

PUM 2025 — projekt 1. Wytyczne i zasady oceny.

1. Wymagania merytoryczne

Projekt polega na przeprowadzeniu klasyfikacji dźwięków z wybranej bazy danych, przy użyciu wybranego klasyfikatora, a następnie ocenie jakości modelu. Zakres wymagań na poszczególne oceny opisany jest poniżej — wymienione elementy powinny być zrealizowane w kodzie i zawarte w raporcie (wytyczne techniczne znajdują się na końcu dokumentu). Lista przykładowych baz danych znajduje się na upelu.

Skuteczność klasyfikacji nie będzie miała wpływu na ocenę — wyniki mogą być słabe, ale należy umieć je zinterpretować i prawidłowo przeprowadzić wszystkie etapy przygotowania danych i klasyfikacji oraz ocenę klasyfikatora. Nie trzeba ograniczać się do algorytmów omawianych na zajęciach.

Oceniany jest raport przesłany na platformę upel. Dodatkowo, na **czwartych zajęciach projektowych (15.12.2025)** każdy będzie prezentował swój projekt — można w ten sposób podnieść sobie ocenę o pół stopnia (jeśli prezentacja będzie dobra) lub o tyle samo obniżyć (jeśli prezentacja nie zostanie w ogóle wygłoszona).

Na ocenę 3.0:

1. Wstęp — krótki opis problemu badawczego: co będzie klasyfikowane, w jakim celu. Opis wykorzystywanej bazy danych:
 - nazwa i link, lub opis jak powstały dane;
 - typ danych — czy są to pliki audio, czy już przetworzone dane, co zawierają pliki;
 - ile jest klas i obiektów w każdej z nich.
2. Preprocessing: przygotowanie i weryfikacja danych (czy są brakujące dane, padding/przycięcie sygnału, w jakiej formie dane są podawane do klasyfikatora (zastosowana metoda reprezentacji sygnału), redukcja wymiarowości itd.).
3. Klasyfikacja wybraną metodą (trening + predykcja), z uzasadnieniem jej wyboru.
4. Optymalizacja wybranych hiperparametrów modelu (z wykorzystaniem pakietu `optuna` lub metodą `grid search/random search`).
5. Analiza wyników:
 - przedstawienie wyników klasyfikacji w formie macierzy pomyłek (z podaniem wartości progu klasyfikacji, jeśli był zastosowany);
 - wyznaczenie co najmniej dwóch wybranych metryk;
 - ocena jakości klasyfikatora na podstawie wybranej metryki — z uzasadnieniem, dlaczego akurat ta została zastosowana.
6. Podsumowanie:
 - co zostało zrobione;
 - jakie są wyniki i czy są zadowalające;
 - jak można je poprawić lub rozwinąć projekt.

Na ocenę 4.0:

1. Wstęp — krótki opis problemu badawczego: co będzie klasyfikowane, w jakim celu. Opis wykorzystywanej bazy danych:
 - nazwa i link, lub opis jak powstały dane;
 - typ danych — czy są to pliki audio, czy już przetworzone dane, co zawierają pliki;
 - ile jest klas i obiektów w każdej z nich.
2. Preprocessing:
 - przygotowanie i weryfikacja danych (czy są brakujące dane, padding/przycięcie sygnału);
 - jeżeli wybieramy jedną metodę klasyfikacji — zastosowanie dwóch wybranych metod reprezentacji sygnału (np. spekrogram, parametry czasowe openSMILE, MFCC itp.);
 - w jakiej formie dane są podawane do klasyfikatora (reshape, uśrednianie, inna metoda?), ew. redukcja wymiarowości itd.
3. Klasyfikacja — do wyboru:
 - (a) dwiema wybranymi metodami (trening + predykcja), z uzasadnieniem ich wyboru;
 - (b) dla dwóch metod reprezentacji sygnału, jedną wybraną metodą klasyfikacji, z uzasadnieniem jej wyboru.
4. Optymalizacja wybranych hiperparametrów modelu (z wykorzystaniem pakietu `optuna` lub metodą *grid search/random search*).
5. Analiza wyników:
 - przedstawienie wyników klasyfikacji w formie macierzy pomyłek (z podaniem wartości progu klasyfikacji, jeśli był zastosowany);
 - wyznaczenie co najmniej czterech wybranych metryk;
 - wskazanie, który/a z zastosowanych klasyfikatorów/metod reprezentacji sygnału jest lepszy/a, na podstawie wybranej metryki — z uzasadnieniem, dlaczego akurat ta została zastosowana.
6. Podsumowanie:
 - co zostało zrobione;
 - jakie są wyniki i czy są zadowalające;
 - jak można je poprawić lub rozwinąć projekt.

Na ocenę 5.0:

Przeprowadzenie klasyfikacji na plikach audio:

1. Wstęp — krótki opis problemu badawczego: co będzie klasyfikowane, w jakim celu. Opis wykorzystywanej bazy danych:
 - nazwa i link, lub opis jak powstały dane; opis, co zawierają pliki;
 - ile jest klas i obiektów w każdej z nich.
2. Preprocessing:
 - przygotowanie i weryfikacja danych (czy są brakujące dane, padding/przycięcie sygnału);
 - zastosowanie dwóch wybranych metod reprezentacji sygnału (np. spekrogram, parametry czasowe openSMILE, MFCC itp.).

- w jakiej formie dane są podawane do klasyfikatora (reshape, uśrednianie, inna metoda?), ew. redukcja wymiarowości itd.

3. Klasyfikacja dwiema metodami (trening + predykcja), z uzasadnieniem ich wyboru.
4. Optymalizacja wybranych hiperparametrów modelu (z wykorzystaniem pakietu `optuna` lub metodą *grid search/random search*).
5. Analiza wyników:
 - przedstawienie wyników klasyfikacji w formie macierzy pomyłek (z podaniem wartości progu klasyfikacji, jeśli był zastosowany);
 - wyznaczenie co najmniej czterech wybranych metryk;
 - wskazanie, która z zastosowanych kombinacji parametry-klasyfikator jest najlepsza, na podstawie wybranej metryki — z uzasadnieniem, dlaczego akurat ta została zastosowana.
6. Podsumowanie:
 - co zostało zrobione;
 - jakie są wyniki i czy są zadowalające;
 - jak można je poprawić lub rozwinąć projekt.

2. Wytyczne techniczne

Można przygotować projekt w formie **notatnika Jupyter**, podobnie jak pracowaliśmy na zajęciach. Wówczas wszelkie opisy (wstęp oraz opis kolejnych kroków) powinny znaleźć się w komórkach tekstowych (nie zawierających kodu wykonywalnego) — komentarze w kodzie oczywiście mogą być, ale nie powinny być konieczne do prześledzenia postępowania. Komórki tekstowe korzystają z języka **markdown**; ściągawkę z jego składni można znaleźć na stronie <https://www.markdownguide.org/cheat-sheet/>.

Alternatywnie, można przygotować projekt w formie raportu w **pliku pdf** — wówczas należy dołączyć plik(i) z kodem w **formacie .ipynb lub .py** z komentarzami ułatwiającymi poruszanie się po kodzie, lub osobny plik README (jeśli plików jest więcej i nie mają samowyjaśniających się nazw).

Jeśli nie jest to konieczne (ze względu na rozmiar), proszę **nie pakować** plików do archiwum, tylko załadować je pojedynczo na platformę upel.