadania na pytania można będzie potraktować jako alternatywę do egzaminu (szczegóły zostaną podane wkrótce).
**Uwaga 2**: Na zajęciach 15 będą przeprowadzone 2 konkursy: poprawiania literówek oraz zanurzania słów. W późniejszym terminie będzie konkurs odpowiadania na pytania

**Zadanie 1. (6p)** Zaproponuj sposób obliczania wektorowych zanurzeń słów, który dla słów z określonej dziedziny przypisuje podobne wektory. Twoja propozycja będzie oceniana na teście z wykładu (cos(a,a') > cos(a,b)), dla różnych a i a' z jednej klasy i b z innej klasy.

Plan minimum dla to przebicie zanurzeń z Wikipedii (będą dostępne na SKOS) o co najmniej 1 punkt procentowy na zbiorze `trudniejsze_klastry.txt`, ocenia program `basic_embedding_test.py`.

Możesz korzystać z następujących źródeł:

1. Wikipedia lub Wikipedyjka
2. N-gramy z NKJP,
3. Przesiane gramatycznie N-gramy z NKJP (będą na SKOSie)
4. Słowosieć (będzie wersja na SKOS-ie)

Z zadaniem związany jest Konkurs na Najlepsze Zanurzenia.

**Zadanie 2. (4p)** Wróćmy do zadania z synonimami z pierwszej listy, z analizą początków artykułów Wikipedii ukierunkowaną na znajdywanie synonimów[1]. Zmodyfikuj Twój program w ten sposób, by wyświetlał tylko synonimy zawierające pojedyncze słowa. Wykorzystaj zanurzenia słów (ze SKOS-u, lub z dowolnego innego źródła) jako dodatkowy filtr, wyświetlając jedynie te pary słów, dla których jednocześnie:

a) istnieje wzorzec regularny, pozwalający stwierdzić, że słowa *mogą* być synonimami,

b) wartości zanurzeń dla nich są wystarczająco podobne do siebie.

Warunki sprawdzane w punkcie a) powinny być nieco łagodniejsze od tych w zadaniu na pierwszej liście.

**Zadanie 3. (7p)** Dodaj do wyszukiwarki frazowej dla Wikipedii obsługę synonimów i wyrazów bliskoznacznych. W tym celu:

1. Wybierz zanurzenia dla form bazowych.
2. Zastosuj jakikolwiek algorytm podziału całego zbioru lematów na rozłączne podzbiory, w którym podobne wektory są w jednym klastrze. Algorytm może być wzięty z jakiejś biblioteki (na przykład algorytm k-średnich), może też być wymyślony przez Ciebie
3. Stwórz indeks frazowy, w którym termami są identyfikatory otrzymanych w poprzednim punkcie klastrów.

Uwaga: wyszukiwarka powinna zaznaczać znalezioną frazę (jest to dość kluczowe, bo w przeciwieństwie do poprzednich zadań, znalezienie tej frazy nie musi być wcale jednoznaczne).

**Zadanie 4. (6p+2p)** W zajmujemy się odpowiadaniem na pytania, zgodnie z następującym schematem:

a) Wybieramy jakiś typ pytania (na przykład pytania o rok, wiek, miasto, osobę, etc)

b) Znajdujemy wszystkie wystąpienia frazy tego typu w Wikipedii (lub jej podzbiorze).

[1] Jeżeli tego zadania nie robiłeś, możesz zrobić je teraz za 80% punktów

c) Definiujemy *kontekst* frazy. W kontekście powinny znaleźć się wyrazy w otoczeniu, można też uwzględnić końce zdań, akapitów, lub dowolnie heurystycznie modyfikować pojęcie kontekstu.

d) Indeksujemy konteksty (podobnie jak w przypadku wyszukiwania definicji, można to robić za pomocą wyrazów, lematów, zanurzeń, lub jakiejś ich kombinacji). Można też na przykład rozszerzyć kontekst o tytuł artykułu

e) W odpowiedzi na pytania tego wybranego typu wyszukiwać kontekstu podobnego do pytania.

Zadanie warte jest 6p dla wybranego typu pytań, i dodatkowo po jednym punktcie za drugi i trzeci typ.

**Zadanie 5. (3p)** Wykorzystaj w jakikolwiek sposób kategorie Wikipediowe w systemie odpowiadania na pytania.

**Zadanie 6. (4p)** Część pytań w turnieju 1 z 10 dotyczy przysłów. Dodaj do systemu odpowiadania na pytania tego typu (na SKOSie powinna pojawić się lista przysłów)

**Zadanie 7. (4p)** Dodaj do systemu odpowiadania na pytania możliwość wyszukiwania odpowiedzi w całej Wikipedii (być może tylko dla niektórych pytań, dla którym pełna Wikipedia daje lepsze działanie systemu)

**Zadanie 8. (7p)** $\star$ Wykorzystaj pretrenowane modele typu Transformer w systemie odpowiadania na pytania. Uwaga: będzie o tym więcej na wykładzie 9 czerwca.

**Zadanie 9. (1-5p)** $\star$ To jest zadanie bonusowe, które daje:

- 1p, jeżeli Twój system odpowiadania na pytania odpowie na 220-239 pytań
- 2p, jeżeli Twój system odpowiadania na pytania odpowie na 240-259 pytań
- 3p, jeżeli Twój system odpowiadania na pytania odpowie na 260-279 pytań
- 4p, jeżeli Twój system odpowiadania na pytania odpowie na 280-300 pytań
- 5p, jeżeli Twój system odpowiadania na pytania odpowie na ponad 300 pytań