

دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده برق و کامپیوتر



# Deep Generative Models

## تمرین شماره ۳

طراحان:

محمد جواد مؤمنی

حسین آزاد ملکی

نیکو نقویان

زمان تحویل: ۱۴۰۲/۱۰/۱۱

آذر ۱۴۰۲

فهرست

سوال ۱ - LLM	۳
سوال ۲ - Prompt engineering	۵
سوال ۳ - Speech synthesis	۶
مقدمه	۶
مدل SpeechT5	۶
تولید گفتار به زبان فارسی	۸
مجموعه داده	۸
شبکه‌های کمکی	۹
پیاده‌سازی و آموزش	۹
سیاست نمره‌دهی	۱۰
نمونه کد بارگذاری مدل ها و ابزار های لازم برای پردازش ورودی	۱۱
نکات پیاده سازی و تحویل	۱۲

در این تمرین بر روی حوزه جذاب مدل‌های زبانی بزرگ (LLMs) تمرکز می‌کنیم. وظیفه اصلی در نظر گرفته شده خلاصه‌سازی مکالمات می‌باشد بدین جهت از مجموعه داده‌ی "TweetSumm"<sup>۱</sup> استفاده می‌نماییم که یک مجموعه داده خلاصه‌سازی گفتگو، شامل حدود ۸۷۹ گفتگو با خلاصه‌های مربوطه می‌باشد. هدف این تمرین آشنا شدن شما با تجربه عملی استفاده و بکارگیری مدل‌های زبانی بزرگ (LLMs) است.

بررسی وظایف:

برای این تمرین از مجموعه داده "TweetSumm" که مجموعه‌ای از گفتگوها می‌باشد استفاده خواهیم کرد. شما باید این گفتگوها را پردازش و سپس با استفاده از رویکردهای بیان شده، خلاصه نمایید. هدف آن است که این مکالمات را به خلاصه‌هایی سازگار و آگاهانه تبدیل کنید. برای این کار، شما از مدل قدرتمند زبانی "stablalm-3b"<sup>۲</sup>، یک مدل زبانی بزرگ با ۳ میلیارد پارامتر استفاده خواهید کرد.

گام اول in-context learning (۱۰ نمره):

گام اول شما، آزمایش و ارزیابی استراتژی‌های مختلف in-context learning می‌باشد. بدین منظور لازم است روش‌های ذیل را آزمایش و ارزیابی نمایید:

• **zero-shot**: از طریق ساخت یک prompt مناسب، کارایی مدل در انجام وظیفه خلاصه‌سازی گفتگو را بصورت zero-shot ارزیابی نمایید.

• **one-shot**: در هنگام ساخت دستورالعمل و prompt از رویکرد one-shot استفاده نموده و کارایی مدل را در این رویکرد ارزیابی نمایید، آیا بهبودی ایجاد شده است؟

گام دوم (۳۰ نمره)

در این قسمت مدل را با استفاده از روش LoRA تنظیم دقیق (fine-tune) می‌نماییم. رویکرد LoRA بر این پایه استوار است که تفاوت بین وزن‌های مدل پیش‌آموزش و تنظیم دقیق شده اغلب دارای رتبه ذاتی پایین (low intrinsic rank) می‌باشند ازین‌رو لازم نیست تمام وزن‌های مدل آموزش ببینند و به عبارت دیگر، می‌توان وزن‌های مدل تنظیم دقیق شده را به طور مؤثر با یک ماتریس با رتبه پایین تقریب زد.<sup>۳</sup> از آنجایی که

<sup>1</sup> <https://huggingface.co/datasets/Salesforce/dialogstudio/viewer/TweetSumm>

<sup>2</sup> <https://huggingface.co/stabilityai/stablalm-3b-4e1t>

<sup>3</sup> Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L. & Chen, W. (2021) 'LoRA: Low-Rank Adaptation of Large Language Models', arXiv. Available at: <https://doi.org/10.48550/arXiv.2106.09685>

مدل‌های زبانی بزرگ میلیاردها پارامتر دارند و تنظیم دقیق و آموزش تمام این پارامترها نیاز به زیرساخت و هزینه‌ی بالا دارد، بدین جهت از رویکرد LoRA می‌توان در آموزش مدل‌ها بهره جست.

در این قسمت ابتدا توضیح مختصری از نحوه عملکرد LoRA داشته و سپس این رویکرد را بر روی مدل stablelm-3b پیاده‌نمایید (برای این منظور می‌توانید از کتابخانه PEFT استفاده نمایید). فرآپارامتر rank را بر روی مقادیر ۱۶ و ۳۲ گذاشته و برای هر یک ابتدا گزارش دهید چند درصد از پارامترهای مدل پس از اعمال LoRA قابل آموزش هستند و چه تعداد پارامتر به مدل‌ها افزوده شده، سپس به آموزش مدل با استفاده از رویکرد بیان شده بپردازید و نتایج را باهم مقایسه کنید. تمام فرآپارامترهای بکار گرفته شده در هنگام آموزش به‌همراه دلیل استفاده از آن‌ها را ذکر کرده و توابع خطا و نحوه آموزش مدل‌ها را بصورت مختصر توضیح دهید. (به عنوان مثال توضیح دهید LoRA را بر روی کدام یک از لایه‌های مدل اعمال کرده‌اید، همچنین توجه داشته باشید هنگام مواجه با محدودیت‌های حاصله از حافظه کارت گرافیک می‌توانید از رویکردهای quantization بهره ببرید، بدین جهت می‌توانید از کتابخانه‌هایی چون bitsandbytes استفاده کنید. در صورت بکارگیری، رویکرد مورد استفاده را به‌همراه میزان کاهش حجم مدل گزارش دهید و بطور خلاصه بیان کنید این رویکردها چه مزیت‌ها و پیامدهایی خواهند داشت).

ارزیابی و مقایسه نتایج: در نهایت، شما نتایج رویکردهای مختلف از جمله in-context learning با روش‌های zero-shot و one-shot را و مدل‌های آموزش دیده شده با تنظیم دقیق LoRA را مقایسه خواهید کرد. بدین جهت از معیار ROUGE بصورت (rouge1 , rouge2 , rougeL) برای ارزیابی کیفیت و کارایی استفاده کنید.

توجه: مراحل آماده سازی داده‌های آموزش و دستورالعمل‌ها را بصورت خلاصه ذکر نمایید. در صورت استفاده از کتابخانه‌های آماده در هریک از مراحل بصورت انتزاعی و خلاصه نحوه عملکرد آن‌ها را شرح دهید.

در این بخش قصد داریم تاثیر رویکرد instruction tuning را بر روی تسک [Sport Understanding](#) از بنچمارک BBH، با اعمال سه روش مختلف prompt بررسی کرده و نتایج را تحلیل کنیم. BBH زیر مجموعه ۲۳ تسک از big-bench است که برای ارزیابی قابلیت های استدلال چند مرحله ای مدل های زبانی بزرگ طراحی شده است. همچنین در انتها با دو مقاله درباره ی ایده های بهبود روش chain of thought(COT) آشنا می شویم.

۱. ابتدا نیاز است تا بخش های ۱، ۲، ۳، ۴ و ۷ از مقاله "[Scaling Instruction-Finetuned Language Models](#)" را به صورت خلاصه بیان کنید.(۵ نمره)

۲. از API مدل [flan-t5-large](#) استفاده کرده و با سه روش 3-shot، answer-only و COT مجموعه داده تست را به مدل بدهید. دقت (accuracy) هر روش را حساب کرده، نمودار barplot آن را نمایش داده و نتایج به دست آمده را تحلیل کنید.(برای طراحی پرامپ ورودی از مقاله قسمت قبل کمک بگیرید و همچنین برای مثال های روش few-shot از [دیتاست آموزش](#) می توانید استفاده کنید)(۱۰ نمره)

۳. یک راه برای انتخاب مثال های روش few-shot در مقاله "[Active Prompting with Chain-of-Thought for Large Language Models](#)" معرفی شده است. متد و متریک های مقاله را توضیح دهید.(۳ نمره)

۴. در مقاله "[SELF-CONSISTENCY IMPROVES CHAIN OF THOUGHT REASONING IN LANGUAGE MODELS](#)" روشی برای بهبود رویکرد COT آورده شده است، آن را توضیح دهید. چرا این روش نسبت به COT بهتر عمل می کند؟(۲ نمره)

توجه : لطفا فایل csv خروجی هر روش قسمت ۲ را نیز آپلود کنید.

### مقدمه

در این سوال قصد داریم تا از مدل از پیش‌آموزش دیده SpeechT5 استفاده کنیم و یک مینی پروژه تبدیل متن به گفتار را برای زبان فارسی پیاده‌سازی نماییم. این فرایند به ما این امکان را می‌دهد تا عملکرد مدل ذکر شده را در شناسایی متن و تولید گفتار به زبان فارسی (که با توجه به داده‌های مورد استفاده برای پیش‌آموزش، یک زبان جدید به شمار می‌آید)، مورد ارزیابی قرار دهیم. از این طریق، شما با چالش‌های آموزش مدل‌های پیشرفته گفتاری، شامل درک نحوه کارکرد مدل، تهیه و پیش‌پردازش داده‌های مورد نیاز و اعمال تغییرات لازم برای آموزش مدل روی یک زبان جدید آشنا خواهید شد.

### مدل SpeechT5

در سال ۲۰۲۱ مایکروسافت مقاله مربوط به یک سیستم پیشرفته پردازش گفتار بنام SpeechT5 را منتشر نمود. این مدل بر پایه معماری مدل T5 می‌باشد، اما با توسعه و افزودن قابلیت‌های مرتبط با پردازش گفتار روی معماری پیشین، اکنون قادر است که وظایف متنوعی از جمله تشخیص گفتار، تبدیل صدا، و تولید گفتار و... را انجام دهد. اگر بخواهیم دقیق‌تر صحبت کنیم، باید بگوییم که این مدل همزمان سه مدل گفتاری در یک معماری ثابت است:

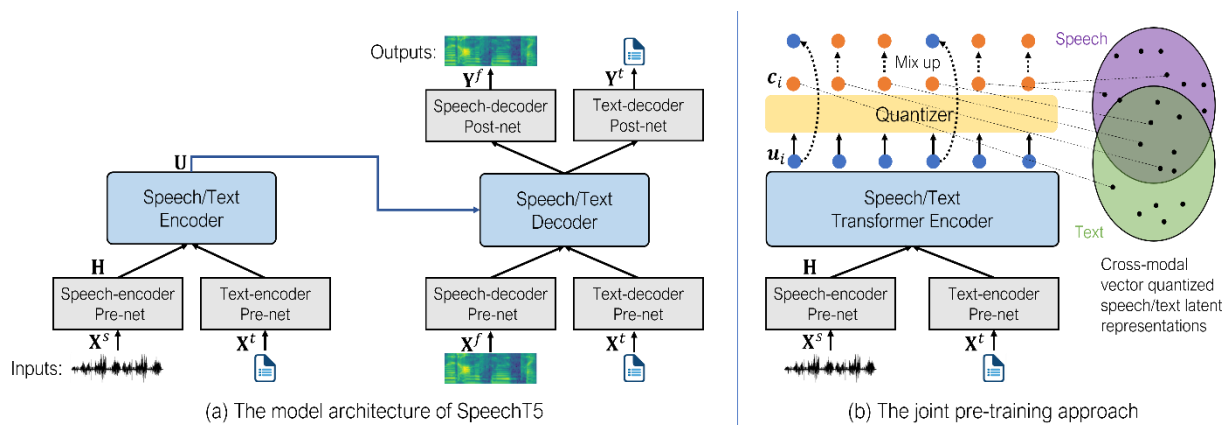
- گفتار به متن: وظایفی چون تشخیص گفتار<sup>۴</sup> و یا شناسایی گوینده<sup>۵</sup> و...
- متن به گفتار: تولید گفتار<sup>۶</sup>
- گفتار به گفتار: وظایفی چون تبدیل صدا<sup>۷</sup>، از بین بردن نویز<sup>۸</sup>، تقویت گفتار<sup>۹</sup> و...

ایده اصلی که این مدل را قادر می‌سازد تا برای همه کارکردهایی که بالا ذکر کردیم، قابل استفاده باشد، این است که یک مدل واحد را روی ترکیبی از داده‌های تبدیل متن به گفتار، گفتار به متن، متن به متن و گفتار به گفتار آموزش دهیم! به این ترتیب، مدل به طور همزمان از متن و گفتار یاد می‌گیرد. نتیجه این رویکرد

---

Speech Recognition<sup>۴</sup>  
Speaker Identification<sup>۵</sup>  
Speech synthesis<sup>۶</sup>  
Voice conversion<sup>۷</sup>  
Audio denoising<sup>۸</sup>  
speech enhancement<sup>۹</sup>

پیش‌آموزش مدلی است که دارای فضایی یکپارچه از بازنمایی‌های پنهان<sup>۱۰</sup> است که هم در متن و هم گفتار مشترک است.



شکل ۱- نمای کلی از معماری مدل SpeechT5

این مدل از یک ساختار ترنسفورمری کدگذار-رمزگشا<sup>۱۱</sup> بهره می‌برد و به عبارت بهتر یک مدل تبدیل دنباله به دنباله<sup>۱۲</sup> به شمار می‌آید که این امکان را فراهم می‌کند تا هم متن و هم گفتار را به عنوان ورودی قبول کرده و خروجی‌های مناسب را در هر دو فرمت تولید کند. برای این منظور از یک سری به اصطلاح پیش‌شبکه‌ها<sup>۱۳</sup> و پس‌شبکه‌ها<sup>۱۴</sup> استفاده می‌شود. وظیفه پیش‌شبکه‌ها تبدیل متن یا گفتار ورودی به بازنمایی‌های پنهانی است که توسط ترنسفورمر استفاده می‌شود. پس‌شبکه‌ها خروجی‌ها را از ترنسفورمر می‌گیرند و دوباره به متن یا گفتار تبدیل می‌کنند. یکی دیگر از ویژگی‌های برجسته SpeechT5، توانایی آن در حفظ هویت گویندگان است. در واقع این مدل با دریافت یک بردار تعبیه<sup>۱۵</sup> که نوعی بازنمایی از هویت و ویژگی‌های گوینده است، می‌تواند گفتاری با صدای گوینده مورد نظر تولید کند. این قابلیت به ویژه در مواردی مانند وظیفه تبدیل صوت، که می‌خواهیم گفتار ورودی با صدای یک گوینده مشخص بازتولید شود و یا در ترجمه گفتاری، که نیاز است صدای گوینده در خروجی حفظ شود، اهمیت دارد.

<sup>۱۰</sup> Hidden representation  
<sup>۱۱</sup> Transformer encoder-decoder  
<sup>۱۲</sup> Sequence to sequence  
<sup>۱۳</sup> Pre-net  
<sup>۱۴</sup> Post-net  
<sup>۱۵</sup> Embedding vector

## تولید گفتار به زبان فارسی

اگرچه همانطور که در مباحث مقدماتی این سوال مشاهده کردیم، SpeechT5 یک مدل از پیش آموزش دیده همه فن حریف است، اما یک محدودیت قابل توجه دارد، مانند مدل های گفتاری چون MMS قابلیت تولید گفتار (یا سایر وظایف) برای زبان های غیر انگلیسی را ندارد. لذا در این سوال ما قصد داریم تا این مدل را بر روی زبان فارسی فاین تیون کنیم و عملکرد مدل در تولید گفتار فارسی بیازماییم.

### مجموعه داده

اولین چالشی که برای آموزش یک مدل تولید گفتار وجود دارد، یافتن یا جمع آوری یک مجموعه داده غنی می باشد. یعنی دیتاست نه تنها باید حجم مناسبی داشته باشد بلکه باید شامل موضوعات مختلف و همچنین گوینده های متفاوتی باشد تا مدل توانایی تولید گفتار با صداها و هویت های متفاوت را نیز پیدا کند. از این رو ما برای این تمرین از یک مجموعه داده فارسی از مجموعه Common Voice که توسط موزیلا ارائه شده است استفاده خواهیم نمود که اگرچه برای یک پروژه واقعی احتمالا کافی نخواهد بود، اما برای آنچه در این تمرین از شما انتظار داریم، از حجم و البته تنوع گوینده کافی برخوردار است.

برای دانلود این دیتاست از طریق این [لینک](#) به صفحه دانلود رفته، زبان فارسی را انتخاب کرده و سپس نسخه ۱۳ دیتاست را انتخاب نمایید، در نهایت با وارد کردن ایمیل و تایید موارد خواسته شده لینک دانلود برای شما فعال خواهد شد (در صورتی که از کولب استفاده می کنید، لینک دانلود را داخل "" قرار دهید تا هنگام دانلود با خطا مواجه نشوید). در نهایت نیز لازم است از جداول train و test که با فرمت tsv در دیتاست ذخیره شده اند استفاده نمایید. این جداول شامل فهرستی از جملات به زبان فارسی و نیز آدرس فایل های صوتی مرتبط با هر جمله می باشند. توجه کنید که لزومی بر استفاده از تمام دیتاست وجود ندارد، حتی با حدود یک سوم از مجموعه داده ای که در فایل train.tsv فهرست شده نیز می توانید مدل را آموزش دهید و نتیجه بگیرید. البته در این صورت تنظیم های پارامترها به حساسیت و دقت بیشتری نیاز دارد. اما لازم است هر حجمی از دیتاست (مجموع زمان فایل های صوتی) را که برای آموزش استفاده می کنید در گزارش خود ذکر نمایید. (برای این کار لازم نیست زمان همه فایل ها محاسبه شود، می توانید به طور میانگین مدت زمان هر فایل صوتی را ۴ ثانیه در نظر بگیرید)



## شبکه‌های کمکی

مدل SpeechT5 برای آموزش سه نوع ورودی اصلی دریافت می‌کند: متن خام (سطح کاراکتر)، گفتار خام (شکل موج) و بردار تعبیه گوینده<sup>۱۶</sup> X-Vector که در بخش معرفی مدل به آن اشاره کردیم. دو مورد اول را از طریق دیتاست در اختیار دارید، اما برای تولید بردار تعبیه گوینده لازم است از یک مدل اضافی استفاده نمایید. جهت مشاهده توضیحات مربوط به این مدل، نحوه استخراج بردارهای تعبیه گوینده و سایر موارد به این [لینک](#) مراجعه نمایید. توجه به این نکته مهم است که مدل مذکور بر روی گفتار انگلیسی آموزش داده شده است، اما از آنجایی که چنین مدلی برای زبان فارسی به راحتی قابل دسترس نیست، به استفاده از همین مدل بسنده می‌کنیم. مدل دومی که برای استفاده از SpeechT5 نیاز است، یک Vocoder می‌باشد که البته ربطی به فاز آموزش مدل ندارد، اما برای تولید گفتار به آن نیاز خواهید داشت. از آنجایی که مدل به عنوان خروجی نوعی اسپکتروگرام<sup>۱۷</sup> تولید می‌کند، برای تبدیل آن به سیگنال صوتی لازم است تا از مدل HiFi-GAN استفاده نمایید.

## پیاده‌سازی و آموزش

از آنجایی که برخی از مدل‌های از پیش آموزش‌دیده‌ای که پیشتر راجع به آن‌ها صحبت کردیم، فقط در Huggingface وجود دارند، لازم است تا برای فاین تیون کردن SpeechT5 از کتابخانه transformers و ابزارها و مدل‌های ارائه‌شده توسط آن استفاده نماییم. اما توجه کنید که برای این سوال، در استفاده از این ابزارها یک سری محدودیت‌ها وجود دارد. بطور کلی شما تنها برای بارگذاری و استفاده از مدل‌های از پیش آموزش‌دیده و ابزارهای پیش‌پردازش داده‌های متنی (Tokenizer) و گفتاری (FeatureExtractor) می‌توانید از کتابخانه transformers استفاده نمایید. لذا استفاده از کلاس‌هایی مانند Processor و Trainer مجاز نمی‌باشد. همچنین توجه داشته باشید که loss در خود مدل SpeechT5 محاسبه شده و در خروجی باز می‌گردد، و لازم نیست تابع خطا را پیاده‌سازی نمایید.

---

<sup>۱۶</sup> تعبیه‌های گوینده روشی برای نمایش هویت گوینده به صورت فشرده، به عنوان یک بردار با اندازه ثابت، بدون توجه به طول گفتار است. این تعبیه‌ها اطلاعات ضروری را در مورد صدای گوینده، لهجه، لحن و دیگر ویژگی‌های منحصر به فرد که یک گوینده را از دیگری متمایز می‌کند را حفظ می‌کنند. متداول ترین تکنیک‌ها برای ایجاد بردارهای تعبیه گوینده عبارتند از:

- I-Vectors: بر اساس مدل مخلوط گاوسی (GMM) هستند و به روشی بدون نظارت (Unsupervised) تولید میشوند.
- X-Vectors: با استفاده از شبکه‌های عصبی عمیق بدست می‌آیند و عملکرد بالاتری نسبت به بردارهای I-Vectors دارند. در این روش یک شبکه عصبی برای تمایز بین گویندگان آموزش می‌بیند و گفتارهای با طول متغیر را به بردارهای تعبیه ای با ابعاد ثابت تبدیل می‌کند.

<sup>۱۷</sup> Log-mel-spectrogram

می‌دانیم که برای وظیفه تولید گفتار معیار قابل محاسبه و بی‌طرفی<sup>۱۸</sup> وجود ندارد، لذا لازم است تا ارزیابی نحوه آموزش توسط خودتان صورت گیرد. بدین ترتیب باید بعد از هر ایپاک، با استفاده از متن‌های موجود در مجموعه ارزیابی، تعدادی گفتار تولید کنید و عملکرد مدل را بررسی نمایید.

با توجه به محدودیت‌های سخت‌افزاری، ما سعی کردیم که مدل گفتاری و مجموعه داده را طوری انتخاب کنیم که در صورت پیاده‌سازی صحیح و تنظیم مناسب هایپرپارامترها، قادر باشید تا مدل را در محیط کولب آموزش دهید و قبل از به پایان رسیدن محدودیت زمانی استفاده از GPU، تولید گفتارهای قابل قبول را مشاهده نمایید (البته فقط برای برخی جملات ساده‌تر، آموزش کامل مدل به زمان بیشتر و سخت‌افزار بهتری نیاز دارد). لذا هر زمان که مدل شما قادر به تولید حتی یک گفتار صحیح شد، می‌توانید آموزش را متوقف کنید. پس از آن، گفتار تولید شده را از نظر میزان تطابق صدای گوینده، تلفظ صحیح، لحن و میزان روان بودن گفتار با نمونه اصلی مقایسه کنید و نتیجه را همراه با نمودار خطا مربوط به مجموعه آموزش و ارزیابی گزارش نمایید. همچنین در نهایت، علاوه بر مواردی که تاکنون خواسته شده، لازم است تا گزارش مختصری راجع به (۱) آموزش مدل و تغییراتی که به جهت فاین تیون کردن مدل روی زبان جدید اعمال کردید و (۲) پیش‌شبکه و پس‌شبکه‌هایی که در وظیفه تولید گفتار به کار گرفته شدند، همراه توضیح مناسب برای هر یک و (۳) تشریح رویکرد joint pre-training، علت بکار گرفتن آن در مدل SpeechT5 و نحوه عملکرد آن؛ ارائه دهید.

## سیاست نمره‌دهی

باتوجه به نوع وظیفه خواسته شده از شما در این سوال، امکان شکستن سوال به زیربخش‌های مختلف جهت توزیع نمره وجود ندارد. چراکه تمام بخش‌ها بهم وابسته بوده و نمی‌توان هیچ‌یک را جدا از دیگری ارزیابی نمود. از این جهت بارم بندی این سوال به شکل زیر خواهد بود:

- پیاده‌سازی کد - ۲۰ نمره
- آموزش صحیح مدل (کاهش معقول خطا در طی فرایند آموزش) - ۵ نمره
- ارائه گفتار تولید شده توسط مدل با کیفیت مناسب - ۵ نمره
- گزارش - ۱۰ نمره

## نمونه کد بارگذاری مدل ها و ابزار های لازم برای پردازش ورودی

```
from transformers import SpeechT5FeatureExtractor, SpeechT5Tokenizer
                        from transformers import AutoModel
from speechbrain.pretrained import EncoderClassifier
                        from transformers import SpeechT5ForTextToSpeech
                        from transformers import SpeechT5HifiGan

spk_model_name = "speechbrain/spkrec-xvect-voxceleb"
model_checkpoint= "microsoft/speecht5_tts"
vocoder_checkpoint = "microsoft/speecht5_hifigan"

feature_extractor = SpeechT5FeatureExtractor.from_pretrained(model_checkpoint)
tokenizer = SpeechT5Tokenizer.from_pretrained(model_checkpoint)
model = SpeechT5ForTextToSpeech.from_pretrained(model_checkpoint)
vocoder = SpeechT5HifiGan.from_pretrained(vocoder_checkpoint)
speaker_model = EncoderClassifier.from_hparams(source = spk_model_name,
                                              run_opts={"device": device},
                                              savedir = os.path.join("/tmp", spk_model_name))
```

## نکات پیاده سازی و تحویل

- مهلت ارسال این تمرین تا پایان روز "دوشنبه ۱۱ دی ماه" خواهد بود.
- این زمان قابل تمدید نیست و در صورت نیاز میتوانید از grace time استفاده کنید.
- پیاده سازی با زبان برنامه نویسی پایتون باید باشد و کدهای شما باید قابل اجرا بوده و به همراه گزارش آپلود شوند.
- انجام این تمرین به صورت یک نفره می باشد.
- در صورت مشاهده هر گونه تشابه در گزارش کار یا کدهای پیاده سازی، این امر به منزله تقلب برای طرفین در نظر گرفته خواهد شد.
- استفاده از کدهای آماده بدون ذکر منبع و بدون تغییر به منزله تقلب خواهد بود و نمره تمرین شما صفر در نظر گرفته می شود
- در صورت رعایت نکردن فرمت گزارش کار نمره گزارش به شما تعلق نخواهد گرفت.
- تحویل تمرین به صورت دستنویس قابل پذیرش نیست.
- تمامی تصاویر و جداول مورد استفاده در گزارش کار باید دارای توضیح (caption) و شماره باشند.
- بخش زیادی از نمره شما مربوط به گزارش کار و روند حل مسئله است.
- لطفا گزارش، فایل کدها و سایر ضمیمات مورد نیاز را با فرمت زیر در سامانه بارگذاری نمائید.

HW3\_[Lastname]\_[StudentNumber].zip

به طور مثال:

HW3\_Zilouchian\_12345678.zip

- در صورت وجود سوال و یا ابهام میتوانید از طریق رایانامه زیر با موضوع DGM\_HW3 با دستیاران آموزشی در ارتباط باشید:

○ سوال اول

محمد جواد مؤمنی : [momeni.nezhad@mail.ut.ac.ir](mailto:momeni.nezhad@mail.ut.ac.ir)

○ سوال دوم

نیکو نقویان : [nikoo.naghavian@gmail.com](mailto:nikoo.naghavian@gmail.com)

○ سوال سوم

حسین آزاد ملکی : [hosein.azadmaleki@gmail.com](mailto:hosein.azadmaleki@gmail.com)

شاد و سلامت باشید.