Projekt WIMU 25Z

Przeglądarka embeddingów audio

Analiza Literaturowa

Zespół nr 6

Rafał Kuśmierz
Jakub Sałek
Aleksander Gryzik

Prowadzący projekt

mgr inż. Tomasz Radzikowski

CLAP

Elizalde, B., Deshmukh, S., Al Ismail, M., & Wang, H. (2022). *CLAP: Learning audio concepts from natural language supervision*. Microsoft Research. Pobrano z CLAP: https://arxiv.org/abs/2206.04769

CLAP (Contrastive Language–Audio Pretraining) to dwumodalny model (enkoder audio + enkoder tekstu) uczony kontrastywnie tak, by odwzorowywać dźwięki i opisy w wspólną przestrzeń embeddingów. Autorzy trenują CLAP na 128 010 parach audio–tekst z FSD50k, ClothoV2, AudioCaps i MACS, a następnie ewaluują na 16 zadaniach w 8 domenach (m.in. zdarzenia akustyczne, muzyka, mowa). Model ustanawia SoTA w trybie zero-shot oraz SoTA w 5 zadaniach przy uczeniu nadzorowanym. Wnioskowanie opiera się na podobieństwie kosinusowym między embeddingami audio i tekstu. CLAP dostarcza gotową przestrzeń audio - tekst do retrievalu i porównań podobieństwa bez konieczności finetuningu.

Hugging Face Transformers. CLAP model documentation
https://huggingface.co/docs/transformers/model_doc/clap

Dokumentacja prezentuje gotowy interfejs do pracy z CLAP-em. W praktyce korzysta się z dwóch elementów: ClapProcessor, który przygotowuje dane (ekstrakcja cech z audio i tokenizacja tekstu), oraz ClapModel, który zwraca reprezentacje wektorowe. Model może jednocześnie generować embeddingi audio jak i tekstowe do bezpośredniego porównywania audio - tekst. Checkpointy ładuje się standardową metodą from_pretrained, co umożliwia szybkie uruchomienie gotowego modelu. Dla dłuższych

nagrań przewidziano tryb fusion, w którym procesor dzieli sygnał na segmenty i łączy ich reprezentacje w jeden spójny embedding. Z punktu widzenia naszego projektu oznacza to powtarzalny, prosty pipeline do ekstrakcji embeddingów i ewaluacji podobieństwa, bez konieczności trenowania modeli od podstaw. Mamy również zapewnioną pełną integrację z narzędziami HF.

SLAP

Guinot, J., Riou, A., Quinton, E., & Fazekas, G. (2025). *SLAP: Siamese Language–Audio Pretraining without Negative Samples for Music Understanding*. W *Proceedings of the 26th International Society for Music Information Retrieval Conference (ISMIR 2025), Daejeon, South Korea*. https://arxiv.org/abs/2506.17815
Pliploop. (b.d.). *SLAP: Siamese Language–Audio Pretraining*. Repozytorium GitHub https://github.com/Pliploop/SLAP

Artykuł przedstawia model SLAP (Siamese Language–Audio Pretraining), który rozwija koncepcję znaną z modelu CLAP (Contrastive Language–Audio Pretraining), czyli wspólne uczenie modeli tekstowych i dźwiękowych w jednej przestrzeni reprezentacji.

Autorzy modelu proponują nowy sposób uczenia relacji między dźwiękiem a tekstem, oparty na podejściu Bootstrap Your Own Latent (BYOL). Jest to metoda uczenia samonadzorowanego (*self-supervised learning),* w której model uczy się dopasowując swoje przewidywania do wyników drugiego, powoli aktualizowanego modelu, tzw. *target encodera*. Oba modele przetwarzają różne przekształcone wersje tych samych danych, a sieć predykcyjna uczy się przekształcać reprezentację jednego modelu tak, aby jak najlepiej odpowiadała reprezentacji drugiego.

W SLAP autorzy całkowicie zrezygnowali z użycia negatywnych przykładów, dzięki temu model można trenować bez konieczności wykorzystywania bardzo dużych batchy, co znacząco poprawia wydajność i skalowalność w porównaniu z CLAP.

Opisany model wykorzystuje architekturę typu siamese, w której każda z dwóch części (dla dźwięku i dla tekstu) ma własny moduł predykcyjny oraz enkoder aktualizowany metodą EMA (Exponential Moving Average). Enkoder audio oparty jest na modelu HTS-AT (Hierarchical Token-Semantic Audio Transformer), natomiast enkoder tekstowy wykorzystuje model RoBERTa (Robustly Optimized BERT Pretraining Approach).

W przeprowadzonych eksperymentach autorzy trenowali model na prywatnym zbiorze PrivateCaps, zawierającym około 260 tysięcy par danych składających się z pełnych utworów muzycznych oraz odpowiadających im opisów tekstowych.

Do oceny modelu wykorzystano kilka publicznych zbiorów danych. W zadaniach polegających na wyszukiwaniu utworu na podstawie tekstu (*text–music retrieval*) zastosowano MusicCaps oraz Song Describer Dataset, natomiast do testów klasyfikacyjnych i analizy jakości reprezentacji (*zero-shot classification* i *downstream probing*) użyto zbiorów GTZAN (klasyfikacja gatunków muzycznych), MagnaTagATune (MTAT) (automatyczne tagowanie muzyki) oraz OpenMic (rozpoznawanie instrumentów).

Wyniki eksperymentów pokazują, że SLAP osiąga lepsze wyniki niż CLAP w zadaniach wyszukiwania muzyki na podstawie tekstu (*text–music retrieval*), jak i w testach klasyfikacyjnych (*zero-shot classification*) oraz analizie jakości reprezentacji (*downstream probing*). Autorzy wskazują również na zmniejszenie tzw. *modality gap*, czyli różnicy między przestrzeniami reprezentacji tekstu i dźwięku oraz na większą stabilność i odporność modelu na zmiany wielkości batcha w trakcie uczenia.

W celu wykorzystania modelu w projekcie skontaktowano się z autorami z prośbą o udostępnienie wag wytrenowanego modelu opisanego w artykule. Niestety autorzy nie zamierzają udostępniać publicznie wag w najbliższej przyszłości. Brak gotowych wag równoznaczny jest z koniecznością samodzielnego wytrenowania modelu, co jest procesem czasochłonnym, a przede wszystkim wymaga odpowiednich zasobów obliczeniowych. Z tych powodów SLAP nie zostanie wykorzystany w projekcie.

Tovstogan, P., Serra, X., & Bogdanov, D. (2022). Visualization of deep audio embeddings for music exploration and rediscovery. W Proceedings of the 19th Sound and Music Computing Conference (SMC 2022) (ss. 108–115). Music Technology Group, Universitat Pompeu Fabra. Pobrano z https://repositori-api.upf.edu/api/core/bitstreams/9eed935a-aa7f-4cf6-a803-103389d66369/content

Autorzy prezentują webowy interfejs do wizualizacji osobistych kolekcji muzycznych na podstawie głębokich embeddingów z auto-taggerów (MusiCNN, VGG; trenowane na MSD i MTAT), wykorzystując zarówno taggramy (warstwa wyjściowa, 50D), jak i embeddingi przedostatniej warstwy (200D/256D). Audio jest dzielone na segmenty ~3 s, a przestrzenie reprezentacji odwzorowywane do 2D różnymi metodami (PCA, STD-PCA, t-SNE, UMAP) w dwóch skoordynowanych widokach, z możliwością zaznaczania i porównywania tych samych segmentów między projekcjami. W badaniu z 8 uczestnikami interfejs okazał się użyteczny do eksploracji, tworzenia playlist i „redyskrypcji"; użytkownicy częściej wskazywali UMAP/t-SNE jako lepsze do struktury lokalnej, a PCA/STD-PCA jako szybsze i czytelniejsze w skali globalnej. Dla naszego projektu praca stanowi bezpośrednie uzasadnienie użycia PCA/t-SNE/UMAP do oceny i

porównywania przestrzeni embeddingów (także między modelami), oraz inspirację dla interaktywnych widoków do analizy podobieństw.

OpenL3

Cramer et al., ICASSP 2019 – Look, Listen and Learn More: Design Choices for Deep Audio Embeddings
https://www.justinsalamon.com/uploads/4/3/9/4/4394963/cramer_looklistenlearnmore_icassp_2019.pdf

Wielomodalna sieć neuronowa L3-Net (Look, Listen and Learn) została zaprojektowana do uczenia się powiązań między dźwiękiem a obrazem w trybie samonadzorowanym. Model opiera się na zadaniu Audio-Visual Correspondence (AVC), którego celem jest określenie, czy dana klatka wideo oraz odpowiadający jej 1-sekundowy fragment audio pochodzą z tego samego źródła i zachodzą na siebie w czasie.

Architektura L3-Net składa się z trzech głównych modułów:

- Podsieć audio – przyjmuje na wejście spektrogram audio i generuje jego reprezentację wektorową (embedding).
- Podsieć wideo – przetwarza pojedynczą klatkę obrazu, zwracając embedding wizualny.
- Podsieć fuzji – łączy embeddingi audio i wideo, aby określić ich zgodność i wyznaczyć prawdopodobieństwo poprawnego dopasowania.

Model umożliwia również uzyskanie embeddingów dla pojedynczej modalności, co jest szczególnie istotne w kontekście przygotowywanego przez nas projektu. Podsieć audio akceptuje jako wejście: spektrogram liniowy lub melspektrogram z 128 lub 256 binami. Wyjściem jest macierz embeddingów, w której każdy wiersz odpowiada kolejnemu oknu audio. Dodatkowo generowany jest wektor timestampów wskazujący środek każdego okna. Pojedynczy embedding posiada 6144 cechy, co pozwala na uchwycenie szczegółowych cech akustycznych.

Sieć była trenowana na zbiorze AudioSet, opracowanym przez Google Research, zawierającym około 2 milionów filmów z YouTube, z dźwiękiem w jakości 48 kHz oraz wideo w 30 klatkach na sekundę.

Do implementacji wykorzystano bibliotekę OpenL3, która pozwala łatwo generować embeddingi audio lub wideo i wykorzystywać je w zadaniach downstream z wykorzystaniem języka Python (https://github.com/marl/openl3.git).

MERT

Li et al., 2023/2024 – Acoustic Music Understanding Model
https://arxiv.org/html/2306.00107v5

MERT to model oparty na architekturze transformera z wstępną warstwą konwolucyjną, która działa jako ekstraktor cech audio przed warstwami transformera. Dzięki temu model potrafi efektywnie wydobywać lokalne wzorce akustyczne z sygnału audio przed przetwarzaniem sekwencyjnym.

Model trenuje się z wykorzystaniem MLM (masked language modelling), czyli fragmenty sygnału audio są zakrywane, a zadaniem modelu jest przewidzenie brakujących fragmentów na podstawie kontekstu audio. Takie podejście umożliwia modelowi uczenie się głębszych zależności czasowych i harmonicznych w sygnale, bez potrzeby ręcznego oznaczania danych.

Dodatkowo MERT korzysta ze strategii teacher-student, gdzie teacher models służą jako źródło wiedzy. Student uczy się nie bezpośrednio na etykietach, lecz na predykcjach modelu nauczyciela. W MERT wykorzystywane są dwa typy teacherów:

- Acoustic teacher – generuje dyskretne kody akustyczne, czyli reprezentacje opisujące sygnał na poziomie akustycznym.
- Musical teacher – ekstrahuje cechy harmoniczne i tonalne, wspierając model w rozumieniu struktury muzycznej.

Wejściem do modelu jest audio próbkowane z częstotliwością 24 kHz, dzielone na 5-sekundowe fragmenty. Dla każdego segmentu model zwraca wektor stanu ukrytego w przestrzeni o wymiarach 512 lub 1024, który może być wykorzystany do downstream tasks.

Takie połączenie konwolucyjnej ekstrakcji cech, transformera oraz uczenia z teacher models pozwala MERT na efektywne przenoszenie wiedzy z danych akustycznych i muzycznych, stabilizuje proces uczenia i zwiększa dokładność modelu na różnych zadaniach związanych z audio i muzyką.