مقاله پژوهشی

مروری بر روش‌های نوین بازشناسی گفتار

حسین هادیان
دکترای تخصصی
آزمایشگاه پردازش گفتار و زبان طبیعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
hn.hadian@gmail.com

سروش رستمی گوران
دانشجوی دکترا
آزمایشگاه پردازش گفتار و زبان طبیعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
gooran@sharif.edu

صدرا صبوری
کارشناسی
آزمایشگاه پردازش گفتار و زبان طبیعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
sadra@ee.sharif.edu

سارا صادقی
کارشناسی ارشد
آزمایشگاه پردازش گفتار و زبان طبیعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
srsadeghi0@gmail.com

یاسین امینی
کارشناسی ارشد
آزمایشگاه پردازش گفتار و زبان طبیعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
amini.yasin1995@gmail.com

حسین صامتی*
دانشیار
آزمایشگاه پردازش گفتار و زبان طبیعی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف
sameti@sharif.edu

تاریخ دریافت: ۱۴۰۱/۱۰/۲۳ تاریخ پذیرش: ۱۴۰۱/۱۲/۲۸

چکیده

این مقاله مروری‌است بر روش‌های سنتی و نیز روش‌های نوین بازشناسی گفتار. بازشناسی گفتار سابقه‌ای در حدود چندین دهه دارد و با روش‌های مبتنی بر پردازش سیگنال و پیچش زمانی پویا آغاز شده است. روش‌های آماری در دهه ۱۹۸۰ به بعد مورد توجه و استقبال قرار گرفت و روش‌های مبتنی بر مدل مخفی مارکوف به‌عنوان سرآمد این روش‌ها شناخته می‌شد. ولی از دهه ۲۰۰۰ میلادی به بعد روش‌های آماری کم‌کم جای خود را به مدل‌های مبتنی بر شبکه‌های عصبی دادند و با روی کار آمدن شبکه‌های عصبی ژرف، نتایج بهتری از این مدل‌ها نسبت به مدل مخفی مارکوف به‌دست آمد. مدل‌های مبتنی بر شبکه‌های عصبی ژرف نیز دچار تحول شدند و انواع مختلفی از آنها ابداع گردید. سپس مدل‌های مبتنی بر مبدل‌ها و مدل‌های از پیش آموزش دیده جای آنها را گرفتند و به دقت‌های بالاتری دست یافتند. در این مقاله بعد از مروری بر روش‌های مبتنی بر مدل مخفی مارکوف به روش‌های مبتنی بر شبکه‌های عصبی ژرف و ساختارهای متنوع آنها پرداخته می‌شود و در نهایت روش‌های مبتنی بر مدل‌های از پیش آموزش دیده تشریح می‌شود و آخرین روش‌های از این دست مورد بررسی قرار می‌گیرد. در انتها نیز نتایج به‌دست آمده از روش‌های تشریح شده براساس نرخ خطای کلمه ارائه می‌شود و مقایسه بین آنها صورت می‌گیرد.

واژگان کلیدی: بازشناسی گفتار، مدل مخفی مارکوف، شبکه‌های عصبی ژرف، مبدل‌ها، مدل‌های از پیش آموزش دیده

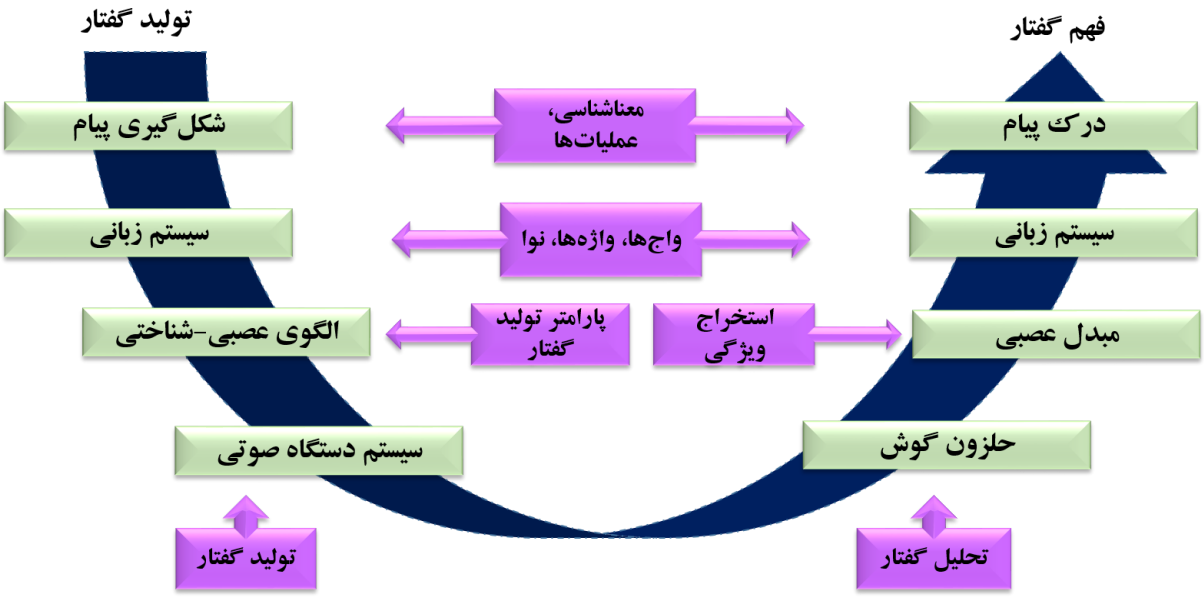
۱. مقدمه

گفتار برای انتقال اطلاعات از گوینده به شنونده به‌صورت طبیعی استفاده می‌شود. تولید و ادراک گفتار هر دو اجزای مهم زنجیره گفتاری هستند. شروع تولید گفتار در مغز انسان با تولید مفاهیم مورد نظر برای انتقال است. سپس این مفاهیم به کلمات و جملات یک زبان طبیعی تبدیل می‌شوند و پس از آن اصوات متناظر با بیان این جملات یا کلمات تعیین می‌شود. در آخرین مرحله دستور لازم به اجزای سامانه تکلمی انسان با تحریک‌های عضلانی برای تولید صداهای گفتاری ارسال می‌شود. شنونده گفتار را در سامانه شنوایی دریافت می‌کند و برای تبدیل به سیگنال‌های عصبی قابل درک برای مغز پردازش می‌کند. گوینده با دریافت گفتار خود به‌عنوان بازخورد، به طور مداوم اجزای تکلمی خود را زیر نظر دارد و کنترل می‌کند. همان‌طورکه در شکل ۱ نشان داده شده است، فرآیند تولید گفتار با یک پیام معنایی در ذهن فرد آغاز می‌شود که باید به شنونده از طریق گفتار منتقل شود. همتای رایانه‌ای فرآیند شکل‌گیری پیام، معناشناسی کاربردی است که مفهومی را ایجاد می‌کند که باید بیان شود. پس از تشکیل مفهوم پیام، مرحله بعدی تبدیل پیام به دنباله‌ای از کلمات است. هر کلمه از دنباله‌ای از واج‌ها تشکیل شده است که با تلفظ کلمات مطابقت دارد. هر جمله همچنین شامل یک الگوی عروضی است که مدت زمان هر واج، آهنگ جمله و بلندی صداها را در بر دارد. هنگامی که دنباله واجی جمله و ویژگی‌های آهنگین صداها مشخص شد، سخنگو مجموعه‌ای از سیگنال‌های عصبی–ماهیچه‌ای را تولید می‌کند. دستورات عصبی–ماهیچه‌ای نگاشت دقیقی را برای کنترل تارهای صوتی، لب‌ها، فک، زبان و نرم‌کام انجام می‌دهند و در نتیجه توالی صداها را به‌عنوان خروجی نهایی تولید می‌کنند. فرآیند درک گفتار به‌ترتیب معکوس عمل می‌کند. ابتدا سیگنال پس از پردازش اولیه در گوش بیرونی و گوش میانی، به حلزونی گوش داخلی فرستاده می‌شود که تحلیل فرکانسی را به‌صورت بانک فیلتر انجام می‌دهد. سپس سیگنال طیفی استخراج شده توسط اعصاب شنوایی به مغز فرستاده می‌شود. بقیه فرآیند بازشناسی گفتار و درک معنای زبانی در مغز صورت می‌گیرد.

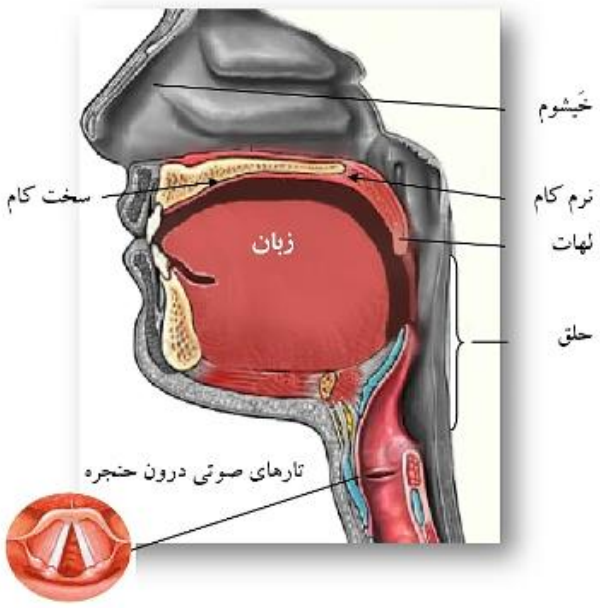
ساز و کار تولید صدا در انسان شامل سه بخش است که همگی آنها در مجرای صوتی قرار دارند و در شکل ۲ قابل مشاهده‌اند. این سه بخش عبارت‌اند از:

- ریه: که فشار هوای کافی را برای ارتعاش تارهای صوتی فراهم می‌کند.
- تارهای صوتی حنجره: که به‌عنوان یکی از منابع تولید صوت مورد استفاده قرار می‌گیرند.
- قسمت فوقانی حنجره : که شامل زبان و نرم‌کام و لب‌ها و سایر اجزای تکلمی است و مانند یک فیلتر عمل می‌کند و صدای تولید شده توسط حنجره را با یک الگوی خاص طیفی تقویت یا تضعیف می‌کند. همچنین بخش‌هایی از این قسمت خود می‌توانند نقش اصلی در شکل گیری سیگنال تحریکی که در اثر لرزش تارهای صوتی به وجود نیامده‌اند را داشته باشند.

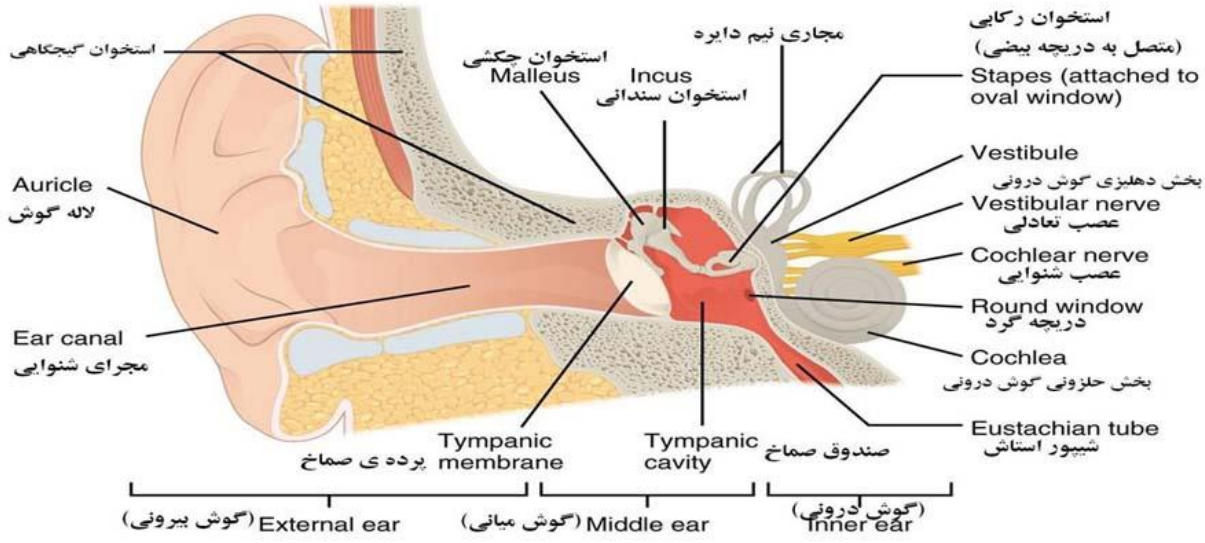
برای ایجاد ارتباط، صدای تولید شده توسط فرد اول می‌بایست توسط فرد دوم شنیده شود. اندام شنوایی در انسان همان گوش است که علاوه بر وظیفه شنوایی وظیفه حفظ تعادل انسان را نیز دارد. ساختار گوش انسان به‌عنوان یک گیرنده و فیلتر عمل می‌کند که در آن محرک های شنوایی با لرزاندن پرده گوش، انتقال از طریق گوش میانی به گوش درونی و پردازش طیفی در گوش درونی به اطلاعاتی تبدیل می‌شوند که توسط مغز کدگشایی می‌شوند.



شکل ۱. عوامل اصلی در تولید و درک گفتار [۱]



شکل ۲. نمایی از دستگاه تولید گفتار [۱]



شکل ۳. ساختار سیستم شنوایی جانبی شامل گوش خارجی، گوش میانی و گوش درونی [۲]

بنابراین، عملکرد اصلی گوش تشخیص، پردازش، تبدیل صداها به سیگنال‌های الکتریکی و ارسال آن به مغز است. ساختار کلی گوش انسان را در شکل ۳ می‌بینیم.

تلاش‌های متعددی برای شبیه‌سازی این شکل از ارتباط بین انسان‌ها توسط کامپیوتر انجام شده‌است. از مجموعه فعالیت‌های تعریف شده برای شبیه‌سازی این ساختارهای عملکردی در انسان در زیر مجموعه پردازش صوت می‌توان به بازشناسی گفتار (تبدیل کردن گفتار به نوشتار) و سنتز گفتار (تبدیل کردن نوشتار به گفتار) اشاره کرد. در این مقاله به روش‌های نوین بازشناسی گفتار می‌پردازیم. این روش‌ها مبتنی بر شبکه‌های عصبی هستند و در واقع در مقابل روش‌های سنتی (که از مدل مخلوط گاوسی استفاده می‌کنند) قرار می‌گیرند. به طور کلی تقسیم‌بندی شناخته‌شده‌ای برای این روش‌ها وجود ندارد؛ ولی ما در اینجا برای شهود بهتر، این روش‌ها را به دو دسته اصلی تقسیم میکنیم: روش‌هایی که از هم‌ترازی‌های یک مدل قبلی استفاده می‌کنند و روش‌هایی که مستقل از چنین اطلاعاتی هستند. این مساله مهم است، چراکه اولا هم‌ترازی‌های یک مدل قبلی منجر به یک پیش‌قدر[۱] (هرچند اندک) در مدل جدید می‌شوند که در اکثر موارد منجر به بهبود می‌شود ولی در شرایطی خاص ممکن است مطلوب نباشد. دوما، بدیهی است که برای استفاده از این روش‌ها نیاز به آموزش یک مدل اولیه خواهیم داشت که سرعت ساخت مدل را کاهش می‌دهد. روش‌های نوین که نیاز به هم‌ترازی اولیه دارند، عبارتند از روش ترکیبی[۲] HMM-^{3}DNN و روش مرز دانش[4] LF-MMI .

در مقابل، روش‌های نوینی که مستقل از هم‌ترازی اولیه هستند، عبارتند از روش CTC[5]، روشRNN-Transducer[6]، روشهای مبتنی بر ساز و کار توجه[۷] و نیز مبتنی بر[8]CNN. هم‌چنین تعدادی از مدل‌های جدید در انتها معرفی می‌شوند. در ادامه ابتدا کلیات بازشناسی گفتار به‌صورت خلاصه مرور می‌شود و سپس در بخش‌های بعد، روش‌های نوین در این دو دسته توضیح داده می‌شوند.

۲. ساختار کلی روش‌های بازشناسی گفتار مبتنی بر مدل مخفی مارکوف

همان‌طورکه در مقدمه اشاره شد مدل مخفی مارکوف به خاطر تئوری قوی و ساده بودن، سال‌هاست در سیستم‌های بازشناسی گفتار استفاده می‌شود. گرچه برای رسیدن به دقت‌های بالا، نیاز به در نظر گرفتن جزئیات پیچیده‌ی

فراوانی است [۳]. اگر بازشناسی گفتار را به‌صورت یک مساله‌ی آماری/احتمالی مدل کنیم، در حالت کلی هدف ما پیدا کردن W است که احتمال پسین $p(W|X)$ آن به شرط سیگنال ورودی X بیشینه شود. مدل مخفی مارکوف، یک مدل تولیدی است (که قاب‌های سیگنال گفتار را به‌عنوان مشاهدات تولید می‌کند)، بدین معنی که توزیع $P(X|W)$ را مدل می‌کند. بنابراین با اعمال قاعده بِیز، برای بازشناسی گفتار رابطه زیر را بیشینه می‌کنیم:

$$W^* = arg\ max_W P(W|X)$$
$$= arg\ max_W P(X|W)P(W).$$

مشاهده می‌شود که عبارت $P(X|W)$ در سمت راست ظاهر شده است. این عبارت که اصطلاحا مدل آکوستیکی نامیده می‌شود، معمولا توسط مدل مخفی مارکوف مدل می شود. عبارت $P(W)$، ، مدل زبانی نامیده می‌شود و توزیع حاشیه‌ای تمام جملات ممکن را مدل می‌کند.

در عمل، سیگنال ورودی X با دنباله‌ای زمانی از قاب‌ها بازنمایی می‌شود و به آن دنباله مشاهدات[۹] یا دنباله قاب[۱۰] و یا به طور دقیق‌تر دنباله بردارهای ویژگی گفته می‌شود. یک قاب، یک قطعه از سیگنال به طول معمول ۲۰ تا ۵۰ میلی‌ثانیه است که با یک بردار ویژگی بازنمایی می‌شود. معمولا قاب‌های گفتار هم پوشانی دارند. به فرایند تبدیل سیگنال گفتار به دنباله ای از قاب ها، استخراج ویژگی گفته می شود. یکی از مرسوم ترین روش ها برای استخراج ویژگی، روش MFCC[11] است. البته مرسوم است که مشتق اول و دوم زمانی قاب ها نیز به بردار ویژگی چسبانده شود و بعضا از تبدیل‌های خطی مثل LDA[12] [۴] برای کاهش بعد استفاده می‌شود. سه مفهوم مدل زبانی، مدل آکوستیکی و استخراج ویژگی در شکل ۴ که فرایند بازشناسی گفتار را نشان می‌دهد، آمده است.

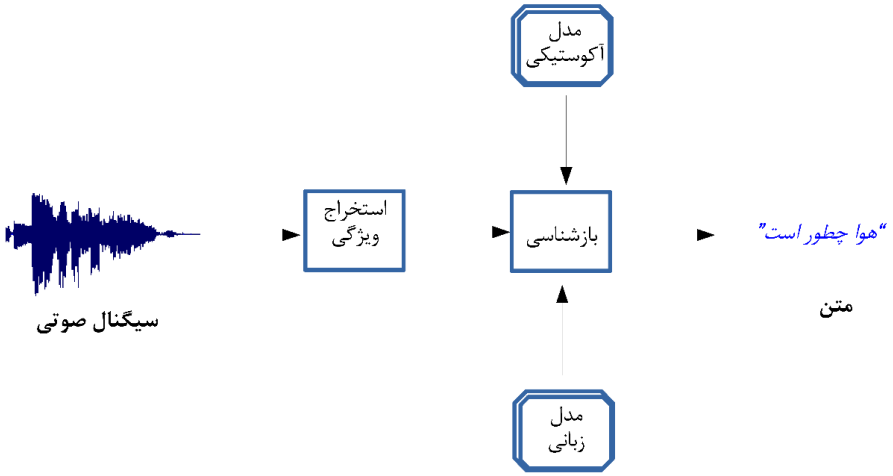
در عمل، متن W، دنباله‌ای از برچسب‌ها است. برای مثال در بازشناسی پیوسته گفتار با واژگان بزرگ LVCSR[13]، برچسب‌ها کلمات زبان هستند. یکی از ویژگی‌های مهم مدل مخفی مارکوف این است که مدل‌های زبانی n-گرام [۵] به سادگی با آن ترکیب می‌شوند و فرایند جست‌وجو و کدگشایی[۱۴] (طبق رابطه ۲) به‌صورت کارآمد توسط الگوریتم ویتربی[۱۵] انجام می‌شود [۳]. برای مثال با استفاده از ماشین‌های حالت محدود وزن‌دار[۱۶] (به اختصار WFST)، یک چارچوب کلی و کارآمد برای ترکیب قسمت‌های مختلف در سیستم بازشناسی گفتار ارائه شده است [۶], [۷]. ابزار کلدی[۱۷] که یک ابزار قوی و به‌روز برای تحقیق در زمینه بازشناسی گفتار است، نیز از این چارچوب استفاده می‌کند [۸].

همچنین ارزیابی سیستم های LVCSR تقریبا همیشه براساس نرخ خطای کلمه است که به‌صورت رابطه (۱) بین یک متن مرجع و متن بازشناسی شده تعریف می‌شود:

$$WER = \frac{S + D + I}{N} \qquad (۱)$$

که در آن، S تعداد کلماتی در متن بازشناسی است که (نسبت به متن مرجع) جایگزین شده‌اند، D تعداد کلماتی است که حذف شده‌اند، I تعداد کلماتی است که اضافه شده‌اند و N تعداد کل کلمات در متن مرجع است.

مساله مدل‌سازی آکوستیکی در واقع یک مساله دسته‌بندی زمانی است و با استفاده از مدل آکوستیکی، یک دنباله از قاب‌ها به دنباله‌ای از واحدهای آوایی (مثلاً واج یا واژه) نگاشت می‌شود. برای سادگی توضیحات، در اینجا فرض می‌کنیم واحد آوایی مدنظر ما واج است. بنابراین یک مدل آکوستیکی باید دو مدل‌سازی انجام دهد،



شکل ۴. نمای کلی از یک سامانه بازشناسی گفتار

یکی در راستای زمان (ساختار ترتیبی سیگنال) و دیگری به‌صورت محلی (خارج از بعد زمان):

۱. یک مدل آکوستیکی باید بتواند یک هم‌ترازی زمانی بین قاب‌ها و واج‌ها انجام دهد. به این معنی که دنباله بردارهای ویژگی را قطعه‌بندی[18] کند، طوری که تمام قاب‌های هر قطعه به یک واج یکسان نگاشت شوند. لزومی ندارد این کار به‌صورت صریح انجام شود، بلکه می‌تواند به‌صورت ضمنی باشد.

۲. پس از هم‌ترازی، برچسب هر قاب مشخص می‌شود. بنابراین مدل آکوستیکی باید بتواند توزیع احتمالاتی قاب‌ها را برای هر واج مدل کند. در واقع در اینجا، یک مدل‌سازی محلی (خارج از بعد زمان) انجام می‌شود و هر قاب به یک واج دسته‌بندی می‌شود.

البته بدیهی است که این دو مدل‌سازی با هم ارتباط تنگاتنگی دارند و همیشه نمی‌توان آنها را جداگانه تصور کرد. به طور خاص، در روش‌های مبتنی بر مدل مخفی مارکوف، دو توزیع احتمالی اصلی مدل می‌شوند. اولی، احتمال گذر بین حالت‌ها است که به دلیل گسسته بودن حالت‌ها، با یک جدول یا ماتریس قابل نمایش است. این توزیع به همراه ساختار مارکوف، در واقع مدل‌سازی زمانی را انجام می‌دهند که در بالا توضیح داده شد. دیگری، توزیع احتمالاتی تولید مشاهدات (همان قاب‌ها) در هر حالت است. در LVCSR، هر حالت نماینده‌ی یک زیرواج است (منظور از زیرواج، بخشی فرضی از یک واج است). در واقع هر واج توسط یک مدل مخفی مارکوف (معمولاً ۳ حالته) ، مدل می‌شود. درنتیجه، هر حالت، یک زیرواج از واج مربوطه را مدل می‌کند. توزیع احتمالاتی تولید مشاهده در واقع، مدل‌سازی محلی را انجام می‌دهد و به عبارتی توزیع احتمالاتی تمام قاب‌های زیرواج مربوطه را تخمین می‌زند. بنابراین، مجموعه پارامترهای قابل آموزش در مدل مخفی مارکوف متشکل از تمام پارامترهای این دو توزیع است.

همان‌طورکه گفته شد، در روش مرسوم برای LVCSR در قالب مدل مخفی مارکوف، برای هر واج یک مدل مخفی مارکوف (معمولاً ۳ حالته) درنظر گرفته می‌شود که به اختصار به آن *مدل واج* می‌گوییم. به این حالت (استفاده از یک مدل واج برای هر واج)، مدل‌سازی مستقل–از–بافت (CI[19]) یا تک واجی گفته می‌شود، چراکه بافت اطراف هر واج درنظر گرفته نمی‌شود. در مقابل، می‌توان مدل‌سازی وابسته به بافت (CD[20]) انجام داد که در آن هر واج به همراه بافت چپ/راست یا هردو مدل می‌شود. در این حالت، برای هر بافت (که مثلا می‌تواند متشکل از دو یا سه واج باشد) یک مدل واج درنظر گرفته می‌شود. در نتیجه، تعداد

کل حالت‌های مارکوف می‌تواند به‌راحتی به چندین هزار برسد که مدل‌سازی را سخت می‌کند. برای حل این مشکل، معمولاً از تکنیک درخت تصمیم و گره‌زنی[۲۱] استفاده می‌شود. به این صورت که یک درخت تصمیم روی بافت‌ها و حالت‌های مارکوف آموزش داده می‌شود تا مشخص کند کدام حالت‌ها به هم گره زده بشوند. یک نمونه مدل واج ۳ حالته در شکل ۵ نشان داده شده است [۳، ۹]. معمولاً این مدل‌های واج به تنهایی کاربردی ندارند بلکه به‌عنوان جزیی از یک گراف مارکوف بزرگتر استفاده می‌شوند. به طور خاص، دو نوع گراف مارکوف استفاده می‌شود: یکی برای آموزش پارامترهای سیستم (که متشکل از پارامترهای تمام مدل‌های واج است) و دیگری در مرحله آزمون، برای کدگشایی. لازم به توضیح است که حالت نهایی در شکل ۵ که به رنگ خاکستری مشخص شده است، غیرتولیدی است و به هنگام اتصال واج‌ها برای ساختن گراف‌های ترکیبی با حالت بعدی ادغام می‌شود. همان‌طورکه در مقدمه نیز اشاره شد، یک روش مرسوم سنتی برای مدل‌سازی محلی، مدل مخلوط گاوسی است. وقتی که مدل مخفی مارکوف با مدل مخلوط گاوسی استفاده شود، اصطلاحاً به آن HMM-GMM گفته می‌شود. یک روش قوی دیگر برای مدل‌سازی محلی روش SGMM[۲۲] است که در اینجا توضیح داده نمی‌شود [۱۰]. مرسوم‌ترین تابع هدف برای آموزش پارامترهای مدل HMM-GMM ، روش بیشینه–درست‌نمایی است [۱۱].

اگر فرض کنیم U *بیان‌آموزشی*[۲۳] $\{(\boldsymbol{x}^{(1)}, \boldsymbol{w}^{(1)}), \ldots, (\boldsymbol{x}^{(u)}, \boldsymbol{w}^{(u)})\}$ داریم (قلم پررنگ برای تاکید بر این است که این متغیرها یک دنباله را نشان می‌دهند)، تابع هدف بیشینه–درست‌نمایی به این صورت خواهد بود:

$$\mathcal{F}_{ML} = \sum_{u=1}^{U} \log p_\lambda(\boldsymbol{x}^{(u)}|\mathbb{M}_{\boldsymbol{w}^{(u)}}), \qquad (۲)$$

که در آن، λ مجموعه ی تمام پارامترهای سیستم را نشان می‌دهد. همچنین $\mathbb{M}_{\boldsymbol{w}}(\mathrm{u})$ گراف مارکوف متن $\boldsymbol{w}^{(u)}$ است که به آن گراف آموزشی نیز گفته می‌شود. این گراف در واقع، یک مدل مخفی مارکوف ترکیبی متشکل از اتصال مدل‌های واج متناظر با واج های دنباله کلمات $\boldsymbol{w}^{(u)}$ است. این گراف ترکیبی معمولا شامل مدل‌های سکوت اختیاری در ابتدا و انتها و بین کلمات و همچنین تمام تلفظ‌های جایگزین کلمات $\boldsymbol{w}^{(u)}$ است. در واقع این مدل ترکیبی، تمام دنباله واج‌های ممکن (به همراه سکوت) که متن متناظرشان $\boldsymbol{w}^{(u)}$ است را در یک گراف نمایش می‌دهد. ترم $p_\lambda(\boldsymbol{x}|\mathbb{M}_{\boldsymbol{w}})$، احتمال تولید دنباله مشاهده $\boldsymbol{x}$ توسط مدل ترکیبی مربوطه است و با فرض اینکه طول این دنباله – T باشد، داریم:

$$p_\lambda(\boldsymbol{x}|\mathbb{M}_{\boldsymbol{w}}) = \sum_{\boldsymbol{q}} p_\lambda(\boldsymbol{x}, \boldsymbol{q}|\mathbb{M}_{\boldsymbol{w}})$$
$$= \sum_{\boldsymbol{q}\in\mathbb{M}_{\boldsymbol{w}}} \delta_{q_0;\mathrm{start}}\ p(x_0|q_0) \qquad (۳)$$
$$* \prod_{t=1}^{T-1} p(q_t|q_{t-1}) p(x_t|q_t)\ \delta_{q_{T-1};\mathrm{final}}.$$

در این رابطه، q یک دنباله حالت در گراف را نشان می‌دهد و $\delta_{q_0;\mathrm{start}}$ فقط در صورتی که حالت اول برابر با حالت شروع در گراف $\mathbb{M}_{\boldsymbol{w}}$ باشد، یک و درغیر این صورت صفر است و همین‌طور برای $\delta_{q_{T-1};\mathrm{final}}$. احتمال گذر بین حالت‌ها با $p(q_t|q_{t-1})$ نشان داده شده است. نهایتا، $p(x_t|q_t)$ مدل‌سازی محلی را انجام می‌دهد و همان‌طورکه گفته شد درست نمایی حالت در زمان t به شرط قاب t اُم را نشان می‌دهد. برای آموزش مدل HMM-GMM، از روش Baum-Welch استفاده می‌شود که حالت خاصی از روش بیشینه‌سازی امید ریاضی[۲۴] است و به روزرسانی‌های آن نیز جهانی است (پس از پردازش روی کل بیان‌های آموزشی) [۹]. در این بخش توضیح بیشتری در این باره داده نمی‌شود. شکل ۶ ساختار کلی روش HMM-

GMM را نشان می‌دهد. در این شکل، بخش اول از یک گراف آموزشی (که دو واج اول آن b و u هستند) نشان داده شده است. همچنین نمودار طیف‌نگار[۲۵]سیگنال ورودی در پایین شکل نشان داده شده است.

۳. شبکه عصبی برای مدل‌سازی محلی: روش CE

یک روش بسیار مرسوم برای آموزش شبکه عصبی، که به روش ترکیبی[۲۶] یا بی‌نظمی متقاطع[۲۷](CE) شناخته می‌شود، این است که مستقیماً از هم‌ترازی‌های یک مدل قبلی که معمولاً HMM-GMM است، استفاده شود. یک هم‌ترازی $\boldsymbol{z}$ برای یک دنباله بردارهای ویژگی $\boldsymbol{x}$ در واقع یک دنباله حالت (هم طول با دنباله بردارهای ویژگی) است که معمولاً با الگوریتم ویتربی به‌دست می‌آید و نشان می‌دهد هر قاب توسط کدام حالت تولید شده است. به طور کلی، در مدل HMM-GMM به تعداد کل حالت‌های گره زده شده[۲۸] (که با N نشان می‌دهیم)، مدل مخلوط گاوسی داریم [۱۲]. در مقابل، در مدل HMM-DNN ، فقط یک شبکه عصبی داریم که وظیفه‌ی همه‌ی مخلوط‌ها را انجام می‌دهد. بنابراین این شبکه N نورون خروجی دارد. از آنجایی که این شبکه عملاً قاب‌ها را به زیرواج‌ها دسته‌بندی می‌کند، بهترین انتخاب برای ساختار خروجی آن بیش نرم (softmax) است [۱۳–۱۵] که در آن هر نورون خروجی $y_t^{(u)}(s)$ احتمال پسین حالت[۲۹] S به ازای قاب ورودی $x_t^{(u)}$ را نشان می‌دهد:

$$y_t^{(u)}(s) \triangleq P(s|x_t^{(u)}) = \frac{\exp(a_t^{(u)}(s))}{\sum_{s'} \exp(a_t^{(u)}(s'))}, \quad (۴)$$

که در آن $a_t^{(u)}$ مقدار فعال‌سازی[۳۰] لایه قبل از بیش‌نرم است. اگر هم‌ترازی قاب t از بیان u را با $z_t^{(u)}$ نشان دهیم، معمولاً برای آموزش این شبکه از تابع هدف بی‌نظمی–متقاطع (به اختصار CE) – که در عمل معادل با واگرایی[31]KL بین توزیع پسین تخمین زده شده توسط شبکه و توزیع مرجع (که توسط هم‌ترازی‌ها مشخص می‌شود) است – استفاده می‌شود:

$$\mathcal{F}_{CE} = \sum_{u=1}^{R} \sum_{t=1}^{T_u} \log y_t^{(u)}(z_t^{(u)}). \quad (۵)$$

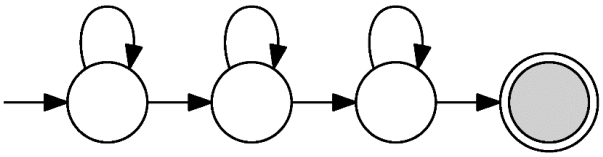
ولی همان‌طورکه گفته شد در مدل مخفی مارکوف به درست‌نمایی حالت‌ها نیاز داریم که با قاعده‌ی بیز از احتمال پسین آنها تخمین زده می‌شود:

$$\begin{aligned} &\log p(x_t^{(u)}|s) \\ &= \log y_t^{(u)}(s) + \log p(x_t^{(u)}) \\ &- \log P(s) \triangleq \log y_t^{(u)}(s) \\ &- \log P(s), \end{aligned} \quad (۶)$$

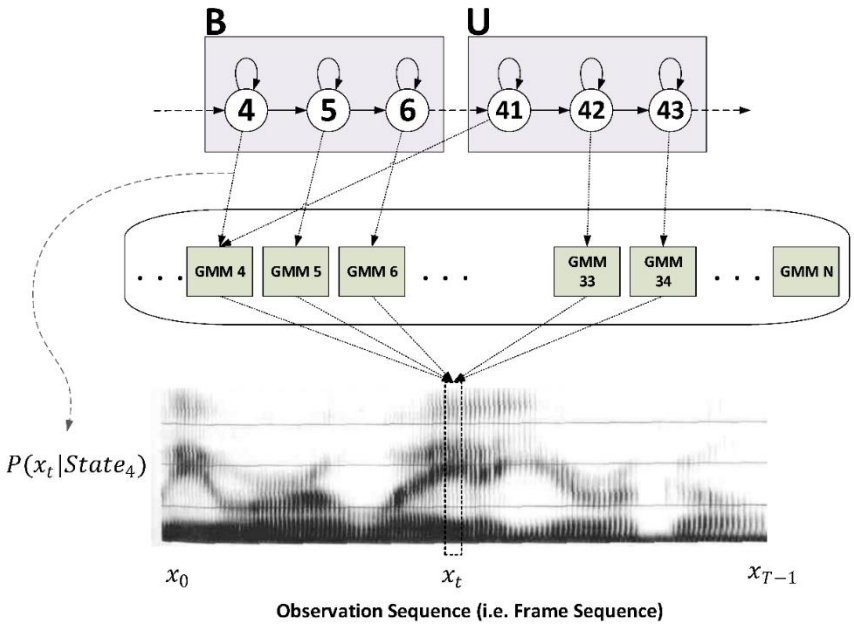
که در آن $P(s)$ احتمال پسین حالت s را نشان می‌دهد. این احتمال از میانگین‌گیری روی هم‌ترازی‌ها محاسبه می‌شود [۱۵–۱۷]. از آنجایی که احتمال پیشین قاب $p(x_t^{(u)})$ در کدگشایی طبق رابطه ۲) تأثیری ندارد، حذف می‌شود و نیازی به محاسبه آن نیست. شکل ۷ ساختار کلی این روش را نشان می‌دهد. بردار هم‌ترازی در بالای شکل دیده می‌شود.

گرادیان این تابع هدف نیز عبارت است از:

$$\frac{\partial \mathcal{F}_{CE}}{\partial a_t^{(u)}(s)} = \delta_{s;z_t^{(u)}} - y_t^{(u)}(s), \quad (۶)$$



شکل۵ یک مدل واج ۳ حالته



شکل۶. ساختار کلی مدل‌های HMM-GMM در قالب یک مثال. دقت کنید که حالت ۴ و ۴۱ به هم گره زده شده‌اند.

که وقتی صفر می‌شود که شبکه عیناً هم‌ترازی هدف را پیش‌بینی کند؛ یا به عبارتی وقتی که تنها دنباله حالت با احتمال غیر صفر در گراف آموزشی هر بیان همان دنباله حالت ویتربی باشد. البته معمولاً آموزش شبکه قبل از اینکه گرادیان صفر شود، با استفاده از اعتبارسنجی متقابل[۳۲] متوقف می‌شود.

در واقع در این روش از به کارگیری شبکه عصبی، ما صرفاً داریم همان برچسب‌هایی که با استفاده از مدل مخلوط گاوسی به‌دست آمده است را دوباره با شبکه عصبی یاد می‌گیریم. استفاده از شبکه عصبی برای مدل‌سازی محلی سه مزیت بالقوه دارد، اول اینکه ما از قدرت شبکه عصبی و توان تعمیم‌دهی آن بهره می‌بریم. دوم، با استفاده از شبکه عصبی بازگشتی می‌توانیم قاب‌های گذشته و آینده را نیز در مدل‌سازی تأثیر دهیم. و سوم آنکه، شبکه‌ی عصبی به‌صورت تمایزی آموزش داده می‌شود و بنابراین بهتر می‌تواند احتمال‌های پسین را تخمین بزند. نکته مهم آن است که در روش CE، مزیت اول و سوم به‌صورت کامل بهره‌برداری نمی‌شوند چراکه برچسب‌ها از مدل قبلی دیکته می‌شوند [۱۸]. لازم به ذکر است که معمولاً این روش به همراه پیش آموزش[۳۳] شبکه با استفاده از شبکه‌های باور عمیق استفاده می‌شود [۱۹] و کارهای زیادی روی ساختار شبکه (برای همین تابع هدف) انجام شده است [۲۲-۲۰]. مبنای این روش‌های پیش آموزش الگوریتم CD۳۴ است که آموزش یک ماشین بولتزمن محدود[۳۵]را ساده می‌کند[۱۵، ۲۳]. این

مسأله خاص بازشناسی گفتار نیست و مورد بحث ما قرار نمی‌گیرد. همچنین توابع هدف غیردنباله‌ای دیگری مثل [۲۴] نیز ارائه شده‌اند که در اینجا بررسی نمی‌شوند.

٤. روش بیشینه‌سازی اطلاعات مشترک بی‌شباک یا LF-MMI

٤-١. بیشینه‌سازی اطلاعات مشترک

قبل از پرداختن به روش LF-MMI تابع هدفی که در این روش استفاده می‌شود در اینجا شرح داده خواهد شد. این تابع هدف که تمایزی است، بیشینه اطلاعات مشترک نام دارد و انواع مختلف آن در چارچوب روش HMM-GMM استفاده شده‌اند [۲۵–۲۷] پایه‌ی توابع هدف تمایزی براساس بیشینه‌سازی اطلاعات مشترک[۳۶] (به اختصار MMI) بین دو رویداد $\mathbb{M}_{\boldsymbol{w}}$ و $\boldsymbol{x}$ (به عبارتی گراف ترکیبی و دنباله بردارهای ویژگی آن) به ازای تمام بیان‌های آموزشی است [۲۸]. اطلاعات مشترک بین این دو رویداد برابر است با:

$$\mathcal{I}(\boldsymbol{x}, \mathbb{M}_{\boldsymbol{w}}) = \log \frac{p(\boldsymbol{x}, \mathbb{M}_{\boldsymbol{w}})}{p(\boldsymbol{x})p(\mathbb{M}_{\boldsymbol{w}})} = \log \frac{p(\boldsymbol{x}|\mathbb{M}_{\boldsymbol{w}})}{p(\boldsymbol{x})}. \quad (۷)$$

با کمی دقت می‌توان دید که افزایش اطلاعات مشترک معادل با افزایش واگرایی KL بین دو توزیع $p(\boldsymbol{x})p(\mathbb{M}_{\boldsymbol{w}})$ و $p(\boldsymbol{x}, \mathbb{M}_{\boldsymbol{w}})$ را دید که این عدم استقلال در حالتی بیشینه است که این دو متغیر تابعی قطعی از یکدیگر باشند، که در آن صورت اطلاعات مشترک آنها برابر خواهد شد با بی‌نظمی مشترک[37] آنها و بی‌نظمی شرطی[۳۸] آنها برابر با صفر خواهد شد. به عبارتی دیگر، در آن صورت، متن یک بیان، یک تابع قطعی از آن خواهد بود. تابع هدف اطلاعات مشترک بیشینه یا به اختصار MMI برابر است با:

$$\mathcal{F}_{MMI} = \sum_{u=1}^{U} \mathcal{I}(\boldsymbol{x}^{(u)}, \mathbb{M}_{\boldsymbol{w}^{(u)}}) \quad (۸)$$

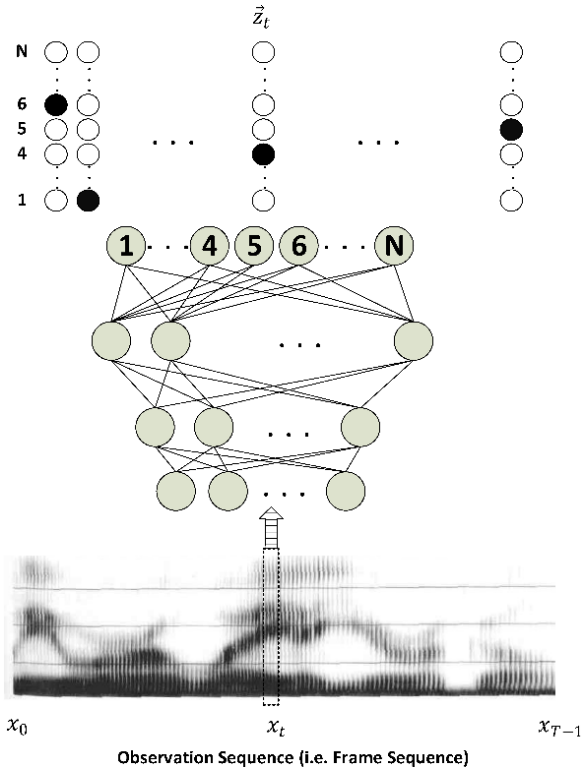
$$= \sum_{u=1}^{U} \log \frac{p_\lambda(\boldsymbol{x}^{(u)}|\mathbb{M}_{\boldsymbol{w}^{(u)}})}{p_\lambda(\boldsymbol{x}^{(u)})}.$$

ازآنجایی‌که $p(\mathbb{M}_{\boldsymbol{w}^{(u)}})$ که در واقع طبق مدل زبانی محاسبه می‌شود (یعنی فرض می‌شود که $p(\mathbb{M}_{\boldsymbol{w}^{(u)}}) \triangleq p_{LM}(\boldsymbol{w}^{(u)})$ مستقل از پارامترهای مدل مخفی مارکوف است و در بیشینه‌سازی تابع هدف تأثیری ندارد، می‌توانیم تابع هدف MMI را به‌صورت زیر بنویسیم:

$$\begin{aligned} \mathcal{F}_{MMI} &= \sum_{u=1}^{U} \log \frac{p_\lambda(\boldsymbol{x}^{(u)}|\mathbb{M}_{\boldsymbol{w}^{(u)}})p(\mathbb{M}_{\boldsymbol{w}^{(u)}})}{p_\lambda(\boldsymbol{x}^{(u)})} \\ &= \sum_{u=1}^{U} \log p(\mathbb{M}_{\boldsymbol{w}^{(u)}}|\boldsymbol{x}^{(u)}) \\ &= \sum_{u=1}^{U} \log p\left(\boldsymbol{w}^{(u)}\middle|\boldsymbol{x}^{(u)}\right). \end{aligned} \quad (۹)$$

این معادله یک دیدگاه دیگر به ما می‌دهد: تابع هدف MMI در واقع توزیع پسین معادله‌ی ۲ را مستقیماً بیشینه می‌کند درنتیجه تمایزی است.

روش مرسوم برای استفاده از این تابع در مدل HMM-GMM این است که ابتدا با روش استاندارد بیشینه–درست‌نمایی پارامترهای سیستم را آموزش می‌دهیم و سپس با روش Extended Baum-Welch تابع هدف MMI بهینه سازی می‌شود [۲۹–۳۰]. البته در اینجا بیش از این به HMM-GMM نمی‌پردازیم. توابع هدف MPE[39] و MWE[40] و sMBR[41] نیز بسیار مشابه هستند و فقط صورت کسر را با جمع وزن‌دار روی همه‌ی مدل‌ها (و نه فقط مدل مرجع) محاسبه می‌کنند. این وزن تابعی از شباهت بین مدل مرجع و مدل‌های دیگر است. گرادیان تابع MMI برابر است با [۱۴]:



شکل ۷ .ساختار کلی مدل‌های HMM-DNN بردار هم‌ترازی در بالای شکل، برای هر قاب مشخص می‌کند که کدام حالت آن را تولید کرده است. برای مثال، می‌توان دید که قاب tام با حالت ۴ هم‌تراز شده است

$$\frac{\partial \mathcal{F}_{MMI}}{\partial a_t^{(u)}(s)} = \delta_{s;z_t^{(u)}} - {}^{DEN}\gamma_t^{(u)}(s) \qquad (۱۰)$$

که در آن ${}^{DEN}\gamma_t^{(u)}(s)$ احتمال اشغال حالت s در گراف مخرج در لحظه t توسط قاب‌های بیان u است. احتمال‌های اشغال با اجرای الگوریتم پیش‌رو–پس‌رو[۴۲] بر روی یک گراف مارکوف به‌دست می‌آیند [۹]. همان‌طورکه مشاهده می‌شود این گرادیان فقط وقتی غیرصفر است که احتمال‌های اشغال در صورت و مخرج کسر باهم برابر نباشند. به عبارت دیگر، وقتی که محتمل‌ترین مسیرها در گراف صورت و مخرج مشابه نباشند. گراف صورت همان گراف آموزشی است که در بخش ۲ توضیح داده شد که برای بیشینه–درست‌نمایی استفاده می‌شود. ولی گراف مخرج مساله‌ی چالش برانگیز در MMI است. همان‌طورکه پیداست، مخرج کسر یک احتمال حاشیه‌ای است و درواقع به ازای تمام مدل‌های ممکن محاسبه می‌شود. معمولاً برای امکان‌پذیری محاسباتی از لیست n-best یا لاتیس استفاده می‌شود [۴، ۲۵، ۳۱، ۳۲]. این لیست یا لاتیس با استفاده از مدل قبلی (مثلاً CE یا HMM-GMM) محاسبه می‌شود و در واقع زیرفضایی کوچک از فضای همه مدل‌های ممکن را بازنمایی می‌کند.

نکته دیگر در مورد روش‌های مبتنی بر MMI این است که می‌توان از هم‌ترازی‌های یک مدل قبلی استفاده نکرد و به جای آن احتمال‌های اشغال را با استفاده از الگوریتم پیش‌رو–پس‌رو بر روی گراف صورت محاسبه کرد. اصطلاحاً به احتمال‌های اشغال، هم‌ترازی نرم[۴۳] نیز گفته می‌شود، چراکه طبق آنها هر دنباله حالت یک هم‌ترازی است ولی با یک احتمال مشخص که توسط احتمال‌های اشغال مشخص می‌شود. در مقابل به هم‌ترازی‌های معمولی (که در بخش ۲ تعریف شد)، بعضاً هم‌ترازی سخت[۴۴] گفته می‌شود چراکه فقط به یک دنباله حالت احتمال ۱ و به بقیه دنباله حالات،

احتمال صفر می‌دهد. در رابطه ۱۱ از هم‌ترازی‌های سخت استفاده شده است. اگر از هم‌ترازی نرم استفاده کنیم، گرادیان تابع هدف به این صورت خواهد بود:

$$\frac{\partial \mathcal{F}_{MMI}}{\partial a_t^{(u)}(s)} = {}^{NUM}\gamma_t^{(u)}(s) - {}^{DEN}\gamma_t^{(u)}(s), \quad (۱۱)$$

که در آن ${}^{NUM}\gamma_t^{(u)}(s)$ احتمال اشغال حالات بر روی گراف صورت را نشان می‌دهد و همان‌طورکه گفته شد با استفاده از الگوریتم پیش‌رو–پس‌رو روی گراف آموزشی بدست می‌آید. در عمل، بین نتایج استفاده از نرم و سخت تفاوت زیادی نیست [۱۴] (کمتر از ٪۳ بهبود نسبی در نرخ خطای کلمه) ولی باید توجه داشته باشیم که هم‌ترازی‌های سخت با استفاده از مدل CE به‌دست می‌آیند و در نتیجه با شبکه عصبی سازگارند. روش مرسوم استفاده از این تابع هدف در زمینه شبکه‌های عصبی این است که ابتدا پارامترهای سیستم با روش CE (بخش ۳) آموزش داده می‌شوند، سپس با مقداردهی اولیه شبکه عصبی از همان نقطه CE ، طی چند تکرار پارامترهای شبکه با این تابع هدف به روزرسانی می‌شوند. تعداد تکرارها با استفاده از cross-validation مشخص می‌شود.

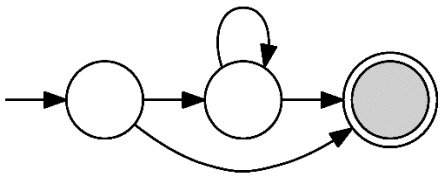
٤-۲. روش LF-MMI

روش LF-MMI در واقع گسترش نوآورانه‌ای از روش MMI با استفاده از شبکه‌های عصبی است و ساختار کلی آن مشابه روش‌های HMM-DNN است (شکل ۷). این روش بدون استفاده از مدل زبانی تقویت شده (برخلاف [۳۳]) یا ترکیب سیستم‌ها [۳۴]، در زمان ارائه دقت مرز دانش (با اختلاف قابل توجه) روی چند دادگان از جمله سوییچ برد را ارائه کرده است. همان‌طورکه در مقدمه گفته شد، این روش از همان تابع هدف MMI استفاده می‌کند. تفاوت اصلی آن با روش‌های قبلی این است که (۱) مخرج کسر را با یک گراف کدگشایی کامل محاسبه می‌کند و نه با لاتیس و (۲) از یک سیستم CE برای مقداردهی اولیه شبکه استفاده نمی‌کند بلکه آموزش شبکه از مقداردهی اولیه شروع می‌شود. استفاده از گراف کامل برای مخرج منجر به کندی شدید محاسبات می‌شود. برای حل آن از ۳ تکنیک استفاده می‌شود:

۱. مدل زبانی گراف مخرج، براساس واج است و نه کلمه. بنابراین اندازه گراف، دو تا سه مرتبه کوچک‌تر می‌شود.

۲. محاسبات گرادیان که نیاز به پیش‌رو–پس‌رو دارد، روی پردازش‌گر گرافیکی[۴۵] انجام می‌شود. لازمه‌ی این کار این است که طول هر بیان آموزشی اولاً کوتاه باشد تا در حافظه‌ی پردازش گر گرافیک بگنجد و دوما برای کل دادگان ثابت باشد تا از قابلیت‌های کشینگ[۴۶] کرنل‌های CUDA[۴۷] استفاده بهینه شود[۳۵]. بنابراین در روش LF-MMI، تمام بیان‌های آموزشی به قطعه‌های با طول ثابت شکسته می‌شوند. این کار عواقبی به همراه دارد؛ برای مثال ضرایب گرادیان در نزدیکی مرز قطعه‌ها دیگر معتبر نیست و بنابراین در آموزش شبکه تأثیر داده نمی‌شود.

۳. نرخ قاب در خروجی شبکه به یک سوم کاهش داده شده است. برای مثال به ازای ۹۰ قاب ورودی، فقط ۳۰ خروجی توسط شبکه تولید می‌شود. این کار با استفاده از شبکه‌های TDNN به‌صورت کارآمد انجام می‌شود [۳۶] .این کار نه تنها باعث می‌شود که تعداد دفعاتی که گرادیان محاسبه می‌کنیم به یک سوم کاهش پیدا کند، بلکه محاسبات داخلی شبکه عصبی را نیز کم می‌کند. مشابه این تکنیک قبلاً در [۳۷] دیده می‌شود.



شکل۸. مدل واج در روش LF-MMI

به خاطر کاهش نرخ قاب، توپولوژی مدل واج هم باید عوض شود. در واقع، باید بتوانیم در یک گذر از یک واج رد شویم، چراکه در حالت معمول که توپولوژی واج ۳ حالته است با دست‌کم ۳ گذر می‌توانیم رد شویم. بنابراین توپولوژی واج در روش LF-MMI ۲ حالته است ولی حالت اول به حالت غیرتولیدی نهایی وصل است. این توپولوژی در شکل۸ نشان داده شده است. همچنین گراف صورت برخلاف روش‌های معمول MMI ، گراف آموزشی نیست بلکه به شیوه‌ی خاصی ساخته می‌شود. برای ساخت این گراف برای هر بیان آموزشی، ابتدا با استفاده از مدل قبلی HMM-GMM یک هم‌ترازی لاتیسی تولید می‌شود. هم‌ترازی لاتیسی مشابه هم‌ترازی معمولی (یا اصطلاحاً خطی) است با این تفاوت که شامل مسیرهای ممکن دیگر (در مقابل بهترین مسیر) نیز است. سپس محدوده زمانی رخداد هر واج طبق این لاتیس مشخص می‌شود. نهایتاً این محدوده زمانی به یک میزان مشخص (که با یک پارامتر به نام tolerance تعیین می‌شود) گسترش پیدا می‌کند و طبق آن یک گراف ساخته می‌شود که هر مسیر ممکن در آن دقیقاً هم طول با دنباله بردارهای ویژگی است و هر قاب می‌تواند با واج‌هایی که طبق محدوده زمانی‌شان شاملش هستند، هم تراز شود. در واقع این کار تقریباً شبیه این است که طوقه‌ها را در گراف آموزشی (به اندازه یک محدوده زمانی مشخص) گسترش دهیم. گراف صورت در LF-MMI طوقه یا دور ندارد.

گرادیان تابع هدف در روش LF-MMI ، طبق رابطه ۱۲ محاسبه می‌شود (از هم‌ترازی‌های سخت استفاده نمی‌شود). همان‌طورکه در مقدمه اشاره شد به خاطر استفاده از گراف مخرج کامل و همچنین تمایزی بودن این روش، تمایل آن به بیش برازش نسبتاً زیاد است [۳۸][۳۹] . به همین دلیل از ۳ روش تنظیم برای افزایش تعمیم بخشی آن استفاده می‌شود:

۱. تنظیم بی‌نظمی متقاطع. در این تکنیک، یک لایه خروجی جدید (موازی با لایه خروجی اصلی) به همراه یک نسخه جدید از آخرین لایه مخفی به شبکه اضافه می‌شود. تابع هدف این لایه، تابع CE است و برچسب‌های آن هم‌ترازی نرم است که از پیش‌رو-پس‌رو بر روی گراف صورت به‌دست می‌آید.
۲. تنظیم استاندارد L2-norm که بر روی خروجی اصلی اعمال می‌شود.
۳. تکنیک Leaky-HMM . این تکنیک که روی گراف مخرج اعمال می‌شود به این صورت است که هر حالت از گراف مخرج را با یک گذر که احتمال بسیار کوچکی دارد به تمام حالات دیگر وصل می‌کنیم. این تکنیک به این خاطر استفاده می‌شود که بیان‌ها قطعه قطعه شده‌اند و درنتیجه یک قطعه ممکن است از وسط یک واج شروع شود و این تکنیک کمک می‌کند که نوعی نرم سازی اتفاق بیافتد.

این روش دقت‌های بسیار بالایی روی چندین دادگان گرفته است.

۵. روش طبقه‌بندی زمانی پیوندگرا یا CTC

روش CTC ، یک روش نوین برای ساخت مدل‌های بازشناسی گفتار است که می‌تواند به‌صورت تخت‌آغاز و بدون نیاز به هیچ مدل اولیه آموزش داده شود. ساختار شبکه عصبی در روش CTC بسیار مشابه به ساختار شبکه عصبی در روش‌های بخش قبل است با این تفاوت که در روش‌های بخش قبل، به ازای هر قاب ورودی به شبکه، احتمال‌های پسینِ حالت‌های مدل مخفی مارکوف را از خروجی شبکه می‌گرفتیم در شکل ۷ آمده است ولی در اینجا، احتمال‌های پسین برچسب (که می‌تواند واج یا حرف باشد) را می‌گیریم. شکل ۹ را ببینید. می‌توان نشان داد که روش CTC حالت خاصی از روش ترکیبیHMM-DNN با تابع هدف درست‌نمایی است [۴۰-۴۲].

برای مدل کردن سکوت، یک برچسب اضافه به نام نانوشته[۴۸] نیز مدل می‌شود. اگر مجموعه برچسب‌های زبان را با L نشان دهیم، مجموعه برچسب‌هایی که توسط شبکه مدل می‌شوند $L' = LU$ {blank} خواهد بود که {blank} نانوشته را نشان می‌دهد. بنابراین در این روش خروجی $y_t^{(u)}(l)$ احتمال پسین برچسب $l \in L'$ به گرادیان این تابع نسبت به $y_t^{(u)}(l)$ شرط قاب tاُم از بیان u (یعنی $x_t^{(u)}$) را می‌دهد:

$$y_t^{(u)}(l) \triangleq P\left(l \middle| x_t^{(u)}\right) = \frac{\exp(a_t^{(u)}(l))}{\sum_{l'} \exp(a_t^{(u)}(l'))} \quad (۱۲)$$

که در آن، $a_t^{(u)}$ مقدار فعال سازی لایه‌ی قبل از بیش نرم است. طبق واژه گذاری این روش، یک مسیر $\boldsymbol{\pi}$ عبارت است از یک دنباله برچسب در L' و احتمال آن به این صورت تعریف می‌شود:

$$p(\boldsymbol{\pi}|\boldsymbol{x}^{(u)}) \triangleq \prod_{t=0}^{T-1} y_t^{(u)}(\pi_t). \quad (۱۳)$$

در این روش یک نگاشت چند به یک $\mathcal{B}: L'^T \rightarrow L^{<=T}$ از فضای دنباله مسیرها به فضای دنباله برچسب‌ها تعریف می‌شود که تکرارهای یک برچسب و نانوشته‌ها را حذف می‌کند (T طول دنباله بردارهای ویژگی است). برای مثال $\mathcal{B}(\mathrm{a}-\mathrm{ab}-) = \mathcal{B}(-\mathrm{aa}-\mathrm{abb}) = \mathrm{ab}$ است. برای استفاده از این نگاشت، احتمال پسین یک دنباله برچسب l (که در آن برچسب‌ها از L هستند) برای دنباله بردارهای ویژگی $\boldsymbol{x}$ به‌صورت زیر تعریف می‌شود:

$$p(\boldsymbol{l}|\boldsymbol{x}) \triangleq \sum_{\boldsymbol{\pi}:\mathcal{B}(\boldsymbol{\pi})=\boldsymbol{l}} p(\boldsymbol{\pi}|\boldsymbol{x}). \quad (۱۴)$$

با این تعاریف، تابع هدفی که در این روش تعریف می‌شود، همان درست‌نمایی کل است:

$$\begin{aligned} &\mathrm{totallog-likelihood} \\ &= \sum_{u=1}^{U} \log p\left(\boldsymbol{x}^{(u)}, \mathbb{L}\left(\boldsymbol{w}^{(u)}\right)\right) \\ &= \sum_{u=1}^{U} \log p(\mathbb{L}(\boldsymbol{w}^{(u)})|\boldsymbol{x}^{(u)}) p(\boldsymbol{x}^{(u)}), \end{aligned} \quad (۱۵)$$

که در آن $\mathbb{L}$ ، یک دنباله کلمه $\boldsymbol{w}$ را به یک دنباله برچسب (که بسته به تعریف ما می‌تواند واج یا حرف باشد) نگاشت می‌کند. از آنجایی که‌ترم $p(\boldsymbol{x}^{(u)})$ در بیشینه‌سازی نقشی ندارد، تابع هدف به این صورت تعریف می‌شود:

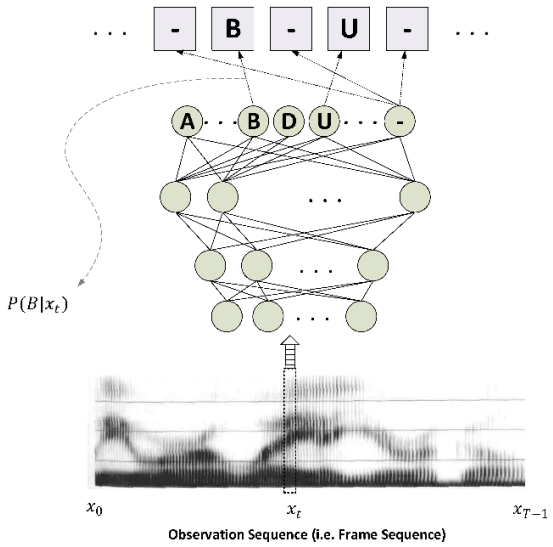
$$\mathcal{F}_{CTC} = \sum_{u=1}^{U} \log p(\mathbb{L}(\boldsymbol{w}^{(u)})|\boldsymbol{x}^{(u)}). \quad (۱۶)$$

گرادیان این تابع نسبت به $y_t^{(u)}(l)$ برابر است با:

$$\frac{۱}{p(\mathbb{L}(\boldsymbol{w}^{(u)})|\boldsymbol{x}^{(u)})[y_t^{(u)}(l)]^2} \sum_{s \in lab(\mathbb{L}(\boldsymbol{w}^{(u)}),l)} \alpha_t^{(u)}(s)\beta_t^{(u)}(s), \quad (۱۷)$$

که در آن $lab(\boldsymbol{l}, k) = s: l'_s = k$ برای یک دنباله برچسب $\boldsymbol{l}$ و یک برچسب k، مجموعه مکان‌هایی در $\boldsymbol{l'}$ را نشان می‌دهد که برچسب k دارد. دنباله برچسب گسترش یافته‌ی $\boldsymbol{l'}$، با اضافه کردن نانوشته بین هر دو برچسب در $\boldsymbol{l}$ و همچنین در ابتدا و در انتهای آن به‌دست می‌آید، بنابراین طول آن یکی بیشتر از دو برابر طول $\boldsymbol{l}$ است. همچنین متغیرهای α و β طی روندی مشابه با الگوریتم پیش‌رو-پس‌رو به‌دست می‌آیند. همان‌طورکه در فصل بعد توضیح داده خواهد شد، از آنجایی که مدل گرافیکی این روش دارای ساختار درختی است، می‌توان به‌راحتی دید که استنتاج در آن با استفاده از الگوریتم عمومی جمع حاصل‌ضرب[۴۹] یک مرحله‌ای یا دومرحله‌ای انجام می‌شود [۴۳]. بنابراین در این مورد توضیح بیشتری نمی‌دهیم.

یکی دیگر از ویژگی‌های بارز روش CTC این است که از شبکه‌های عصبی بازگشتی برای مدل‌سازی استفاده می‌کند. به طور خاص، تقریباً همیشه از لایه‌های LSTM[50] دوجهته[51] در شبکه عصبی استفاده می‌شود. ساختار



شکل ۹. ساختار کلی روش **CTC**

۵-۱. کدگشایی در روش CTC

کدگشایی در این روش عبارت است از پیدا کردن دنباله برچسب با احتمال پسین بیشینه طبق رابطه ۱۵. از آنجایی که در این رابطه یک جمع‌زنی روی تمام هم‌ترازی‌های یک برچسب انجام می‌شود، محاسبه آن برای تمام برچسب‌های ممکن،زمان‌بر است. در واقع، طبق [۴۵] هیچ روش جستجوی دقیقی برای این ارائه نشده است. بنابراین برای این کار، از یک جستجوی تخمینی استفاده می‌شود و در این تخمین، بهترین برچسب، برچسب متعلق به بهترین مسیر است. در واقع به جای اینکه برچسبی پیدا کنیم که مجموع احتمال همه‌ی مسیرهای آن بیشینه باشد.

مسیر π با احتمال بیشینه (طبق رابطه ۱۴) پیدا می‌کنیم و $\mathcal{B}(\pi)$ را برمی‌گردانیم. از آنجایی که برچسب‌ها در یک دنباله برچسب از هم مستقل هستند، برای پیدا کردن محتمل‌ترین مسیر صرفاً برای هر قاب، برچسب با بیشترین احتمال پسین شبکه را انتخاب می‌کنیم. با این روش LSTM یک ساختار شناخته شده برای شبکه‌های عصبی بازگشتی است که می‌تواند بافت گذشته (و یا آینده در حالت دوجهته) دور و نزدیک را به خوبی مدل کند [۴۴].

کدگشایی، دقت CTC برای بازشناسی گفتار پیوسته بسیار پایین است. بنابراین در عمل، از مدل زبانی جداگانه استفاده می‌شود [۴۴][۴۶][۴۷][۴۸][۴۹]. در [۴۴]، از یک الگوریتم تقریبی برای اعمال مدل زبانی استفاده می‌شود. در[۴۶]، از ساختار WFST به‌صورت کاملاً مشابه با مدل مخفی مارکوف، برای اعمال مدل زبانی و جستجوی پرتوی[52] استفاده می‌شود. در واقع به‌راحتی می‌توان دید که در مرحله کدگشایی، روند کار مشابه روش‌های HMM-DNN است. علاوه بر این، در فصل بعد نشان داده خواهد شد که در مرحله‌ی آموزش نیز، CTC درواقع تفاوت بسیار اندکی با مدل مخفی مارکوف دارد و با یک سری فرض دقیقاً معادل است. لازم به ذکر است که معمول است CTC در حالتی که برچسب‌ها به‌صورت حرف تعریف می‌شوند (و نه واج) بدون داشتن واژه نامه آموزش داده، و ارزیابی شود. بدیهی است که در این صورت برچسب‌ها باید تلفظهای جایگزین کلمات را نیز مدل کنند و این مساله مدل‌سازی را سخت‌تر می‌کند و بنابراین در شرایط برابر، دادگان بیشتری برای آموزش مدل CTC (به منظور پوشش تمام حالات کلمات جایگزین) نیاز خواهد بود. به علاوه، مدل‌سازی حرف (به جای واج) قاعدتاً در فارسی مشکل ایجاد می‌کند چراکه واکه‌ها همیشه نوشته نمی‌شوند و درنتیجه خیلی از حروف عملاً دو واج هستند و این مساله مدل‌سازی را سخت و نامتوازن می‌کند، چراکه واریانس قاب‌ها در هر برچسب بسیار بالا می‌رود.

۵-۲. تابع هدف WER در چارچوب CTC

تابع هدف دیگری که در این دسته می‌توان به آن اشاره کرد تابع هدف WER است که در زمینه‌ی روش CTC

تعریف می‌شود [۴۴]. این تابع در واقع گسترشی از تابع هدف CTC است (رابطه ۱۷). در تابع هدف WER به جای اینکه صرفاً درست–نمایی برچسب درست را (با وزن ۱) درنظر بگیریم، یک جمع زنی روی تمام برچسب‌های ممکن انجام می‌دهیم و درست‌نمایی همه‌ی آنها با وزنی متناسب با شباهت آنها با برچسب درست را درنظر می‌گیریم:

$$\mathcal{F}_{WER} = \sum_{u=1}^{U} \sum_{\boldsymbol{l}} \log p(\boldsymbol{l}|\boldsymbol{x}^{(u)}) \text{Accuracy}(\mathbb{L}) \boldsymbol{w}^{(u)}, \quad (۱۸)$$

که در آن تابع Accuracy شباهت بین دو دنباله برچسب یا به عبارتی دقت یک دنباله برچسب نسبت به دیگری را اندازه می‌گیرد. این تابع می‌تواند مثلاً نرخ خطای کلمه را اندازه بگیرد. طبق نظر نویسندگان در [۴۴]، محاسبه گرادیان این تابع به‌صورت مستقیم کارآمد نیست و بنابراین از روش‌های نمونه برداری برای تخمین قسمت‌هایی از آن استفاده می‌شود. نتایج این تابع هدف بهبود اندکی (۳٪ بهبود نسبی در نرخ خطا) نسبت به تابع هدف CTC نشان می‌دهند.

۶. روش مبدل شبکه عصبی بازگشتی یا RNN-T

روش RNN-T[۵۳] یک گسترش بر روش CTC است. این روش با اضافه کردن یک شبکه‌ی عصبی دیگر در روش CTC ، مدل‌سازی زبانی و آکوستیکی را به‌صورت مشترک انجام می‌دهد. این روش ابتدا (در سال ۲۰۱۲) برای بازشناسی واج بر روی دادگان TIMIT ارائه شد و توانست دقت مرز دانش را به‌دست بیاورد [۵۰] ولی برای LVCSR نتیجه‌ای گزارش نشد. بعد از ۵ سال در سال ۲۰۱۷ و ۲۰۱۸ دو پژوهش توانستند با استفاده از تکنیک‌های مختلف و استفاده از دادگان بسیار زیاد (حداقل ۲۰۰۰ ساعت) به دقت‌های مرز دانش دست پیدا کنند[۵۱–۵۲].

اساساً این روش یک شبکه دقیقاً مشابه شبکه‌ای که در CTC استفاده شد دارد که در اینجا به آن شبکه‌ی ترانوشت[۵۴] می‌گوییم و خروجی آن را (برای بیان ورودی u در زمان t و برای برچسب k با $a_t^{(u)}(k)$ نشان می‌دهیم (این خروجی قبل از بیش نرم است). همچنین یک شبکه‌ی دوم در این روش داریم که به آن شبکه‌ی پیش‌بینی[۵۵] اطلاق می‌شود و ورودی آن برچسب‌های دنباله برچسب مرجع و خروجی آن برچسب بعدی است. در واقع این شبکه مشابه یک مدل زبانی است که ورودی و خروجی آن از فضای برچسب‌ها است و هدف آن پیش‌بینی برچسب بعدی است. خروجی این شبکه در زمان t و برای برچسب k را با $b_t^{(u)}(k)$ نشان می‌دهیم (این خروجی قبل از بیش نرم است). در این صورت، رابطه ۱۴ در این روش به این صورت تغییر می‌کند و بقیه روابط مثل CTC است:

$$p(\boldsymbol{\pi}|\boldsymbol{x}^{(u)}) \triangleq \prod_{t=0}^{T-1} softmax(a_t^{(u)}(\pi_t) + b_t^{(u)}(\pi_t)), \quad (۱۹)$$

که در آن

$$softmax(a_t^{(u)}(\pi_t) + b_t^{(u)}(\pi_t)) = \frac{\exp(a_t^{(u)}(\pi_t) + b_t^{(u)}(\pi_t))}{\sum_k \exp(a_t^{(u)}(k) + b_t^{(u)}(k))}. \quad (۲۰)$$

همان‌طورکه مشخص است، عملاً این روش یک RNN-LM را با مدل آکوستیکی CTC ترکیب می‌کند و هر دو را باهم آموزش می‌دهد.

۷. روش توجه–پایه

روش دیگری که در این دسته جای می‌گیرد، روش نسبتاً جدید LAS[۵۶] یا اصطلاحاً توجه–پایه[۵۷] است [۵۳–۵۴]. این روش تا سال ۲۰۱۷ نتایج خیلی خوبی در زمینه LVCSR ارائه نکرده بود [۵۴–۵۵] ولی در این سال نتایج مرز دانش برای این روش گزارش شد [۵۱–۵۲]. در اصل، این روش با عنوان روش دنباله–به–دنباله[۵۸] در زمینه‌ی ترجمه ماشینی

ارائه و استفاده شده است [۵۶–۵۸]. روش LAS (و روش‌های مشابه) که در شکل ۱۰ نشان داده شده است، از دو شبکه عصبی متصل به هم استفاده می‌کنند. شبکه اول (که ورودی را می‌گیرد و معمولاً با نام کدگذار[۵۹] شناخته می‌شود) یک بازنمایی برداری $\boldsymbol{h}$ با طول U از کل دنباله بردارهای ویژگی ورودی $\boldsymbol{x}$ (که طول T دارد) به‌دست می‌آورد (معمولاً U کوچکتر از T است ولی می‌تواند هم‌طول باشد). سپس این بردار (یا تابعی از این بردار و خروجی شبکه عصبی اول) مرتباً به شبکه عصبی دوم (که بازگشتی است و معمولاً با نام کدگشا[۶۰] شناخته می‌شود) داده می‌شود و احتمال پسین برچسب بعدی به شرط دنباله ورودی و برچسب‌های گذشته (یعنی $p(y_t|y_{t-1}, \boldsymbol{x}) = \text{RNN}(\boldsymbol{h})$) را مدل می‌کند. از آنجایی که چکیده‌ی تمام قاب‌ها در h وجود دارد، این مدل‌سازی ممکن خواهد بود. به منظور تشخیص انتهای جمله یک برچسب جدید <eos> به آخر تمام دنباله برچسب‌های آموزشی اضافه می‌شود (و آموزش شبکه با آن صورت می‌گیرد). در زمان کدگشایی، اگر این شبکه این برچسب را با احتمال بالا خروجی دهد، جمله تمام می‌شود. به ابتدای جملات نیز برچسب <eos> اضافه می‌شود تا نبود بافت در ابتدای یک بیان به درستی مدل شود. یک نمونه ساختار توجه–پایه با دو شبکه گوش‌کن[۶۱] که درواقع همان کدگذار است و یک شبکه هجی‌کن[۶۲] که در واقع همان کدگشا است، در شکل ۷ نشان داده شده است. همان‌طورکه مشاهده می‌شود، شبکه عصبی کدگذار از نوع بازگشتی دوجهته است و ورودی را به یک حالت میانی h نگاشت می‌کند. شبکه عصبی دوم بردار h را به دنباله خروجی نگاشت می‌کند. ضرایب که در شکل نشان داده شده است، ضرایب توجه هستند که براساس h و اندیس زمانی محاسبه می‌شوند و در واقع مشخص می‌کنند که به هر زمان در دنباله h چه میزان باید توجه کرد.

طبق آخرین پژوهش‌های این روش تا سال ۲۰۱۷، نتایج آن ٪۳۰ تا ٪۱۰۰ نسبی (۲ تا ۵ درصد مطلق) از خطای کلمه مرز دانش ضعیف‌تر بوده است [۵۵]، ضمن اینکه در بازشناسی جملات نسبتاً بلند یا نسبتاً کوتاه (نسبت به میانگین طول جملات دادگان آموزشی)، ضعیف عمل می‌کند [۵۴].

نتایج جدیدی که در پایان سال ۲۰۱۷ و ابتدای سال ۲۰۱۸ منشر شدند، پیشرفت‌های قابل توجهی در این روش را نشان می‌دهد. تکنیک‌های جدیدی برای بهبود این روش ارائه شده است که همراه با استفاده از دادگان زیاد (بیش از ۲۰۰۰ ساعت) و با استفاده از میزان‌سازی شبکه، منجر به دقت‌های مرز دانش شده‌اند.

۸. روش‌های مبتنی بر شبکه‌های عصبی پیچشی یا CNN

این روش در سال ۲۰۱۴ و توسط محققانی از دانشگاه تورنتو و شرکت مایکروسافت ارائه شده است [۵۹]. ایده آن از اینجا مطرح شد که ترکیب شبکه عصبی عمیق و مدل مخفی مارکوف پنهان عمل‌کرد بهتری را نسبت به ترکیب مدل مخلوط گاوسی و مدل مخفی مارکوف ارائه می‌دهد. همین امر موجب شد که به جای شبکه‌های عمیق پیش‌خور مدل‌های پیچیده‌تری مانند شبکه‌های عصبی پیچشی را قرار دهند. یعنی برای محور زمان از HMM و برای محور فرکانس از CNN استفاده می‌کند. استفاده از CNN برای مدل‌سازی تغییرات محور فرکانس و بهره‌گیری از ساختارهای ویژه‌ای مانند اتصال محلی، اشتراک وزن و تجمیع، درجاتی از تغییرناپذیری نسبت به تغییرات کوچک ویژگی‌های گفتار در امتداد محور فرکانس را نشان می‌دهد، که برای مقابله با تغییرات در ویژگی‌های گوینده و محیط بسیار مهم و کارآمد است. برای پردازش داده‌های گفتار به کمک شبکه‌های عصبی پیچشی ابتدا یک نقشه ویژگی مناسب از طیف‌نگار بدونDCT، چون باعث می‌شود که محلی بودن ویژگی‌های گفتار محو شود)، مشتق مرتبه اول و مشتق مرتبه دوم آن ساخته و سپس از کنار هم قرار دادن

این سه ماتریس یک تصویر رنگی سه کاناله ایجاد کرده و آن را به‌عنوان ورودی به شبکه می‌دهد. و برای هر فریم می‌توان از یکی از دو ترکیب زیر استفاده کرد یا به تعداد فریم‌های بافت از هر یک از سه نقشه ویژگی جدا کرده و با الحاق آنها به هم یک نقشه ویژگی دو بعدی ساخته شود یا به‌صورت ترکیبی بردار هر یک از نقشه ویژگی‌ها را به‌صورت یک در میان در کنار هم قرار داد. در این کار از روش دوم استفاده می‌کند که کانولوشن یک بعدی بر محور فرکانس اعمال می‌گردد و در نهایت بعد از عبور از چند لایه پیچش و تمام متصل احتمال پسین حالت HMM مورد نظر محاسبه می‌گردد.

یکی دیگر از نوآوری‌های این کار استفاده از اشتراک وزن محدود است که توانایی بهتری را برای مدل کردن سیگنال گفتار نشان می‌دهد. اشتراک وزن‌ها معمولاً در پردازش تصویر استفاده می‌شود چون امکان دارد که یک الگوی یکسان در جاهای مختلفی از تصویر مشاهده شود اما ویژگی‌های سیگنال گفتار در باندهای فرکانسی مختلف متفاوت است پس باید اشتراک‌گذاری وزن‌ها متناسب با باندهای فرکانسی باشد لذا استفاده از مجموعه‌های جداگانه وزن برای باندهای فرکانسی مختلف ممکن است مناسب‌تر باشد زیرا امکان تشخیص الگوهای مشخص در باندهای مختلف در امتداد محور فرکانس را فراهم می‌کند. در نهایت این ترکیب CNN-HMM سبب می‌شود که میزان خطای تشخیص واج ۶ تا ۱۰ درصد بروی مجموعه داده‌های TIMIT و(Voice Search) VS نسبت به مدل ANN-HMM کاهش پیدا کند. نمونه‌ای از این شبکه برای صوت را می‌توانید در شکل ۱۱ ببینید.

۹. مدل‌های از پیش آموزش دیده

در این بخش تعدادی از مدل‌های جدید مبتنی بر شبکه عصبی مورد بررسی قرار گرفته‌اند. مدل‌های بررسی شده در این قسمت همگی ایده گرفته شده از مدل‌های زبانی از پیش آموزش دیده طراحی شده‌اند که اولین بار در حوزه پردازش زبان طبیعی معرفی شدند. این مدل‌ها معمولاً به کمک یادگیری بی‌نظارت ویژگی‌های متنی صوت را در یک شبکه بر پایه مبدل‌ها [۶۳] آموزش می‌بینند و سپس برای استفاده روی وظایف مختلف (از قبیل بازشناسی گفتار) آموزش می‌بینند.

۹-۱. مدل کانفورمر[۶۴]

مدل‌های اخیر نشان داده‌اند که شبکه‌های عصبی مبتنی بر مبدل و پیچشی عملکرد بهتری از شبکه‌های عصبی بازگشتی در بازشناسی گفتار دارند. اما هر کدام از این شبکه‌ها با محدودیت‌هایی روبرو هستند. مثلاً مبدل‌ها در درک ارتباطات سراسری بین کلمات عملکرد خوبی دارند اما در استخراج الگوهای محلی با چالش روبرو هستند. از سوی دیگر، شبکه‌های عصبی پیچشی در استخراج ویژگی‌های محلی عملکرد بسیار خوبی دارند اما برای ویژگی‌های سراسری به لایه‌ها و پارامترهای بیشتری نیاز دارند و این کار آموزش آنها را با چالش روبرو می‌کند.

کانفرمر [۶۰] مدلی است که در سال ۲۰۲۰ و توسط شرکت گوگل ارائه شده است. ایده اصلی این مدل ترکیب اطلاعات سراسری و محلی است. این ترکیب سبب می‌شود که مدل کانفرمر به‌صورت همزمان از مزایای هر دو مدل مبدل-پایه (توانایی درک ارتباطات سراسری بین کلمات) و شبکه پیچشی (درک ویژگی‌های محلی کلمات) استفاده کند.

معماری این مدل شامل کدگذار و کدگشا است که شبکه کدگذار از ماژول‌های خودتوجهی چندسری، شبکه پیچشی و دو ماژول پیشخور نصف تشکیل است و کدگشا شامل یک لایه LSTM است. معماری کدگذار این مدل در شکل ۱۲ مشاهده می‌شود.

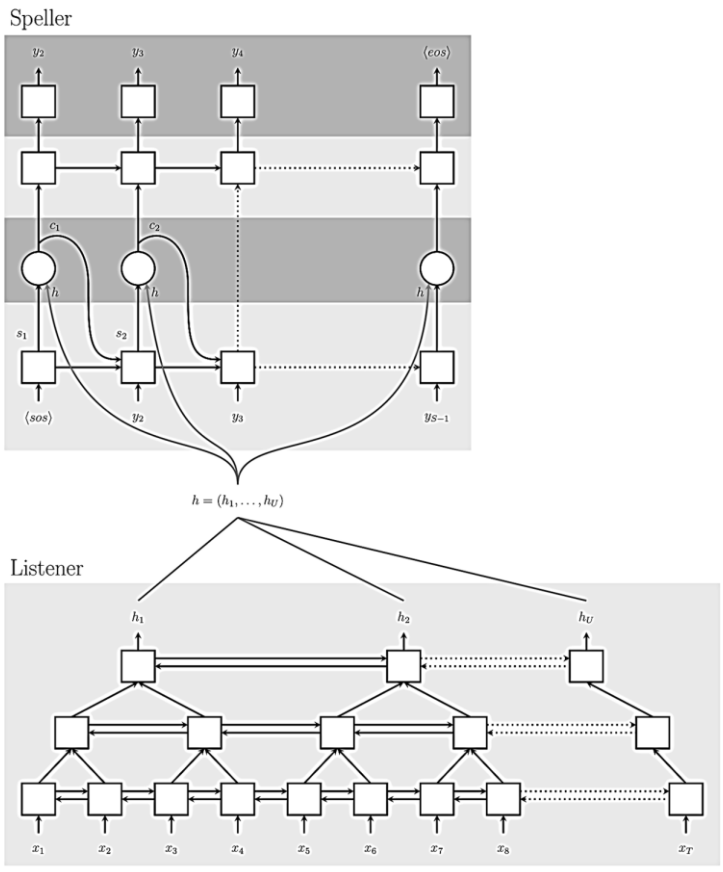
برای آموزش این مدل از مجموعه داده LibriSpeech استفاده شده و مدل توانسته است که به نرخ خطای کلمه ۲.۱ ٪ تا ۴.۳ ٪ بدون مدل زبانی و ۹.۱ ٪ تا ۹.۳ ٪ با مدل

زبانی برسد. از ویژگی‌های اصلی این مدل می‌توان به کوچک بودن اندازه مدل اشاره کرد که در حدود ۱۰ میلیون پارامتر دارد.

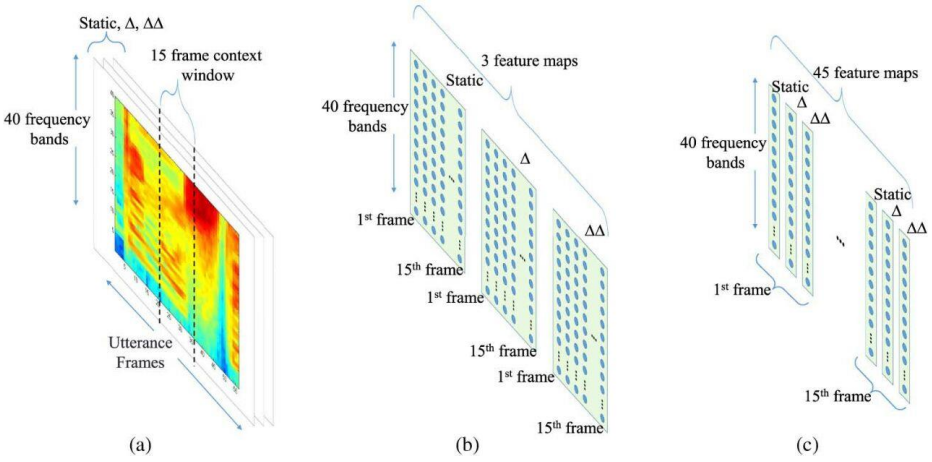
۹-۲. مدل ویوتووک[۶۵]

مدل ویوتووک [۶۱] یک مدل کدگذار-پایه است که در اواخر سال ۲۰۱۹ توسط شرکت فیس‌بوک ارائه شد. این مدل توانست به سرعت جای خود را بین بقیه مدل‌ها در حوزه پردازش گفتار باز کند و این موضوع به مدد نتایج عالی حاصله از این مدل در وظایفی نظیر بازشناسی گفتار و تحلیل احساسات بود. در نسخه دوم این مدل - که به Wav2Vec 2.0 [۶۲] مشهور است - از معماری مبدل‌ها [۶۳] هم استفاده شده است و هم‌چنین در یکی از نسخه‌های آن روی ۱۲ زبان مختلف به‌عنوان یک مدل چندزبانه آموزش دیده است. ساختار کلی مدل Wav2Vec2.0 را می‌توانید در شکل ۱۳ ببینید.

یکی از ادعاهای سازندگان این مدل سرعت یادگیری زبان جدید توسط این مدل است. سازندگان این مدل بر این باور هستند که به کمک انتقال دانش یاد گرفته شده توسط این مدل - ناشی از پیش‌اموزش روی حجم عظیمی از داده بدون برچسب - می‌توان با حدود ۱۰ دقیقه صوت برچسب دار از هر زبانی آن زبان را به این مدل آموزش داد و به نرخ خطای کلمه‌ای کمتر از ۱۰ درصد رسید.



شکل ۱۰ .ساختار شبکه‌های عصبی در روش توجه-پایه



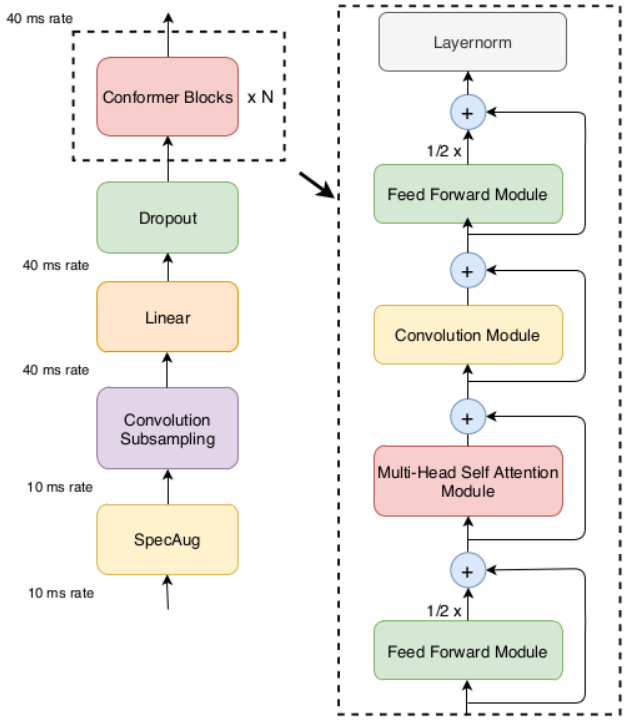
شکل ۱۱. ساختار شبکه‌های عصبی پیچشی

از دیگر ویژگی‌های این مدل استفاده مستقیم آن از داده صوت است. این در حالی است که بسیاری از مدل‌های دیگر ابتدا به کمک فیلترهای MFCC ویژگی‌هایی از صوت استخراج می‌کنند و سپس بر روی این ویژگی‌ها فرآیند آموزش مدل و پردازش صوت را انجام می‌دهند.

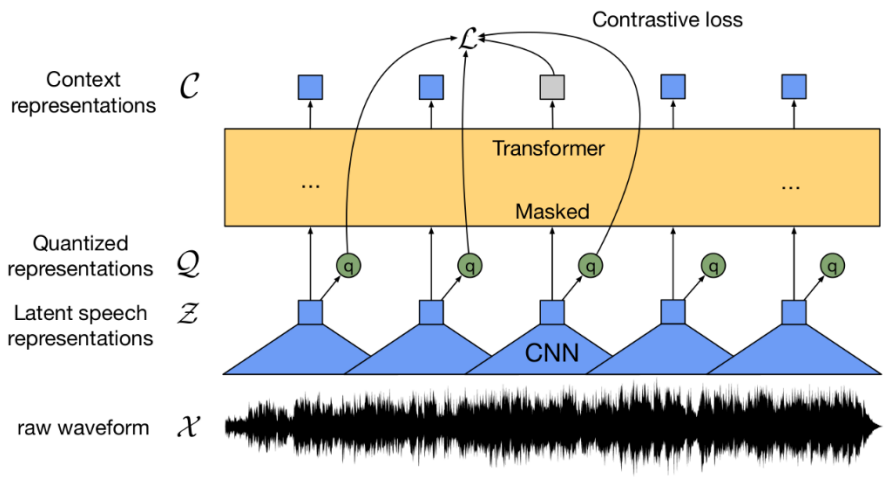
۹-۳. مدل هوبرت[۶۶]

امروزه یکی از رویکردهای غالب برای پردازش گفتار استفاده از مدل‌های خودنظارتی است. هوبرت یک مدل خودنظارتی که برای استفاده از توانایی‌های BERT در پردازش گفتار ارائه شده است. اعمال BERT بروی داده‌های صوتی با سه چالش کلیدی روبرو است که عبارت‌اند از: ۱. واحدهای صوتی متعدد در هر گفتار ورودی وجود دارد، ۲. واژه‌نامه‌ای از واحدهای صوتی ورودی وجود ندارد که در طول مرحله پیش‌آموزش به‌عنوان برچسب برای تابع زیان استفاده شود و ۳. واحدهای صوتی دارای طول‌های متغیر و بدون تقسیم‌بندی صریح هستند. ایده کلیدی این مدل انجام عملیات گسسته‌سازی از طریق خوشه‌بندی k-means است که سبب می‌شود بتوان از BERT برای پردازش صوت استفاده نمود. معماری این مدل در شکل۱۴ قابل مشاهده است.معماری این مدل از دو بخش اصلی تشکیل شده است. بخش اول تولید واحدهای مخفی با استفاده از الگوریتم خوشه‌بندی است که مشابه مدل شبه برچسب[۶۷] است. در نخستین گام سیگنال صوتی قاب‌بندی شده و MFCC متناظر با هر یک از قاب‌ها استخراج شده و سپس هر یک از این ویژگی‌ها به الگوریتم k-means داده می‌شود و به یکی از خوشه‌ها اختصاص می‌یابد. سپس همه فریم‌های صوتی بر حسب اینکه به کدام خوشه تعلق دارند، برچسب‌گذاری می‌شوند و واحدهای مخفی را تشکیل می‌دهند. سپس این واحدهای مخفی به بردارهای تعبیه شده برای استفاده در مرحله دوم آموزش تبدیل می‌شوند. در مرحله دوم مشابه با ویوتووک عمل می‌کند یعنی یک کدگذار CNN مسئول تولید ویژگی‌هایی از صوت خام است و سپس این ویژگی‌ها به طور تصادفی پوشانده شده و به کدگذار BERT وارد می‌شوند. کدگذار BERT ورودی پوشانده شده را دریافت کرده و یک توالی از ویژگی را خروجی می‌دهد و توکن‌های پوشانده شده را پیش‌بینی می‌کند. سپس این خروجی برای مطابقت با برچسب‌ها در بعد پایین‌تری پیش‌بینی می‌شود و شباهت کسینوسی بین این خروجی‌ها و هر تعبیه واحد پنهان تولید شده در مرحله اول محاسبه می‌شود. برای محاسبه خطا و آموزش مدل از تابع زیان آنتروپی متقابل استفاده می‌شود. این مدل نتایج بهتری را نسبت به مدل‌های پیشرو در حوزه بازشناسی گفتار ارائه می‌دهد بازنمایی به‌دست آمده توسط

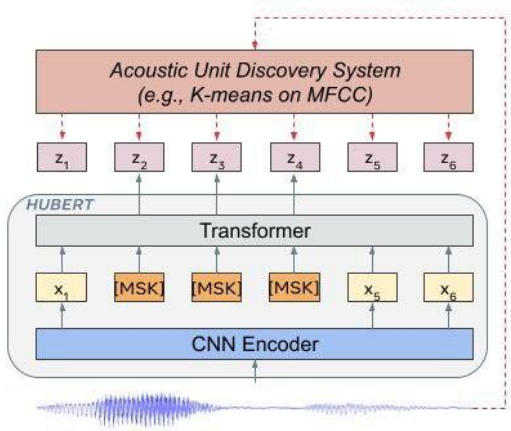
این مدل می‌تواند در وظایف دیگری مانند تولید گفتار نیز مفید واقع شود. از دیگر مزایای این مدل می‌توان به یادگیری توأمان مدل زبانی و مدل آکوستیکی و همچنین استفاده از سیگنال صوتی خام به‌عنوان ورودی اشاره کرد. برای پیش‌آموزش مدل از LibriSpeech و Libri-Light استفاده شده و در فاز تنظیم دقیق با مجموعه‌های برچسب‌گذاری شده ۱۰ دقیقه، ۱ ساعت، ۱۰ ساعت، ۱۰۰ ساعت و ۹۶۰ ساعت می‌تواند به WER ۴/۸ تا ۱/۹ در مدل‌های با پیکربندی‌های مختلف روی مجموعه ارزیابی برسد.



شکل ۱۲. ساختار کدگذار مدل کانفورمر [۶۰]



شکل ۱۳. ساختار مدل Wav2vec2.0 [۶۲]



شکل۱۴. ساختار مدل هوبرت [۶۴]

۹-٤. مدل ویوال‌ام[۶۸]

در سال‌های اخیر روش‌های خودنظارتی به دقت‌های بالایی در حوزه پردازش زبان طبیعی و بینایی ماشین و پردازش گفتار دست پیدا کرده‌اند. این روش‌ها مقادیر زیادی از داده‌های متنی و تصویری را برای ایجاد یک بازنمایی عمومی به کار گرفته و این بازنمایی‌ها می‌توانند برای انواع وظایف در این حوزه‌ها مورد استفاده قرار گیرند. اما اکثر روش‌های خودنظارتی در گفتار بر طبقه‌بندی واج و بازشناسی گفتار تمرکز دارند و بازنمایی‌های ایجاد شده کاربرد عمومی ندارند. سیگنال گفتار حاوی اطلاعات چند وجهی مانند هویت گوینده، اطلاعات فرازبانی، محتوای گفتاری و معنایی و ... است و ایجاد یک بازنمایی همه‌جانبه و عمومی که بتواند روی مسائل مختلف این حوزه به دقت خوبی برسد بسیار چالش برانگیز است چون وظایف مختلف بروی جنبه‌های مختلفی از گفتار تمرکز دارند. برای حل این چالش مدل ویوال‌ام [۶۵] ارائه شده است که گسترش‌یافته مدل هوبرت برای ایجاد یک بازنمایی همه جانبه است.

برای این کار ویوال‌ام در مرحله پیش‌آموزش از داده‌های نویزی (نویز یا دارای همپوشانی گفتاری) و پوشانده شده به‌عنوان ورودی استفاده می‌کند و هدف آن پیش‌بینی بخش‌های پوشانده شده است. آموزش به این روش سبب می‌شود که مدل هم برای وظایف وابسته به محتوای گفتار (بازشناسی گفتار) و هم برای وظایف وابسته به گوینده (جداسازی و تصدیق گوینده) کاربرد داشته باشد و بتواند یک بازنمایی عمومی و سراسری برای پردازش گفتار ارائه دهد. این مدل هم مبتنی بر مبدل است و از کدگذار پیچشی برای کاهش بعد سیگنال گفتار و از کدگذار مبدل-پایه برای ایجاد یک بازنمایی مخفی استفاده می‌کند. علاوه براین از GRPB[۶۹] یا سوگیری در بخش خودتوجه مبدل‌ها استفاده می‌کند و با این کار می‌تواند مدل‌سازی بهتری از ترتیب دنباله گفتار ارائه دهد و متناسب با محتوا یک سوگیری نسبی ایجاد می‌کند. معماری این مدل در شکل ۱۵ مشاهده می‌شود.

دادگان آموزشی شامل ۶۰ هزار ساعت Libri-Light ، ده هزار ساعت GigaSpeech و ۲۴ هزار ساعت VoxPopuli است. این مجموعه داده جدید شامل موارد آموزشی از سناریوها و حوزه‌های مختلف مانند پادکست‌ها، یوتیوب و ضبط رویدادها و مانند آن است. با این مجموعه داده به مدل این امکان داده می‌شود که محدود به داده‌های یک حوزه خاص مثلاً کتاب صوتی نباشد و قابلیت تعمیم مدل برای وظایف دیگر بالا می‌رود. مدل ارائه شده توانسته است که روی ۱۵ وظیفه مختلف در حوزه پردازش گفتار به نتایج پیش‌رو دست پیدا کند.

۹-۵. مدل ویسپر[۷۰]

یکی از مدل‌های موفق حوزه پردازش گفتار مدل که شرکت OpenAI معرفی کرده است، مدل ویسپر [۶۶] است. این مدل که بر پایه روش‌های نظارتی ضعیف[۷۱] ساخته شده است با آموزش بر روی حدود ۶۸۰ هزار ساعت گفتار به نتایجی دست یافته است که قابل رقابت با مدل‌های آموزش دیده شده به‌صورت کاملاً با ناظر است.

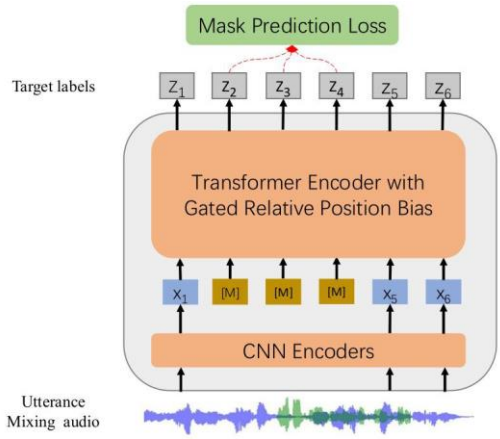
راه حل کلی این مدل به این صورت است که به کمک یک مدل مبدل-پایه دنباله-به-دنباله[۷۲] (شکل ۱۶) با ورودی داده گفتاری آموزش می‌بینند. این مدل می‌تواند وظایف دیگری علاوه بر بازشناسی گفتار را انجام دهد که به‌عنوان

مثال می‌توان به ترجمه گفتاری، شناسایی زبان و شناسایی سکوت در صوت اشاره کرد.

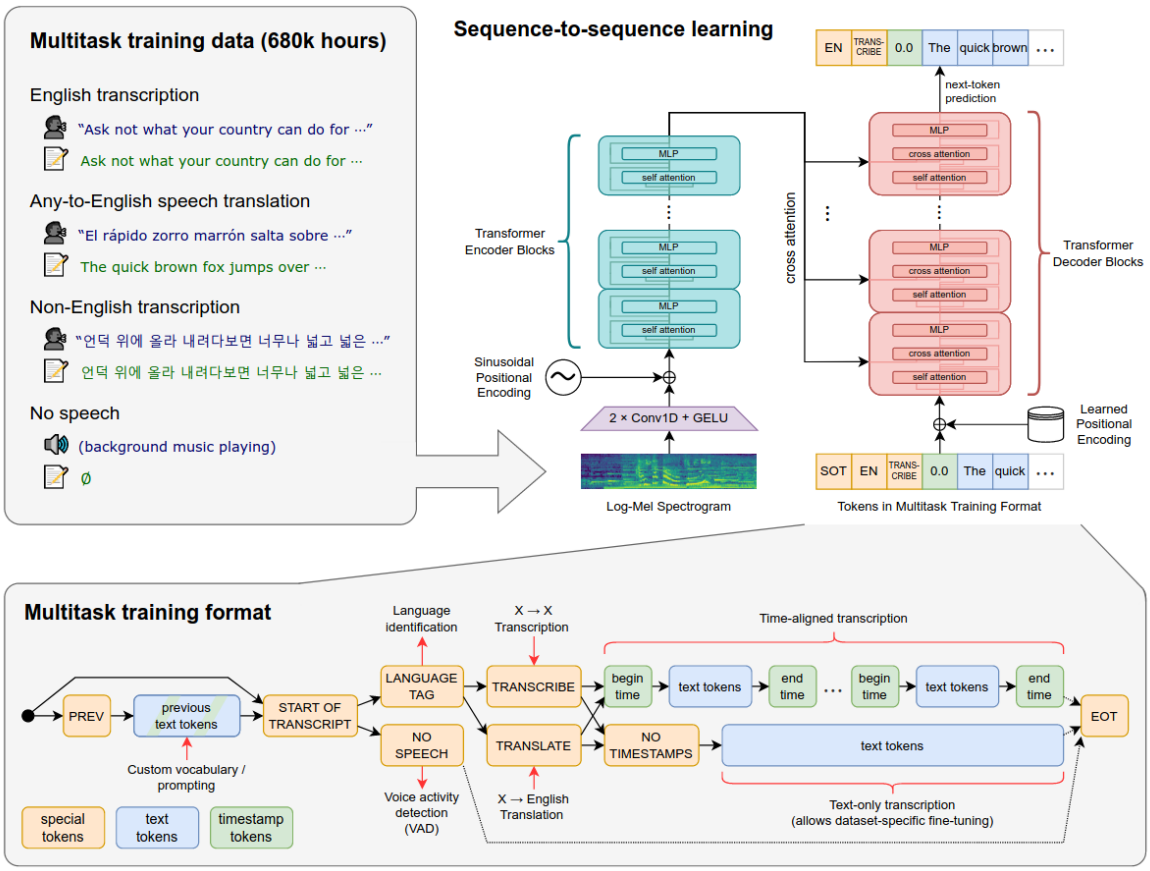
۱۰. جمع‌بندی و نتایج عملکرد

در این مقاله، روش‌های نوین برای حل مساله بازشناسی گفتار توضیح داده شدند. در این‌جا نتایج عملکرد روش‌های ذکر شده را بر روی دادگان سوییچ بورد و ژورنال وال‌استریت (WSJ) ارائه می‌کنیم. اگر ارزیابی توسط ما انجام نشده باشد، مرجع هر کدام از اعداد کنار آن درج شده است. تمام نتایج در جدول ۱ نشان داده شده است. در ستون اول این جدول در کنار نام روش، شماره بخش مربوطه در این مقاله نیز آمده است. روش‌های MMI و گونه‌های دیگر آن (یعنی MPE، MWE و sMBR) همگی قبلاً بر بستر HMM-GMM ارائه و ارزیابی شده‌اند ولی در چند سال اخیر با مرسوم شدن شبکه‌های عصبی ژرف، بر بستر HMM-DNN هم مورد بررسی و ارزیابی قرار گرفته‌اند. همان‌طورکه در جدول مشاهده می‌شود، بهبود حاصل از آموزش شبکه با این توابع هدف قابل توجه است (۷٪ نسبی). از طرف دیگر، روش LF-MMI نیز یکی از گونه‌های MMI است ولی فقط با HMM-DNN ارزیابی شده است و تفاوت زیادی با گونه‌های دیگر دارد. از مهم‌ترین این تفاوت‌ها این است که توپولوژی واج بسیار کوچکتر شده است و فقط با یک گذر می‌توان از آن عبور کرد. همچنین این روش برخلاف گونه‌های دیگر از نقطه صفر شبکه را آموزش می‌دهد و نه از مدل CE. به‌علاوه، گرچه در این روش از هم ترازی‌های قبلی استفاده می‌شود ولی این استفاده به‌صورت غیرمستقیم است و صرفاً برای ساخت گراف صورت می‌باشد. به عبارتی، در تابع هدف این روش، هم ترازی‌های نرم برای گراف صورت محاسبه می‌شوند (در گونه‌های دیگر از هم ترازی‌های سخت ثابت استفاده می‌شود). بهبود این روش قابل توجه است (۶٪ نسبی). در این جدول، عملکرد روش CTC نمایش داده شده است. همان‌طورکه مشاهده می‌شود، دقت این روش در شرایط برابر به میزان قابل توجهی از روش‌های دیگر بدتر است ولی همچنان از نظر عملی ارزشمند است چراکه به هیچ مدل قبلی نیاز ندارد و درنتیجه هزینه محاسباتی آن در مجموع کمتر است. در انتها لازم به توضیح است که دادگان سوییچ‌برد محاوره‌ای است و بازشناسی روی آن بسیار مشکل‌تر از WSJ می‌باشد.

در کل، می‌توان گفت که نقطه قوت توابع هدف تمایزی HMM-DNN این است که احتمال مسیرهای غلط را کاهش می‌دهند. نقطه ضعف آنها، محاسبات زیاد آنها و وابستگی به مدل قبلی است، گرچه مشکل وابستگی در روش LF-MMI کمتر شده است. از طرف دیگر، مزیت روش CTC محاسبات کمتر و عدم وابستگی به مدل قبلی است و نقطه ضعف آن، عملکرد نسبتاً ضعیف در LVCSR است.



شکل۱۵. ساختار مدل ویوال‌ام [۶۵]



شکل ۱۶. ساختار مدل ویسپر [۶۶]

جدول۱. مقایسه روش‌های پیشین براساس نرخ خطای کلمه و ویژگی‌های تابع هدف (برای این نتایج، از دادگان آموزشی ۱۷۰۰ ساعته فیشر نیز استفاده شده است. برای این روش‌ها دقت به ازای آموزش روی سوییچ‌برد تنها گزارش نشده است.)

روش	سال	نوع تابع هدف	نوع مدل احتمالاتی	مدل قبلی موردنیاز	نرخ خطای کلمه روی سوییچ‌برد	نرخ خطای کلمه روی WSJ
روش پایه CE ۳	۲۰۱۰	تولیدی	تولیدی	هم‌ترازی از مدل HMM-GMM	[۱۴] ۱۸/۲	۵/۶۹
MMI ۱.۴	۲۰۱۳	تمایزی	تولیدی	هم‌ترازی و مقداردهی اولیه از مدل CE	[۱۴] ۱۶/۹	۵/۱۵
LF-MMI ۲.۴	۲۰۱۶	تمایزی	تولیدی	هم‌ترازی از مدل HMM-GMM	[۳۸] ۱۵/۹	۴/۱
CTC ۵	۲۰۱۴	تمایزی	تمایزی	–	[۶۷] ۲۱/۰	[۴۴] ۸/۷
CTC-WER ۲.۵	۲۰۱۴	تمایزی	تمایزی	–	–	[۴۴] ۸/۲
RNN-T ۶	۲۰۱۸	تمایزی	تمایزی	–	[۵۲]*۱۳/۲	[۴۴] ۸/۲
Attention ۷	۲۰۱۸	تمایزی	تمایزی	–	[۵۲]*۱۳/۵	[۶۸] ۶/۷

جدول ۲. نتایج نرخ خطای کلمه برای نسخه‌های مختلف مدل‌های از پیش آموزش دیده روی نسخه تمیز شده دادگان [۶۹]

نام مدل	نرخ خطای کلمه(%)
Conformer (xlarge)	**۱/۶۲**
Conformer (large)	۱/۷۰
HuBERT (xlarge)	۱/۷۱
HuBERT (large)	۱/۸۰
Wav2vec2 (large)	۱/۸۴
Whisper (large)	۳/۰۰
WavLM (base)	۸/۰۶

در جدول ۲، نرخ خطای کلمه برای روش‌های از پیش آموزش دیده که در بخش ۹ توضیح داده شدند برای مقایسه ارائه شده است. لازم به توضیح است که هریک از این مدل‌ها دارای نسخه‌های پیاده‌سازی شده متعددی هستند و اعداد نرخ خطای کلمه گزارش شده برای نمونه‌هایی از نسخه‌های پیاده‌سازی شده است. این ارزیابی‌ها روی نسخه تمیز شده دادگان Librispeech انجام شده است.

۱۱. نتیجه‌گیری

بازشناسی گفتار به‌عنوان یک زیرشاخه از هوش مصنوعی که خود شاخه‌ای از علم رایانش است تحول زیادی در طی ۴ دهه اخیر به خود دیده‌است. این تغییرات متأثر از سامانه‌ها و ابزار پردازشی جدید مورد استفاده در هوش مصنوعی به‌صورت کلی است. همچنین دسترسی روزافزون به داده‌های انبوه و امکان استفاده از داده‌های بدون برچسب و نیز شکل‌گیری سامانه‌های مبتنی بر آموزش بی‌نظارت و نیز امکان آموزش خودنظارتی مدل‌ها نیز مانند سایر سامانه‌های هوش مصنوعی تأثیر جدی روی مدل‌های بازشناسی گفتار داشته‌اند. حرکت کلی بازشناسی گفتار از مدل‌های آماری (به طور خاص مدل مخفی مارکوف) به سمت تلفیق با شبکه‌های ژرف (HMM-DNN) و سپس کنار گذاشتن مدل مخفی مارکوف بوده است. پس از آن برای مدل‌سازی بهتر مشخصات محلی و زمانی گفتار ساختارهای مختلفی از شبکه‌های ژرف در بازشناسی گفتار مانند CTC , RNN-T، توجه–پایه و CNN استفاده شده و نتایج خوبی به بار آورده است. تحول جدی بعدی استفاده از مبدل‌ها و مدل‌های از پیش آموزش دیده هستند. نمونه‌های حال حاضر این مدل‌ها شامل کانفورمر، ویوتووک، ویوال‌ام، هوبرت و ویسپر هستند. این مدل‌ها به همراه امکان آموزش نظارتی ضعیف و خود نظارتی و نیز حجم عظیم داده‌های صوتی در دسترس در کنار امکان افزون‌سازی داده‌ها رشد عملکرد مطلوب سامانه‌های بازشناسی گفتار را به شدت سرعت داده است. نتیجه این تحولات حرکت به سمت پیاده‌سازی سامانه‌های واحد برای انواع کاربردهای پردازش گفتار مانند شناسایی گوینده، واژه‌یابی، جداسازی گویندگان، ترجمه گفتاری و مانند آن به‌صورت توام با بازشناسی گفتار بوده است. از دیگر آثار این تحولات توسعه سریع سامانه‌های بازشناسی گفتار به زبان‌های جدید و نیز مدل‌های چندزبانه بوده است. جهت گیری پژوهش در این زمینه و نیز پیاده‌سازی مدل‌های جدید علاوه بر ادامه مسیر طراحی مدل‌های چند منظوره و چندزبانه، معطوف به حل مسائل قدیمی‌تر ولی سخت‌تری مانند بهسازی گفتار، جداسازی و بازشناسی گفتار مخلوط، بازشناسی گفتار مقاوم به نویز محیط، حل مشکل میکروفون دوردست و مانند آن خواهد بود.

اقرارنامه

این مقاله در دوازدهمین کنفرانس بین المللی آکوستیک و ارتعاشات (ISAV 2022) در تاریخ ۲۳ آذر ۱۴۰۱ به‌عنوان سخنرانی مدعو توسط آقای دکتر حسین صامتی، بصورت حضوری و شفاهی، ارایه شده است.

۱۲. مآخذ

[1] X. Huang, A. Acero, H.-W. Hon, and R. Foreword By-Reddy, *Spoken language processing: A guide to theory, algorithm, and system development*. Prentice Hall PTR, 2001.

[2] "how the human auditory system works," 2020. https://www.khouzeyannews.ir/blog/how-the-human-auditory-system-works.

[3] M. Gales and S. Young, "The application of hidden Markov models in speech recognition," *Found. trends signal Process.*, Vol.1, no.3, 2008, pp.195–304.

[4] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol.1, 1992, pp.13–16.

[5] C. D. Manning and H. Schütze, *Foundations of statistical natural language processing*, Vol.999. MIT Press, 1999.

[6] M. Mohri, F. Pereira, and M. Riley, "Speech recognition with weighted finite-state transducers," in *Springer Handbook of Speech Processing*, Springer, 2008, pp.559–584.

[7] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Comput. Speech Lang.*, 2002, Vol.16, no.1, pp.69–88.

[8] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *IEEE workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584, IEEE Signal Processing Society, 2011.

[9] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, 1989, Vol.77, no.2, pp.257–286.

[10] D. Povey, L. Burget, M. Agarwal, P. Akyazi, F. Kai, A. Ghoshal, O. Glembek, N. Goel, M. Karafiát, A. Rastrow *et al.*, "The subspace Gaussian mixture model—A structured model for speech recognition," *Comput. Speech Lang.*, 2011, Vol.25, no.2, pp.404–439.

[11] B.-H. Juang, S. Levinson, and M. Sondhi, "Maximum likelihood estimation for multivariate mixture observations of Markov chains (corresp.)," *IEEE Trans. Inf. Theory*, 1986, Vol.32, no.2, pp.307–309.

[12] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proceedings of the workshop on Human Language Technology*, 1994, pp.307–312.

[13] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[14] K. Vesel`y, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks.," in *Proceedings of the Interspeech,* 2013, pp.2345–2349.

[15] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, 2012, Vol.29, no.6, pp.82–97.

[16] H. A. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*, Vol.247. Springer Science & Business Media, 1994.

[17] H. Bourlard and N. Morgan, "Continuous speech recognition by connectionist statistical methods," *IEEE Trans. Neural Networks*, 1993, Vol.4, no.6, pp.893–909.

[18] A. Senior, G. Heigold, M. Bacchiani, and H. Liao, "GMM-free DNN acoustic model training," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp.5602–5606.

[19] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio. Speech. Lang. Processing*, 2012, Vol.20, no.1, pp.30–42, 2012.

[20] L. Deng , J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams *et al.*, "Recent advances in deep learning for speech research at Microsoft," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp.8604–8608.

[21] L. Deng and D. Yu, "Deep convex net: A scalable architecture for speech pattern classification," in *Proceedings of the Interspeech,* 2011.

[22] O. Abdel-Hamid, A. Mohamed, H. Jiang, and G. Penn, “Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition,” in *2012 IEEE international conference on Acoustics, speech and signal processing (ICASSP)*, 2012, pp.4277–4280.
[23] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” *Neural Comput.*, 2006, Vol.18, no.7, pp.1527–1554.
[24] Z. Huang, J. Li, C. Weng, and C.-H. Lee, “Beyond cross-entropy: towards better frame-level objective functions for deep neural network training in automatic speech recognition.,” in *Proceedings of the Interspeech,* 2014, pp.1214–1218.
[25] D. Povey, “Discriminative training for large vocabulary speech recognition,” Ph.D. dissertation, University of Cambridge, 2005.
[26] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young, “Lattice-based discriminative training for large vocabulary speech recognition,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1996, Vol.2, pp.605–608.
[27] V. Doumpiotis and W. Byrne, “Lattice segmentation and minimum Bayes risk discriminative training for large vocabulary continuous speech recognition,” *Speech Commun.*, 2006, Vol.48, no.2, pp.142–160.
[28] L. Bahl, “Maximum mutual information estimation of hidden Markov model parameters for speech recognition,” in *Proceedings of ICASSP*, 1986, pp.701–704.
[29] P. C. Woodland and D. Povey, “Large scale discriminative training for speech recognition,” in *Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.
[30] P. C. Woodland and D. Povey, “Large scale discriminative training of hidden {Markov} models for speech recognition,” *Comput. Speech Lang.*, 2002, Vol.16, no.1, pp.25–47.
[31] M. Gibson and T. Hain, “Hypothesis spaces for minimum Bayes risk training in large vocabulary speech recognition.,” in *Proceedings of the Interspeech,* 2006.
[32] B. Kingsbury, “Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling,” in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2009, pp.3761–3764.
[33] G. Saon, H.-K. J. Kuo, S. Rennie, and M. Picheny, “The IBM 2015 English Conversational Telephone Speech Recognition System,” in *Proceedings of the Interspeech,* 2015.
[34] M. J. F. Gales and S. J. Young, “Robust continuous speech recognition using parallel model combination,” *IEEE Trans. Speech Audio Process*, 1996, Vol.4, no.5, pp.352–359.
[35] Nvidia, “Cuda C Programming guide.” *https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html*, 2011.
[36] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang, “Phoneme recognition using time-delay neural networks,” *IEEE Trans. Acoust., Speech, Signal Proc.*, 1989, Vol.37, no.3, pp.328–339.
[37] H. Sak, A. Senior, K. Rao, and F. Beaufays, “Fast and accurate recurrent neural network acoustic models for speech recognition,” *arXiv Prepr. arXiv1507.06947*, 2015.
[38] D. Povey *et al.*, “Purely sequence-trained neural networks for ASR based on lattice-free MMI,” in *Proceedings of the Interspeech,* 2016.
[39] H. Su, G. Li, D. Yu, and F. Seide, “Error back propagation for sequence training of context-dependent deep networks for conversational speech transcription,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013, pp.6664–6668.
[40] A. Zeyer, E. Beck, R. Schlüter, and H. Ney, “{CTC} in the Context of Generalized Full-Sum {HMM} Training,” in *Proceedings of the Interspeech,* 2017, pp.944–948.
[41] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, “Flat-start single-stage discriminatively trained HMM-based models for ASR,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 2018, Vol.26, no.11, pp.1949–1961.
[42] H. Hadian, H. Sameti, D. Povey, and S. Khudanpur, “End-to-end Speech Recognition Using Lattice-free MMI\.,” in *Proceedings of the Interspeech,* 2018, pp.12–16.

[43] C. Bishop, and N. M. Nasrabadi. *Pattern recognition and machine learning.* Vol. 4, no. 4. New York: springer, 2006.
[44] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning*, 2014, pp.1764–1772.
[45] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp.369–376.
[46] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp.167–174.
[47] A. Hannun *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv Prepr. arXiv1412.5567*, 2014.
[48] A. Y. Hannun, A. L. Maas, D. Jurafsky, and A. Y. Ng, "First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs," *arXiv Prepr. arXiv1408.2873*, 2014.
[49] A. Maas, Z. Xie, D. Jurafsky, and A. Ng, "Lexicon-free conversational speech recognition with neural networks," in *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp.345–354.
[50] A. Graves, "Sequence transduction with recurrent neural networks," *arXiv Prepr. arXiv1211.3711*, 2012.
[51] K. Rao, H. Sak, and R. Prabhavalkar, "Exploring architectures, data and units for streaming end-to-end speech recognition with {RNN}-transducer," in *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding Workshop (ASRU), IEEE*, 2017, pp.193–199.
[52] E. Battenberg *et al.*, "Exploring neural transducers for end-to-end speech recognition," *arXiv Prepr. arXiv1707.07413*, 2017.
[53] W. Chan, N. Jaitly, Q. V Le, and O. Vinyals, "Listen, attend and spell," *arXiv Prepr. arXiv1508.01211*, 2015.
[54] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent {NN}: first results," *arXiv Prepr. arXiv1412.1602*, 2014.
[55] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp.4945–4949.
[56] I. Sutskever, O. Vinyals, and Q. V Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp.3104–3112.
[57] K. Cho *et al.*, "Learning Phrase Representations using RNN Encoder--Decoder for Statistical Machine Translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734, doi: 10.3115/v1/D14-1179.
[58] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv Prepr. arXiv1409.0473*, 2014.
[59] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Trans. audio, speech, Lang. Process.*, 2014, Vol.22, no.10, pp.1533–1545.
[60] A. Gulati, J. Qin, C.C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition." *arXiv preprint arXiv:2005.08100*, 2020.
[61] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "Wav2vec: Unsupervised pre-training for speech recognition," *arXiv Prepr. arXiv1904.05862*, 2019.
[62] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," *Adv. Neural Inf. Process. Syst.*, 2020, Vol.33, pp.12449–12460.
[63] A. Vaswani *et al.*, "Attention is all you need," in *Proceedings of Advances in Neural Information Processing Systems*, 2017, pp.6000–6010.
[64] W.N. Hsu, B. Bolte, Y.H.H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert:

Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, 2021,Vol.29, pp.3451–3460.
[65] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, and J. Wu, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE J. Sel. Top. Signal Process.*, 2022, Vol.16, no.6, pp.1505–1518.
[66] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," *arXiv Prepr. arXiv2212.04356*, 2022.
[67] Y. Miao, M. Gowayyed, X. Na, T. Ko, F. Metze, and A. Waibel, "An empirical exploration of CTC acoustic models," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp.2623–2627.
[68] J. Chorowski and N. Jaitly, "Towards better decoding and language model integration in sequence to sequence models," in *Proceedings of the Interspeech,* 2017.
[69] P. with Code, "Automatic Speech Recognition on Librispeech (clean)." 2022, Accessed: Mar. 18, 2023. [Online]. Available: *https://paperswithcode.com/sota/automatic-speech-recognition-on-librispeech-7*.

پی‌نوشت

1. Bias
2. Hybrid
3. Deep Neural Network-Hidden Markov Model
4. Lattice Free-Maximum Mutual Information
5. Connectionist Temporal Classification
6. Recurrent Neural Network-Transducer
7. Attention
8. Convolutional Neural Network
9. Observation sequence
10. Frame sequence
11. Mel-frequency cepstral coefficients
12. Linear discriminant analysis
13. Large Vocabulary Continuous Speech Recognition
14. Decoding
15. Viterbi
16. Weighted finite state transducers
17. Kaldi
18. Segmentation
19. Context independent
20. Context dependent
21. State tying
22. Subspace Gaussian mixture model

۲۳. بیان یا بیان آموزشی عبارت است از یک قطعه سیگنال گفتار و متن معادل آن که برای آموزش مدل آکوستیکی استفاده می‌شود. در دادگان‌های گفتار، معمولاً طول هر بیان ۱ تا ۱۵ ثانیه (۱ تا ۴۰ کلمه) می‌باشد. متن معادل می‌تواند در سطح کلمه یا واج باشد.

24. Expectation maximization
25. Spectrogram
26. Hybrid
27. Cross entropy
28. Tied state
29. State posterior
30. Activation
31. KL-divergence
32. Cross validation
33. Pre-training
34. Contrastive divergence
35. Restricted Boltzmann machine
36. Maximum mutual information
37. Joint entropy
38. Conditional entropy
39. Minimum phone error
40. Minimum word error
41. State-level minimum Bayes risk

42. Forward-backward
43. Soft alignment
44. Hard alignment
45. GPU: Graphics processing unit
46. Caching
47. CUDA kernels
48. Blank
49. Sum-product
50. Long short-term memory
51. Bidirectional LSTM (BLSTM)
52. Beam search
53. Recurrent Neural Network Transducer
54. Transcription
55. Prediction
56. Listen, attend, spell
57. Attention-based
58. Sequence-to-sequence
59. encoder
60. decoder
61. Listener
62. Speller
63. Transformers
64. Conformer
65. Wav2Vec
66. HuBERT
67. Pseudo-labeling
68. WavLM
69. Gated relative position bias
70. Whisper
71. Weak supervision
72. Sequence-to-sequence