Zadanie: Klasyfikacja nowotworów na podstawie mutacji somatycznych i ekspresji RNA

Wstep

Ten projekt ma na celu zapoznanie z procesem budowy modelu klasyfikacyjnego nowotworów wykorzystujacego dane o mutacjach somatycznych oraz profil ekspresji RNA. W trakcie realizacji zadania:

- Pobierzemy i przetworzymy surowe dane genomowe z baz takich jak TCGA (z Biquery lab1) i adnotacje genów z OncoKB,
- przeprowadzimy inżynierie cech mutacyjnych i transkryptomicznych,
- zaimplementujemy model sieci neuronowej w PyTorch,
- rozwiażemy problem niezbalansowanych klas i zmniejszymy ryzyko overfittingu,
- zaproponujemy i przetestujemy metody poprawy jakości modelu.

1. Przygotowanie danych

1. **Filtracja genów onkogennych**
   - Z OncoKB™ https://www.oncokb.org/ wybierz liste genów Vogelstein et al. (2013)).
2. **Pobranie surowych mutacji**
   - Wyciagnij z bazy TCGA (The Cancer Genome Atlas) tylko rekordy somatycznych mutacji (SNP, DEL, INS) dla genów zwiazanych z oncokb.
3. **Format Parquet**
   - Zapisz oczyszczone dane mutacji w formacie Parquet.
   - Pobierz profil ekspresji RNA (TCGA RNA-seq), usuń duplikaty i zapisz w Parquet.
4. **Charakterystyka etykiet**
   - Zbiór etykiet `primary_site` jest niezbalansowany – różne klasy maja różna liczebność.

2. Inżynieria cech

1. **Mutacje**
   - Ekstrakcja trójnukleotydowego kontekstu → `Mutational_Motif`.
   - Normalizacja do pirymidyn, komplementacja.
   - Binowanie genomowe co 1,Mb → `Genomic_Bin`.
   - Tokenizacja genów, motywów i binów → funkcja `to_matrix()`.
2. **Ekspresja RNA**
   - Normaliacja poprzez logarytmowanie (`log1p`).
3. **Konkatenacja**
   - Połacz wektory: $X_{bin} \parallel X_{gene} \parallel X_{motif} \parallel X_{rna}$.
   - Przygotuj etykiety: y_primary_site.

3. Budowa i trening modelu

1. **Architektura**
   - Wejścia dla każdej grupy cech (warstwy `Dense`).
   - Łaczenie (`Concatenate`) oraz warstwy ukryte.
   - Wyjście: `softmax` nad klasami lokalizacji pierwotnej.
2. **Trening**
   - Podział na zbiór treningowy/testowy 80/20.
   - Optymalizator: Adam, strata: `categorical_crossentropy`.
   - Monitorowanie *loss* i *accuracy*.
   - Uwzglednienie niezbalansowanych etykiet.

4. Ocena overfittingu

1. **Wizualizacja uczenia**
   - Krzywe *loss*/*accuracy* dla zbioru treningowego i walidacyjnego.
2. **Metryki**
   - Accuracy, precision, recall, F1 na zbiorze walidacyjnym.
3. **Identyfikacja overfittingu**
   - Nierozbieżność miedzy strata/train a strata/val.
   - Spadek lub stagnacja *accuracy*/val przy rosnacym *accuracy*/train.

5. Propozycje poprawy

- **Regularizacja:** Dropout, L2-weight decay.
- **Zmiana architektury:** mniejsza sieć, BatchNormalization.
- **Optymalizacja hiperparametrów:** liczba warstw, rozmiar warstw, learning rate.

Ocena projektu

Projekt oceniany jest w skali 0–10 punktów. Poniżej rozkład punktów za poszczególne etapy:

- Przygotowanie i preprocesing danych (pobranie mutacji TCGA, filtracja OncoKB, zapis Parquet) – 2 pkt
- Inżynieria cech mutacyjnych (motywy, normalizacja, binowanie, tokenizacja) – 2 pkt
- Inżynieria cech ekspresji RNA (logarytmowanie) – 1 pkt
- Konkatenacja wektorów cech i przygotowanie etykiet – 1 pkt
- Budowa i implementacja modelu (architektura, kompilacja) – 1 pkt
- Trening modelu i walidacja (podział danych, *class_weight/oversampling*) – 1 pkt
- Analiza overfittingu i wizualizacje (krzywe loss/accuracy, metryki) – 1 pkt
- Propozycje ulepszeń i refleksja nad rezultatami – 1 pkt

Forma oddawania rozwiazań

Notatniki z rozwiazaniami wraz z krótkim raportem (jedna strona A4) podsumowujacy głowne wyniki projektu należy przesłać mailowo na adres: **m.wierzbinski@uw.edu.pl**. Dla ułatwienia prośba o nadanie tytułu: **[BiolSys] Zadanie zaliczeniowe multiomics**. Dodatkowo, notatnik i raport wrzucamy na osobisty GitHub i dorzucamy link do repozytorium do Google Sheets z nasza baza projektowa.

Kontakt i terminy

Kontakt: Pytania i watpliwości można kierować do twórcy: *[Marcin Wierzbiński], e-mail: m.wierzbinski@uw.edu.pl.*
Deadline: 18 lipca 2025, godz. 20:00.