Semestrální práce z KKY/HDS

Tvorba neurálního modelu svého hlasu

Dominik Zappe – A23N0011P
(zapped99@students.zcu.cz)

Duben 2024

Obsah

1 Nahrávání hlasu

V rámci semestrální práce bylo nahráno 1 024 vět za účelem lepší, srozumitelnější syntézy a za účelem bonusových bodů. Výsledky syntézy nejsou však velmi uspokojivé a zpětně si myslím, že by menší úsilí při natáčí docílilo stejného výsledku.

Při nahrávání vět jsem se osobně potýkal často s neresponzivní webovou aplikací – i přes to však webová aplikace velmi usnadnila proces nahrávání svého hlasu. Kdybych však předem tušil dosažené výsledky, nejspíše bych se tolik nesnažil natočit co nejvíce vět, abych měl syntézu „kvalitní“.

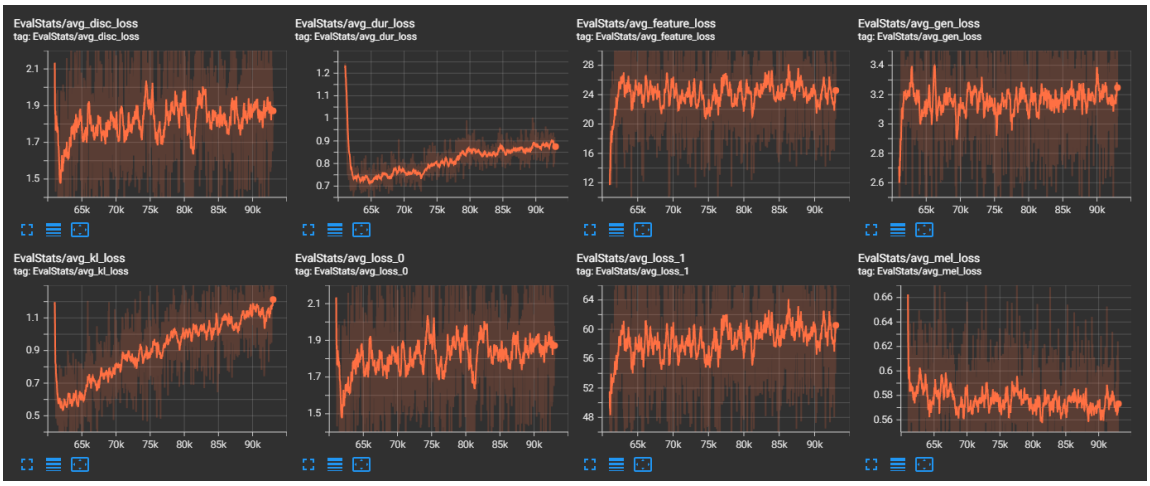
2 Implementace ztrátových funkcí

Osobně mi trvalo nepřiměřeně dlouho docílit implementace ztrátových funkcí, které by nezpůsobovaly šum jako výsledek testovacích nahrávek. Když už se mi povedlo implementovat takové ztrátové funkce, které nezpůsobovaly šum, docílil jsem srozumitelné robotické řeči. Bylo to způsobeno tím, že v dodaném kódu bylo naznačeno, jak si zavolat funkci diskriminátoru, ale nikde nebylo zmíněno, že by se ve větvi pro generátor nemělo užívat funkce *detach* – tedy ztrátové funkce *generator loss* a *feature loss* byly úplně zbytečné, jelikož byly od sítě odpojené. Po správném užití funkcí *detach* se docílilo hlasu, který více připomíná hlas autora, avšak není mu velmi dobře rozumět. Stále však zůstává pár nesrovnalostí, na které bohužel nebylo zodpovězeno ani v e-mailu, který jsem psal, viz později.

- Mám podezření, že výsledky diskriminátoru jsou prohozené, protože po nahlédnutí do zdrojového kódu VITS modelu diskriminátor vrací x a x_hat – v tomto pořadí. „x (Tensor): ground truth waveform.“ a „x_hat (Tensor): predicted waveform.“, viz Coqui-TTS/TTS/tts/layers/vits/discriminator.py. V dodaném ukázkovém kódu se však návratové hodnoty z funkce *disc* berou v opačném pořadí. Zajímavé je, že výsledky mi přišly u obou možností podobné, takže nevím, která varianta je správná.
- Další nesrovnalost je v popisu ztrátové funkce *MEL loss* – „jako ztrátovovou funkci lze použít střední hodnotu rozdílů absolutních hodnot“ – neměla to spíše být střední hodnota absolutní hodnoty rozdílu?
- Poslední nesrovnalost, kterou jsem zmiňoval v e-mailu, ale nebyla zodpovězena ani jedním vyučujícím – skóre vrácena diskriminátorem jsou podle návodu pravděpodobnosti, po vypsání obsahu tenzorů jsem tam však viděl hodnoty menší než 0 a větší než 1. Nejedná se tedy o pravděpodobnost a ručně jsem si všechna skóre nechal projít *sigmoidou* – předpokládám, že záporná hodnota značí *fake*,

kladná *real* a nula „*nevím*“. Sigmoida tyto hodnoty efektivně převede na pravděpodobnost, kde nula značí *fake*, jednička *real* a 0.5 „*nevím*“. Bohužel komunikace ze stran vyučujících zde selhala, a tedy stále nevím, jestli špatně diskriminátor používám, nebo jestli byla špatně popsaná interpretace jeho hodnot, ale kvůli této skutečnosti potom nedávaly smysl ztrátové funkce pro diskriminátor a generátor, kde se daná skóre např. odečítají od jedničky (smysl pouze pro pravděpodobnost, ale pro hodnoty -1.6 až 1.8, co jsem v tom tenzoru viděl, to smysl nemá). Po použití sigmody byl později zpozorován lepší průběh ztrátových funkcí diskriminátoru a generátoru v tensorboardu (bez sigmoidy se ztráta generátoru chovala „logaritmicky“ a kolísala okolo hodnoty 6, zatímco diskriminátor „exponencielně“ klesal k hodnotě 0). Se sigmoidou docházelo k ekvilibriu mezi generátorem a diskriminátorem, ale na výsledné nahrávky to zdá se nemá velký vliv (nejspíše kvůli obrovské váze *MEL loss*).

Finální průběh ztrátových funkcí je vidět na Obrázku 1. Bohužel jsem nedokázal průběhy zlepšit, osobně si myslím, že nejsou ideální.



Obrázek 1: Průběh ztrátových funkcí

3 Trénování neuronové sítě

Osobně mi trvalo poměrně dlouhou dobu vůbec rozjet dodaný Jupyter notebook, protože v něm byla zavedená špatná konfigurace a další chyby.

Vlastní trénování neuronové sítě pak již probíhalo bez problémů.

Bylo vyzkoušeno celkem 25 různých modelů, kde přibližně 20 z nich už nějak mluvilo (nebo se alespoň snažilo mluvu připomínat). Testovány byly zejména různé váhy různých ztrátových funkcí, nejnadějnější pokus pro mě osobně byly váhy – 54 MEL loss, 5 KL loss, 2 Feature loss. Avšak základní váhy, tedy – 45 MEL loss a 5 KL loss – dosahují podobných, spíše i lepších výsledků – proto je odevzdán tento model.

4 Závěr

Výstupem semestrální práce je natrénovaný model na vlastním hlase, kterému však není příliš dobře rozumět. Když je dodán i text, který je syntetizován, tak mu rozumět je lépe, neboť člověk slyší, to co slyšet má, ale bez znalosti, co se model snaží říct mu příliš rozumět není.

Osobně pro mě byla semestrální práce celkem náročná, neboť zorientovat se v tak velkém projektu bez velkých zkušeností s knihovnou *torch* bylo téměř nereálné. Napsané návody tomu bohužel příliš nepomohly.

Vyučující údajně vše testovali, a vše u nich fungovalo bez problémů – má osobní zkušenost však je, že na poprvé byl problém snad se vším.