Tohle je komunikace kluků se Stefani. Občas si taky mailují po půlnoci. V klidu si to počtěte. Nevím jak Vy, ale já se definitivně ztratil někdy už u bodu č. 2.

1) Kuba:

Ahoj Stefanie,

Momentálně rozděluji větší nahrávky na části, které odpovídají délce popěvku strnadů.

Dalším krokem je podle mě začít experimentovat s trénováním CNN (nebo podobného) pro klasifikaci našich spektrogramů a vyzkoušet augmentaci dat na některých méně běžných dialektech.

Jakub

##

2) Zdenda:

Ahoj Stephanie,

včera večer nás napadla taková myšlenka. Co kdybychom trénovali CNN s klasifikátorem, aby při trénování generoval ztrátu, a pro produkci bychom klasifikátor odstranili a vytvořili pouze vektor, který by se dal projít klastrovými algoritmy?

Trénování by vypadalo nějak takto: Vstup -> ResNet -> Hustá neuronová síť (redukce dimenze) -> Klasifikátor

Doba běhu by byla stejná, jen by klasifikátor byl nahrazen K-průměry nebo něčím podobným, abychom byli schopni (doufejme vizuálně) detekovat nové dialekty, aniž bychom jim dávali konkrétní, nesprávné labely s nízkou pravděpodobností.

Zdeněk

##

3) Stefanie:

...dobrý a velmi chytrý nápad! :-)

Existuje několik příkladů, kde by se dalo ukázat, že něco takového funguje dobře.

Základní myšlenkou je vzít část předzpracování CNN a použít vektor příznaků pro jiný algoritmus jako vstup pro klasifikaci. Jedinou podmínkou je, že vektor příznaků nesmí být příliš vysokodimenzionální... Dimenzionalita je samozřejmě jakýmsi metaparametrem finálního systému shlukování, který je třeba upravit...

Mohli bychom také zvážit umístění automatického kodéru na finální (vysokodimenzionální) vektor příznaků, aby se generoval ještě menší latentní prostor.

##

4) Zdenda:

Ahoj Stephanie,

Ano, doufáme, že se nám nějak podaří dosáhnout 2/3/4D vektorového prostoru, to by bylo úžasné, protože vizualizace těchto prvků je snadná nebo alespoň možná.

Myšlenka automatického kodéru je pravděpodobně to, co jsem myslel tou redukcí dimenzí.

Jednou z našich obav je, zda je možné po trénování „vytrhnout“ vrstvy (hustou klasifikační neuronovou síť) z finálního souboru modelu…?

Zdeněk

##

5) Stefanie:

To vlastně není problém, protože většina vestavěných verzí CNN umožňuje oddělit část modelu pro předzpracování od části „klasifikátoru“.

A několik dalších nápadů:

-> předzpracování by mohlo také generovat logaritmické mel spektrogramy… (Pro ptačí zpěv by to mohlo být lepší než standard)

-> vision transformer by mohl být také alternativou k CNN

-> pro trénování sémanticky smysluplných reprezentací sonogramů bychom mohli také použít sítě „Učitel-Žák“

Princip:

Máte dvě neuronové sítě, které se dívají na různé verze stejného vstupu – například dva mírně odlišné spektrogramy stejného ptačího zpěvu (z augmentace dat).

*** Síť Učitel vidí verzi A.**

*** Síť Žák vidí verzi B.**

*** Žák se snaží předpovědět nebo shodovat se s tím, co vidí Učitel.**

Postupem času se Žák naučí:

„I když zvuk vypadá trochu jinak, stále znamená totéž.“ => podobné vkládání

DINO = DESTILACE BEZ labelů

Klastrování lze nakonec provést pomocí HDBSCAN na vkládacím prostoru :-) Protože alespoň pro některé příklady máme labely, mohli bychom výsledek následně ověřit…

Podívám se, jestli najdu nějaký příklad kódu (měli jsme před časem studentský projekt a doufejme, že bych měl někde mít jejich implementaci :-))

##

6) Zdeněk:

Ahoj Stephanie,

Právě jsem se podíval na vision transformers a vypadají skvěle! Ale jaké jsou klíčové rozdíly mezi nimi a CNN? Z toho, co jsem zjistil, je architektura vision transformers podobná architektuře LLM, jen s pixely místo slov… Na druhou stranu, nejsou CNN navrženy k detekci vzorů? Dokážu si představit, že vision transformery by se daly použít k získání „významu“ z obrázku, ale je to pro klasifikátor potřeba?

=> většina vestavěných verzí CNN umožňuje oddělit část modelu pro předzpracování od části „klasifikátoru“.

- To je skvělá zpráva

=> předzpracování by mohlo také generovat logaritmické mel spektrogramy…

- Plánujeme to udělat, Keras má dokonce vrstvu, která to přesně udělá, takže to nemusíme dělat sami – to je super.

=> model žák-učitel

- Nevyžaduje to mít jeden již vytvořený a funkční klasifikátor? Opravte mě, pokud se mýlím, ale já to vidím jako další trénování již vytvořené sítě a zároveň možné snížení celkového počtu parametrů aproximací výsledku, který by produkovala větší síť.

##

7) Stefanie

->Vision transformers:

Máte naprostou pravdu :-) — Vision transformery fungují hodně podobně jako jazykové modely: rozdělí obraz na malé oblasti (jako jsou slova) a pomocí sebepozornosti se učí, jaké existují vztahy mezi jednotlivými oblastmi.

CNN se naopak zaměřují na lokální vzory: posouvají filtry po obrazu a krok za krokem se učí hrany, textury a tvary. CNN jsou tedy detektory vzorů, zatímco vision transformery se učí vztahům. CNN mají silné vestavěné předpoklady o lokalitě, což pomáhá, když máte omezená data. Vision transformery mají méně předpokladů a dokáží zachytit globálnější význam, ale obvykle potřebují více dat nebo předběžné trénování.

Pro náš projekt s ptačím zpěvem: CNN jsou skvělé pro rozpoznávání typických frekvenčních tvarů, ale ViT by mohly lépe zachytit celkovou strukturu písně – užitečné, když chceme smysluplná vnoření spíše než jen klasifikaci.

->model žák-učitel:

Dobrý komentář! A to platí i pro klasickou destilaci znalostí: předem proškolená síť učitelů vede menšího studenta k napodobování jejích předpovědí.

Ale u samoproškolené destilace (jako je DINO nebo BYOL) je to jiné – neexistuje žádný předem proškolený učitel. Učitel je jednoduše exponenciálně zprůměrovaná kopie samotného studenta. Student se učí porovnávat výstup učitele na různých „pohledech“ (rozšířeních) stejného vstupu. Takže místo přenosu znalostí z existujícího klasifikátoru se model sám učí konzistenci – učí se stabilní, smysluplné reprezentace bez jakýchkoli popisků.

##

8) Zdeněk

Ve vědeckém článku o nahrávání netopýrů (co jste mi poslala) jsem četl, že podporují detekci v reálném čase se streamováním do modelu, což mi přišlo velmi zajímavé a skvělé.

Co kdybychom udělali toto: vstup -> CNN (nahrazení birdnet pro nalezení přítomnosti strnada, protože spuštění plného modelu birdnet je pomalé a výpočetně náročné, nepoužitelné pro reálný čas) -> Vision Transformer (pro převod dialektu do vektoru) -> Redukce dimenzí + autoencoder -> klasifikátor husté neuronové sítě (odstraněno pro produkční účely)

Co si o tom myslíte?