گزارش توسعه مدل های TTS

و STT در پروژه متاورس

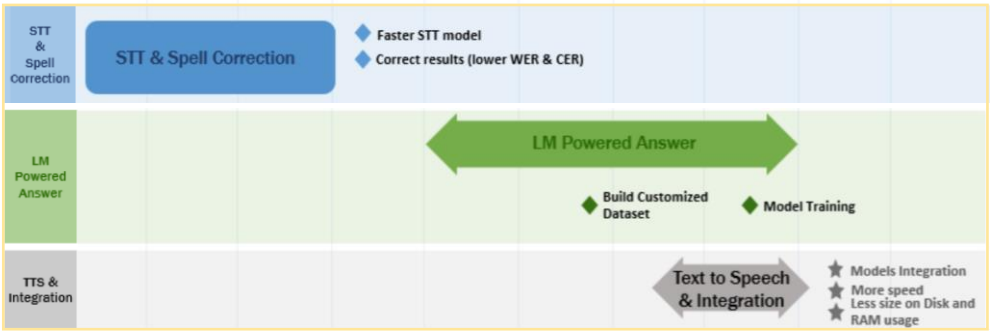
مرکز نوآوری علوم و فناوری های شناختی

دانشگاه علم و صنعت ایران

تاریخ تدوین: ۱۴۰۴/۰۱/۲۷

مقدمه

توسعه چت بات های Interactive مبتنی بر هوش مصنوعی به نیاز اساسی بسیاری از پروژه های نرم فزاری تبدیل شده است. در این پروژه به دنبال ساخت یک Interactive NPC در فضای متاورس هستیم. همانگونه که در تصویر زیر نشان داده شده است، این پروژه از ۳ بخش و ماژول اصلی تشکیل شده است. که در این گزارش جزییات مربوط به ماژول های اول و سوم مورد بحث قرار می گیرد.



۱- ماژول STT(Speech To Text) یا Automatic Speech Recognition(ASR) یا صوت به متن: صوت دریافتی از بازیکن(Player) را تجزیه تحلیل و به متن تبدیل می کند.

۲- ماژول Respond یا LM Powered: متن پردازش شده از مرحله قبل را دریافت و بر اساس تکنیک های مختلف مانند Rag و منابع دانشی مدنظر در این پروژه، به تولید Respond یا پاسخی بهینه، در سریعترین زمان ممکن به بازیکن می پردازد.

۳- ماژول TTS(Text To Speech) یا متن به صوت: پاسخ متنی تولید شده در مرحله قبل را تبدیل به صوت، با صدای شخص مدنظر در این پروژه کرده و سپس صوت را اجرا می کند.

۱- ماژول STT(Speech To Text) یا مبدل صوت به متن:

در این پروژه با توجه به ماهیت Interactive بودن NPC که می بایست به طور Real-time یا در لحظه با player ارتباط برقرار کند، در تصمیم گیری، با توجه به tradeoff موجود بین دقت بالاتر یا سرعت بیشتر، اولویت اصلی ما استفاده از مدلی است که بالاترین سرعت اجرا به همراه دقت مطلوب را ارائه دهد. بنابراین در بررسی مدل های STT ، سرعت بالاتر ملاک قرار گرفته است.

۱-۱- مدل های STT مورد بررسی در زبان عربی:

۱- Google Web Speech API(Chrip model)
۲- Speechmatics(Usra Model)
۳- Assembly Ai(Nano Model)
۴- Faster Whisper
۵- WhisperX

۶- Whisper
۷- Vosk
۸- Klaam(به عنوان ورودی حداکثر ۸ ثانیه صوت را دریافت می کند و به این خاطر مناسب این پروژه نیست)
۹- Google-Cloud-Speech: نیاز به Cloud Acount
۱۰- Pocketsphinx: عدم دسترسی به مدل عربی از طریق ارسال request در حالت رایگان
۱۱- Watson: نیاز به Subscription
۱۲- Revai: نیاز به Subscription
۱۳- Deepgram Nova: عدم پاسخگویی API در حالت رایگان
۱۴- otter.ai: عدم پاسخگویی API در حالت رایگان

در جدول زیر مقایسه میزان سرعت و دقت میان مدل های فوق برای یک صوت ۸ ثانیه ای به زبان عربی، قابل مشاهده است:

	Google Web Speech	Speechmatics	WhisperX	Faster Whisper	Vosk	Whisper	Assembly Ai
زمان/سرعت	0.91 s	2.62 s	1.8 s	1.9 s	4.5 s	3.51 s	9.4 s
WER	11%	6%	6%	6%	16%	6%	32%
دقت	89%	94%	94%	94%	84%	94%	68%
دسترسی	API	API	model	model	model	model	API
هزینه	Free	Limited, Free	Free (نیاز به اجرا بر روی سرور Local/Remote)	Free (نیاز به اجرا بر روی سرور Local/Remote)	Free (نیاز به اجرا بر روی سرور Local/Remote)	Free (نیاز به اجرا بر روی سرور Local/Remote)	Limited, Free

*WER: Word Error Rate

۱-۲- معیارهای انتخاب مدل نهایی STT در زبان عربی(عملیاتی بودن):

در این میان برخی مدل ها عملکرد مطلوبی در زبان مدنظر(زبان عربی) نداشته و برخی به علت محدودیت استفاده(نیاز به Subscription) امکان تست نداشتند. همانطور که در جدول بالا نشان داده شده است، براساس اجرای چندین تست با سمپل های مختلف عربی، مدل Google Web Speech API(Chrip model) بهترین سرعت را در صوت های کوتاه و مدل Speechmatics(Usra Model) بهترین سرعت را در صوت های بلند(در حد چند دقیقه) نشان داد. بنابراین با توجه به هدف این پروژه که "مکالمه محور" است و مکالمات معمولا در حد چند جمله و کمتر از دقیقه هستند، Google Web Speechبه عنوان مدل نهایی برای ماژول STT انتخاب شد.

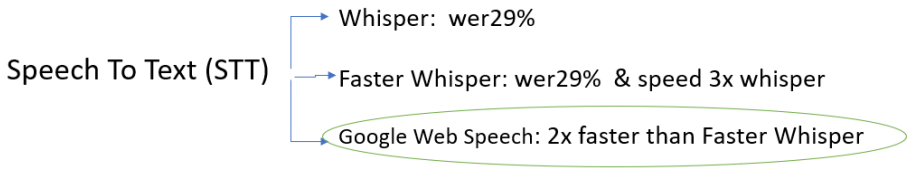
دسترسی به مدل Chrip از طریق Google Web Speech API مقدور است و این مدل opensource نمی باشد اما طبق بررسی های انجام شده در Literature، گوگل از معماری شبکه Conformer (convolution-augmented transformer) استفاده کرده است که سرعت پردازش سیگنال صوتی را بالاتر می برد.

نکته دوم در برتری این مدل رایگان بودن و در دسترس بودن به عنوان API است. مدل Speechmatics نیاز به پرداخت در تعداد بالای Request را دارد درحالیکه این مورد برای Google Web Speech API رایگان است و هزینه های اجرای پروژه را، هم از نظر نیاز به سخت افزار و هم حذف هزینه های مربوط به Subscription، کاهش می دهد.

نکته سوم اینکه علت اصلی بالاتر بودن سرعت این دو مدل به نسبت سایر مدل ها، عدم اجرای آن بر روی سخت افزار local است. درواقع مزیت سوم این مدل (پس از سرعت بالا و رایگان بودن) در این است که سخت افزار local/Remote را برای اجرا، درگیر نمی کند و به بهترین شکل توسط مبدا مورد تنظیم(Tuning) قرار گرفته و بر بستر سخت افزاری شرکت ثالث اجرا می شود. این مورد با توجه به چندمنظوره بودن و تعدد ماژول های مورد نیاز برای این پروژه(به خصوص ماژول TTS)، اهمیت بیشتری می یابد چرا که اگر تمامی ماژول ها بر بستر یک سخت افزار اجرا شود، سرعت پاسخگویی NPC پایین می آید. بنابراین در میان مدل های مورد بررسی، این مدل، عملیاتی است و سه مزیت اساسی(سرعت بالاتر، کمترین وابستگی و درگیری سخت افزار و رایگان بودن) را برای ما ایجاد می کند.

۱-۳- مدل های STT مورد بررسی در زبان فارسی:

- Whisper
- Faster Whisper
- Google Web Speech API(Chrip model)



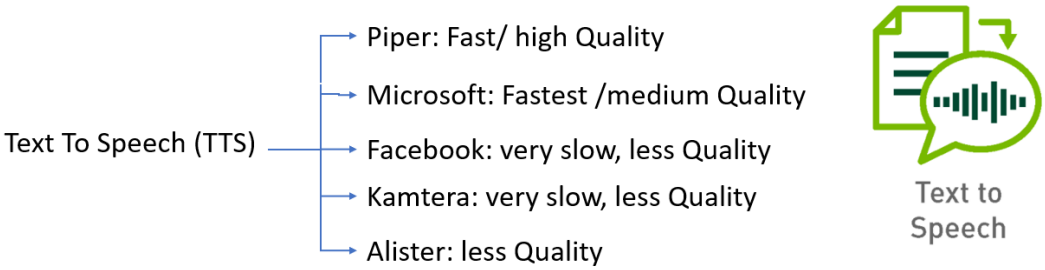
۱-۴- معیارهای انتخاب مدل نهایی STT در زبان فارسی(عملیاتی بودن):

با توجه به محدود بودن دیتاست صوتی موجود در زبان فارسی و متعاقب با آن محدود بودن تعداد مدل های پوشش دهنده این زبان، سه مدل Whisper، Faster Whisper و Google Web Speech مورد بررسی قرار گرفتند، در میان این مدل ها، مدل Chrip(Google Web Speech) در inference سرعتی دو برابر مدل Faster Whisper و مدل Faster Whisper سرعتی ۳ برابر مدل Whisper را نشان داد، در نتیجه مدل Google Web Speech برای اجرای ماژول STT فارسی انتخاب شد. از نظر دقت خروجی، هر سه مدل نتایج مشابهی را نشان دادند. نکته حایز اهمیت در دقت خروجی این مدل ها، کیفیت صدای ورودی است. به این معنا که هر چه صدا با کیفیت بالاتر ذخیره شده باشد و فاقد لحن و لهجه، به فارسی سلیس باشند، دقت خروجی این مدل ها بالاتر و هرچه با نویز و کیفیت پایین تر ذخیره شده و دارای لهجه باشند، دقت خروجی این مدل ها کمتر می شود. این امر اهمیت استفاده از میکروفون های مخصوص را برای دریافت صدای player در بازی، با کیفیتی بالاتر نشان می دهد.

۲- ماژول TTS(Text To Speech) یا مبدل متن به صوت:

ماژول دومی که در این گزارش مورد بررسی قرار می گیرد، ماژول TTS یا مبدل متن به صوت است. این ماژول، پاسخِ متنی تولید شده در مرحلهء "بازخورد مدل زبانی" را تبدیل به سیگنال های صوتی خاص(با صدای شخص مدنظر در این پروژه) کرده و سپس صوت را اجرا می کند. درمورد این ماژول نیز به مانند ماژول STT، با توجه به ماهیت Interactive بودن NPC که می بایست به طور Real-time یا در لحظه با player ارتباط برقرار کند، اولویت اصلی ما استفاده از مدلی است که بالاترین سرعت اجرا را داشته باشد اما در کنار سرعت، در این مرحله، دقت نیز اهمیتی ویژه پیدا می کند، چرا که در این ماژول به دنبال پیاده سازی "دقیق" صوت اشخاص "خاص" هستیم و از این نظر دقت به معنای نزدیک بودن صوت به لحن، گویش و صدای انتخابی، به گونه ای که با صدایی ربات گونه و ماشینی مواجه نباشیم، اهمیت پیدا می کند. بنابراین در بررسی مدل های TTS، سرعت بالا در کنار کیفیت مناسب صدا، هر دو، ملاک قرار گرفته است.

۲-۱- مدل های مورد بررسی در زبان فارسی:



۲-۲- معیارهای انتخاب مدل نهایی TTS فارسی(عملیاتی بودن):

۵ مدلِ مبدلِ متن به صوت در زبان فارسی مورد بررسی قرار گرفتند. مدل های Alister و Kamtera و Facebook به علت کیفیت پایین صوت تولید شده(صدای ربات گونه و ماشینی) مناسب کاربردهای مکالمه محور نیستند. با توجه به این که خوانش متن با صدای یک "شخصیت خاص" جزو اهداف این پروژه در "زبان فارسی" نبود، همانطور که گفته شد به علت اولویت دار بودن سرعت Respond در این پروژه، مدل Microsoft که سرعت بالاتری در زبان فارسی دارد مورد پیاده سازی قرار گرفت.

شایان ذکر است در RoadMap ابتدایی پروژه، هدف بر مدلسازی و تولید صدای یک دوبلور(آقای زینوری) بود به این شکل که پاسخ های صوتیِ کلی، از قبل ضبط شده و سپس با استفاده از مدل So-vits-svc از مدل های Voice Cloning، صدای ایشان بر روی صداهای ضبطی قرار بگیرید. بنابراین دیتاستی از صدای ایشان ساخته و صدای ایشان مورد مدلسازی دقیق قرار گرفت (آدرس فایل مربوط به این مدل در GitHub پروژه قرار داده شده است). با پیشرفت پروژه و تصمیم بر افزایش قابلیت های NPC در پاسخ به تمامی طیف های مختلف از سوالات و همچنین افزایش سرعت اجرای مدل، تصمیم بر استفاده از مدل های TTS گرفته شد. در نظر گرفته شده بود که دیتاستی از صدای یکی از گویندگان ساخته و مدلسازی روی صدای ایشان انجام گیرد که به علت اعلام اولویت دار بودنِ ساخت NPC به زبان عربی، قرار بر این شد که ابتدا ماژول های عربی کدنویسی و ساخته شوند و سپس ماژول های فارسی پیش رود. بنابراین درمورد زبان فارسی، ماژول مربوط به استفاده از مدل Microsoft

کدنویسی شده و این ماژول آماده استفاده است اما درصورتیکه نیاز به استفاده از صدای یک شخصیت خاص در این زبان داشته باشیم، باید مراحل ساخت دیتاست و مدلسازی که به شکل مبسوط در این دو سند فایل توضیح داده شده، اجرا شود.

۲-۳- مدل های مورد بررسی در زبان عربی:

- Piper
- Microsoft
- Facebook
- Klaam

۲-۴- معیارهای انتخاب مدل نهایی TTS عربی (عملیاتی بودن):

با توجه به این که در ماژول TTS عربی به دنبال مدلسازی صدای یک شخصیت خاص هستیم باید مدلی مورد انتخاب قرار می گرفت که opensource بوده تا امکان Fine-tuning و انجام مدلسازی مجدد روی دیتاست/صدای جدید وجود داشته باشد. از میان مدل های فوق مدل Microsoft به شکل Opensource در دسترس نیست و تنها امکان استفاده از چند صوت ثابت وجود دارد، بنابراین نیاز های این پروژه را تامین نمی کند.

طبق تست و بررسی های انجام شده، در میان مدل های Facebook، Klaam و Piper، مدل Piper به علت سرعت بالاتر و کیفیت طبیعی ترِ سیگنال صوتیِ تولید شده مورد انتخاب قرار گرفت. الگوریتم این مدل به طور خاص به گونه ای طراحی شده که بر روی سیستم هایی با سخت افزار محدود مانند دستگاه های Raspberry pi و حتی اسمارت فون ها نیز با سرعت مناسبی امکان پاسخگویی داشته باشد. بنابراین با توجه به بررسی های انجام شده و متناسب با کاربرد مدنظر در این پروژه که نیازمند به پاسخ Real-time هستیم، از مدل Piper برای ماژول TTS در زبان عربی استفاده می شود.

این فرآیند اکنون در حال پیشروی است و در مرحله آماده سازی دیتاست قرار دارد. در کل، اجرای مراحل زیر جهت مدلسازی TTS با مدل piper مورد نیاز است:

ساخت دیتاست شامل:

- بررسی و یافتن صوت های مناسب از صدای مدنظر با کیفیت بالا و نرخ نمونه برداری(44100HZ) و فرمت مناسب جهت مدلسازی(wav)
- انجام عملیات Denoising و Dereverberation در دو مرحله
  - با استفاده از ماژول کدنویسی پایتون و پکیج DeepFilterNet
  - در مرحله دوم حذف Reverb و نویز باقی مانده در دیتای صوتی با استفاده از نرم افزار Audacity
- حذف Background Sounds
- تک صداعه کردن صوت: حذف صدای سایر شخصیت ها از صوت، حذف صوت های اضافه مانند نفس گرفتن، سرفه، کلماتی که به طور کشیده ادا شده اند.
- تغییر Sample Rate صوت به 22050HZ
- ساخت و ایجاد متن transcript اولیه همراه با تایم استپ مربوط به هر متن، با استفاده از مدل Faster Whisper

- بازبینی متن توسط تیم عربی(نکات مربوط به اجرای این مرحله در این سند به تفصیل بیان شده است)
- ادغام داده ها ، Preprocessing و آماده سازی دیتاست جهت مدلسازی

آموزش و ارزیابی مدل شامل:

- Training: انجام مدلسازی روی دیتای صوتی ساخته شده
- Evaluation: بررسی دقت و کیفیت سیگنال های صوتی تولیدی توسط مدل Train شده
- درصورت نیاز، بازبینی دیتاست براساس خروجی تولید شده و مدلسازی مجدد تا رسیدن به کیفیت مدنظر

فرآیند فوق در مرحلهء بازبینی توسط تیم عربی قرار دارد و پس از تکمیل این بخش، فرآیند preprocessing و آموزش مدل آغاز خواهد شد.