بهینه‌سازی مدل آمیزه‌های گوسی با استفاده از شبکه‌های عصبی برای تشخیص زبان فارسی از سایر زبان‌ها در منابع اطلاعاتی گفتاری

علی شادمند[۱]، رامین شقاقی کندوان[۲]، یاشار اعتماد[۳]، فربد رزازی[۴]

۱ـ دانشجوی کارشناسی ارشد، گرایش مخابرات سیستم، دانشگاه آزاد اسلامی واحد نجف آباد، Shadmand.ali@Gmail.com

۲- عضو هیأت علمی دانشگاه آزاد اسلامی واحد شهرری، ramin_shaghaghi@yahoo.com

۳- دانشجوی کارشناسی ارشد، گرایش مخابرات سیستم، دانشگاه آزاد اسلامی واحد علوم و تحقیقات، Yashar.Etemad@Gmail.com

۴- عضو هیأت علمی دانشگاه آزاد اسلامی واحد علوم و تحقیقات، Razzazi@sr.iau.ac.ir

چکیده

تشخیص خودکار زبان از سیگنال صحبت شامل الگوریتم‌ها و روش‌هایی است که برای مدل‌کردن و دسته‌بندی کردن زبان‌های مختلف مورد استفاده قرار می‌گیرند. مدل آمیزه‌های گوسی (GMM)[۱] را می‌توان به‌عنوان ابزاری قدرتمند در دسته‌بندی بردارهای ویژگی برای زبان‌های مختلف استفاده کرد. تکنیک تشخیص زبان فارسی از سایر زبان‌ها، با استفاده از مدل آمیزه‌های گوسی به‌عنوان سیستم پایه برای نشانه‌گذاری و شبکه‌های عصبی به‌عنوان پردازشگر پسین عملی می‌باشد. بانک صوتی مورد نیاز در این تحقیق از چندین کانال تلویزیونی ماهواره‌ای متنوع جمع‌آوری گردیده است. نتایج برای یک سیستم ترکیبی از مدل آمیزه‌های گوسی نشانه‌گذار[۲] با شبکه‌های عصبی و عدم وجود آن مقایسه شده است. در نهایت نشان داده می‌شود که استفاده از شبکه‌های عصبی به عنوان پردازشگر پسین[۳] نتیجه تشخیص زبان را بهبود می‌بخشد.

واژه‌های کلیدی

بردارهای ویژگی، تشخیص زبان فارسی، مدل آمیزه‌های گوسی، نشانه‌گذار، شبکه‌های عصبی

۱– مقدمه

فرآیند تشخیص زبان فارسی، فرآیندی است که در آن زبان فارسی از سایر زبان‌ها توسط کامپیوتر، تشخیص داده می‌شود. تکنیک‌هایی که امروز مورد استفاده برای تشخیص زبان بر پایه تشخیص همین نشانه‌های سطح پایین بنا نهاده شده‌اند، به دو دسته عمده تقسیم می‌شوند: سیستم‌های تشخیصی زبان بر پایه ویژگی‌های طیفی نظیر مدل آمیزه‌های گوسی (GMM) و ماشین‌های بردار پشتیبان[۴] (SVM) و سیستم‌های تشخیص زبان بر پایه توالی نشانه‌ها نظیر روش‌های PRLM[۵] و PPRLM.[۶] استفاده از روش GMM، به علت نرخ بالای شناسایی و نیز وجود روش‌های بهینه برای تخمین پارامترهای آن و همچنین خطای بیشتر روش‌های کلاسیک دیگر در پردازش گفتار مورد توجه واقع شده است. مهم‌ترین اجزای این مبحث، شامل افزودن مشخصه غیرایستابودن گفتار به GMM، همگراشدن سریع پارامترها در هنگام آموزش و چگونگی مرزبندی فضای ویژگی‌ها است.

نتایج مدل آمیزه‌های گوسی بدون استفاده از نشانه‌گذار در مرجع [۱] آمده است. همچنین خاطرنشان شده است که ترکیب مدل آمیزه‌های گوسی با نشانه‌گذار نتایج خوب و قابل مقایسه‌ای را مانند سیستم PPRLM از خود نشان می‌دهد[۱]. عملکرد بهتر مدل آمیزه‌های گوسی در مراجع[۴،۵و۷] آمده است. در این مقاله نشان داده می‌شود که ترکیب شبکه عصبی با مدل آمیزه‌های گوسی با

نشانه‌گذار می‌تواند عملکرد بهتری در مقایسه با مدل آمیزه‌های گوسی با نشانه‌گذار بدون شبکه عصبی باشد.

انگلیسی، عربی، فرانسوی، آلمانی و ترکی. مباحث مطرح شده در این مقاله شامل ضرایب شیفت‌یافته دلتاهای کپسترال، نشانه‌گذار، مدل آمیزه‌های گوسی، ترکیب مدل آمیزه‌های گوسی با شبکه‌های عصبی ، نتایج روش پیشنهادی و نتیجه‌گیری می‌باشد.

۲– فرآیند استخراج بردارهای ویژگی

در شکل (۱) فرآیند استخراج بردارهای ویژگی نشان داده شده است. یکی از مشخصه‌های سیگنال صحبت این است که در گذر زمان ثابت نیست، یعنی نمی‌توان آن را به‌عنوان یک فرآیند ایستان در نظرگرفت. برای اینکه بتوانیم از روش‌های ایستان برای تحلیل آن استفاده کنیم سیگنال را فریم به فریم می‌کنیم. پس از فریم‌بندی، فریم‌ها را ایستان در نظر می‌گیریم. هر فریم شامل ۲۰ میلی ثانیه از صحبت است که در هر ۱۰ ثانیه تکرار می‌شود. البته می‌تواند ثابت هم نباشد اما از لحاظ زمانی باید در محدوده ۲۰ میلی ثانیه باشد تا ایستان باشد و بستگی به فرکانس نمونه‌برداری و فرکانس صحبت دارد و چون بعداً از سیگنال DFT گرفته می‌شود معمولاً طوری انتخاب می‌شود که نمونه‌ها مجذور کامل باشند.

سیگنال صحبت از پنجره Hamming به طول ۲۰ میلی ثانیه عبور داده می‌شود. نقش پنجره Hamming ملایم‌کردن لبه‌های زمان صحبت برای هر گوینده در تهیه پایگاه داده ۳۰ ثانیه می‌باشد. زبان‌های جمع‌آوری‌شده در پایگاه داده عبارتند از فارسی، سیگنال است. چون هر فریم تکه‌ای از سیگنال صحبت است لذا لبه‌های آن سخت (Hard) می‌باشد.

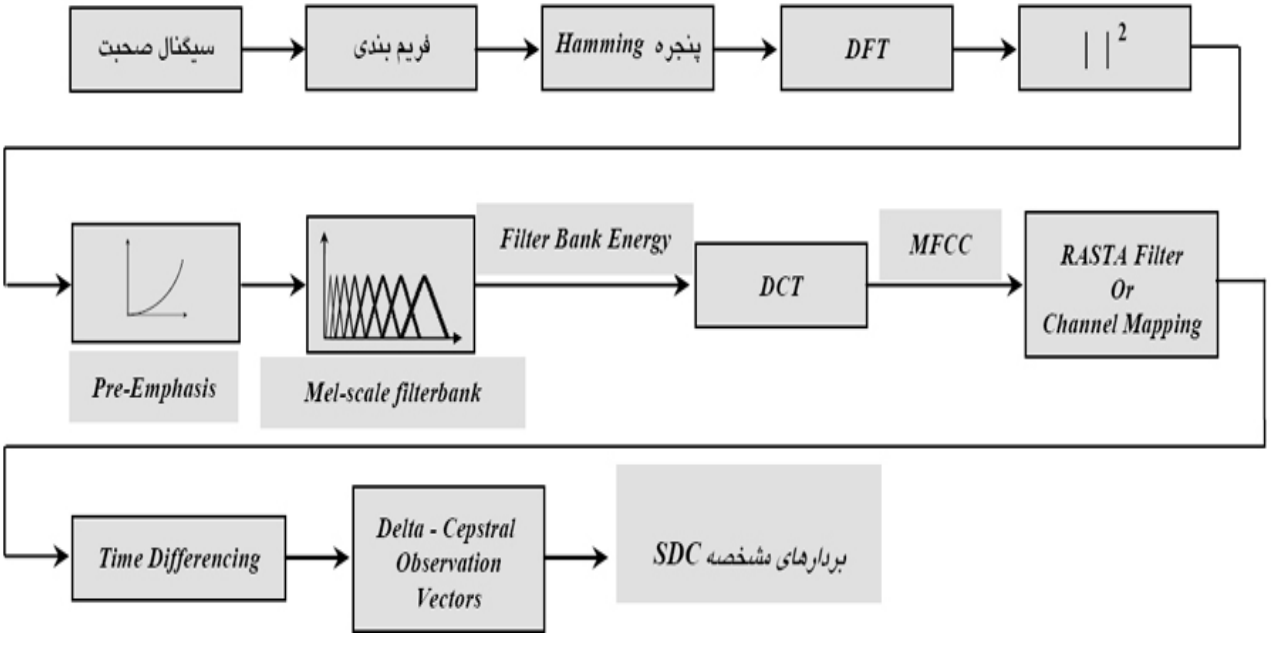
پنجره Hamming با ضرب شدن در سیگنال لبه‌های آن را نرم (Soft) می‌کند. در واقع وزن فرکانس‌ها را نرمالیزه می‌کند. شکل (۲) پنجره Hamming را نشان می‌دهد. ضرایب پنجره Hamming با معادله زیر محاسبه می‌شوند:

$$w[k+1] = 0.54 - 0.46\cos(2\pi \frac{k}{n-1}), k = 0,1,...,n-1 \qquad (۱)$$

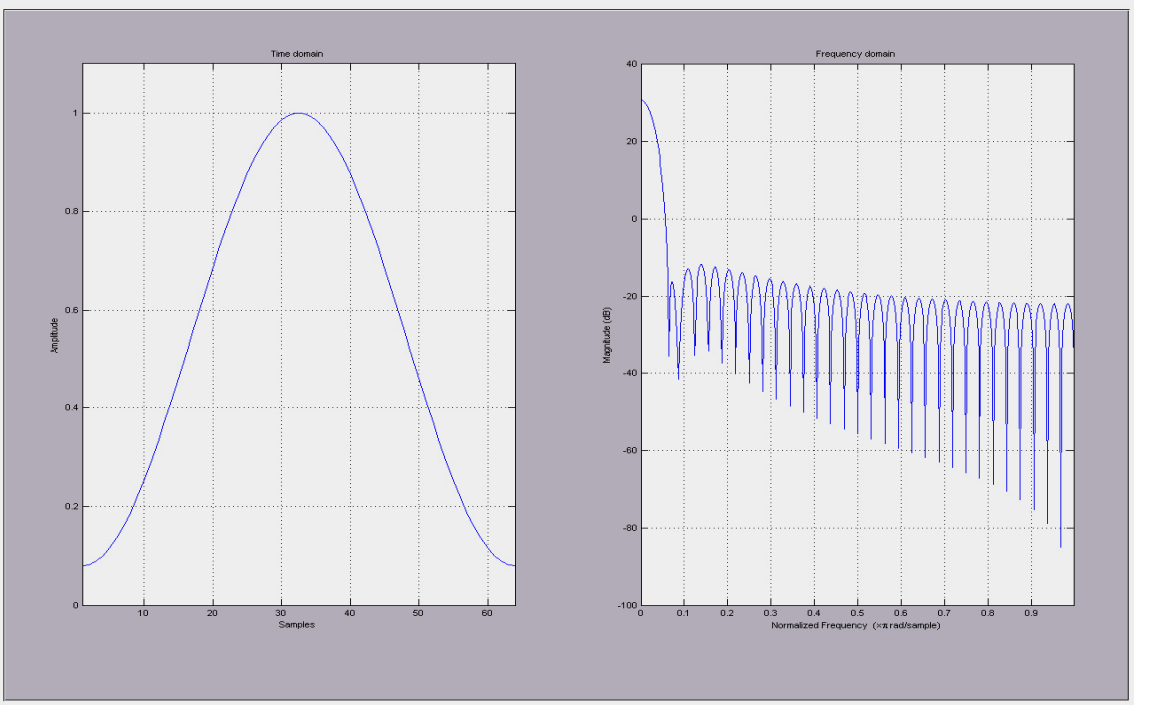
هدف از DFT استخراج طیف سیگنال می‌باشد. سپس اندازه سیگنال را به توان ۲ می‌رسانیم. مجرای صوتی انسان ذاتاً طوری است که در فرکانس‌های بالای صحبت دچار تضعیفی معادل 20dB/decade می‌شود که باعث می‌شود اطلاعات فرکانس بالای گفتار با دامنه کمی ظاهر شوند. فیلتر پیش تأکید با تابع تبدیل

$$H_{pre}(z) = 1 + a_{pre} z^{-1} \qquad (۲)$$

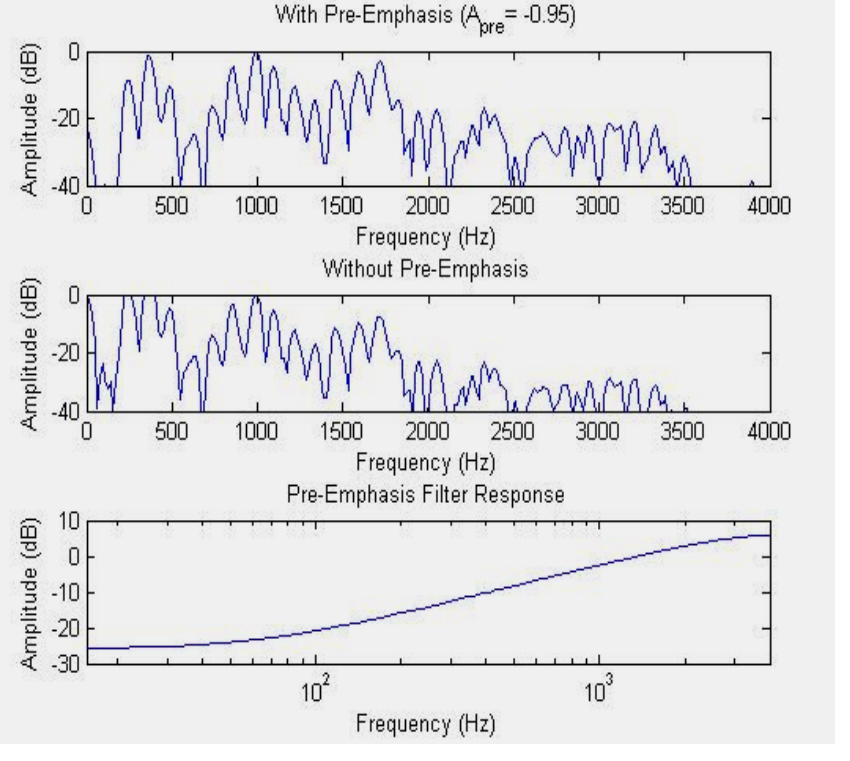
باعث هموارشدن فرکانس‌های بالای طیف سیگنال گفتار می‌شود. در شکل (۳) اثر فیلتر پیش تأکید روی فرکانس‌های بالا نشان داده شده است.



شکل ۱- فرآیند استخراج بردارهای ویژگی



شکل ۲- پنجره Hamming

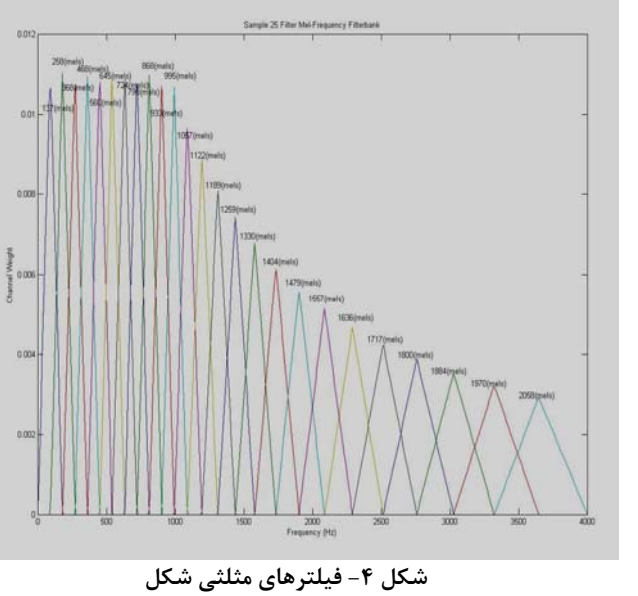


شکل ۳- اثر فیلتر پیش تأکید روی فرکانس‌های بالا

تقریب مقیاس فرکانس Mel کار نگاشت باند شنوائی را برعهده دارد. در واقع با این کار می‌خواهیم حساسیت گوش را به فرکانس‌ها مدل‌سازی کنیم. حساسیت گوش به بعضی از فرکانس‌ها زیاد و به بعضی از آنها کم است. اگر در F_{mel} سیگنال صحبت را رسم کنیم تعدادی فیلتر مثلثی با هم‌پوشانی مساوی و دامنه‌های برابر خواهیم داشت. اما اگر در طیف خطی بخواهیم آن را رسم کنیم قبل از فرکانس ۱۰۰۰ کیلو هرتزخطی و بعد از آن بازتر خواهد شد. لازم به ذکر است که فرکانس‌های مرکزی فیلترهای مرکزی در مقیاس Mel در فواصل ۱۰۰ هرتز از هم قرار دارند. پهنای باند فیلتر مثلثی به گونه‌ای است که فرکانس پایین‌گذر و بالاگذر هر فیلتر مثلثی روی فرکانس مرکزی فیلترهای مجاور قرار بگیرند. بدین‌ترتیب پهنای باند فیلتر در مقیاس خطی به‌تدریج افزایش می یابد ولی در مقیاس Mel پهنای باند فیلتر ها ثابت می‌باشد. در شکل ۴ فیلترهای مثلثی نشان داده شده‌اند.

تعریف فیلترهای مثلثی در حوزه K به‌صورت زیر می‌باشد:

$$F_\ell[k]=\begin{cases}\dfrac{((\frac{k}{n})f_s-f_{c\ell-1})}{f_{c\ell}-f_{c\ell-1}}, L_\ell\le k\le C_\ell\\ \dfrac{(f_{c\ell+1}-(\frac{k}{n})f_s)}{f_{c\ell+1}-f_{c\ell}}, C_\ell\le k\le U_\ell\end{cases} \qquad (۳)$$



شکل ۴- فیلترهای مثلثی شکل

فیلترهای مثلثی شکل را در طیف سیگنال ضرب می‌کنیم و عدد بدست آمده انرژی آن فیلتر است . با اعمال DCT کورولیشن بین فیلترهای مثلثی شکل از بین می‌رود. در واقع DCT نقش یک Decorrelator را دارد. خروجی DCT، MFCC می‌باشد.

MFCC شامل ضرایب C_0 تا C_d می‌باشد که C_0 همان انرژی کل (Total Energy) می‌باشد. به‌علت این‌که وابسته به Tone صحبت هر گوینده می‌باشد لذا از آن صرف‌نظر می‌شود. در واقع به‌علت این‌که افراد موقع صحبت‌کردن خود به خود صدای خود را بالا و پایین می‌آورند لذا برای اینکه Tone صدا در مدل‌سازی تأثیرگذار نباشد C0 حذف می‌شود. البته لازم به ذکر است که از C_0 برای حذف سکوت استفاده می‌شود . چون C_0 نماینگر انرژی کل می‌باشد لذا به‌عنوان آستانه در نظر گرفته می‌شود و انرژی فریم‌هایی که انرژی آنها از انرژی آستانه پایین‌تر باشند حذف می‌شوند.

نقش فیلتر RASTA حذف اثرات نویز کانولوشنال هست. همان‌طور که می دانیم پاسخ ضربه میکروفن یا دهنی تلفن کاملاً خطی نیست و با توجه به خطی‌نبودن آن در حوزه فرکانس در طیف ضرب و در حوزه زمان با سیگنال کانوال خواهد شد. در حالت ایده‌آل نباید میکروفن یا دهنی تلفن تأثیری روی طیف صحبت بگذارد. در واقع RASTA به‌عنوان یک جبران‌ساز عمل می‌نماید. این فیلتر به‌صورت زیر تعریف می‌گردد:

$$H(z)=0.1\times\frac{2+z^{-1}-z^{-3}-2z^{-4}}{z^{-4}(1-0.98z^{-1})} \qquad (۴)$$

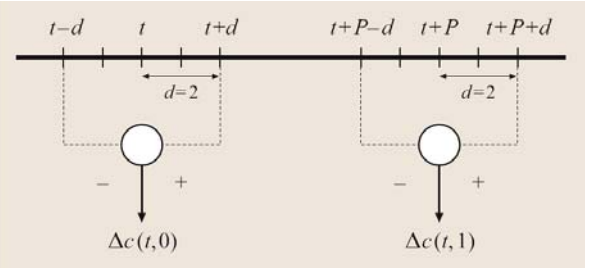
ضرایب این فیلتر همواره ثابت است. ضرایب بدست‌آمده کاملاً تجربی می‌باشند. با گذر از این فیلتر و شیفت زمانی می‌توان به کپسترال و دلتاهای آن دست یافت.

۳- بردارهای ویژگی SDC(Shifted Delta Cepstral)

بردارهای ویژگی عمومی که برای تشخیص زبان استفاده می‌شوند عبارتند از: LPC[۷] ، MFCC[۸] و PLP[۹]. با توجه به نتایج آزمایش‌های انجام شده روی بردارهای ویژگی مختلف در مرجع[۳]، در این تحقیق از بردار ویژگی به نسبت جدیدتری به‌نام SDC[۱۰] استفاده شد. با اعمال شیفت دلتاها به بردارهای ویژگی در هر فریم، بردارهای ویژگی ترکیبی جدید برای هر فریم ایجاد می‌شود.

فرآیند محاسبه بردارهای SDC به‌صورت زیر می‌باشد: ابتدا ضرایب MFCC محاسبه می‌شوند که در این حالت فرض می‌کنیم فاصله بین دلتاهای بردارهای ویژگی آکوستیکی D، فاصله بین بلوک‌ها P ، K تعداد بلوک‌های پی در پی ایجادکننده بردارهای ویژگی شیفت یافته دلتاها و N تعداد ضرایب کپسترال محاسبه شده در هر فریم باشند.

بردارهای آکوستیکی ابتدا به فاصله D نمونه از هم جدا می‌شوند، سپس K تا بردار تفاضلی به اندازه P از هم قرار می‌گیرند که حاصل بردارهای ویژگی جدید خواهد بود. در واقع بردار ویژگی‌های SDC حاصل انباشت k بار دلتا کپسترال می‌باشد که در روابط ۴ و ۵ این انباشت‌ها به بیان ریاضی نشان داده شده‌اند. شکل ۵ نحوه ایجاد دو دلتا کپسترال را نشان می‌دهد.

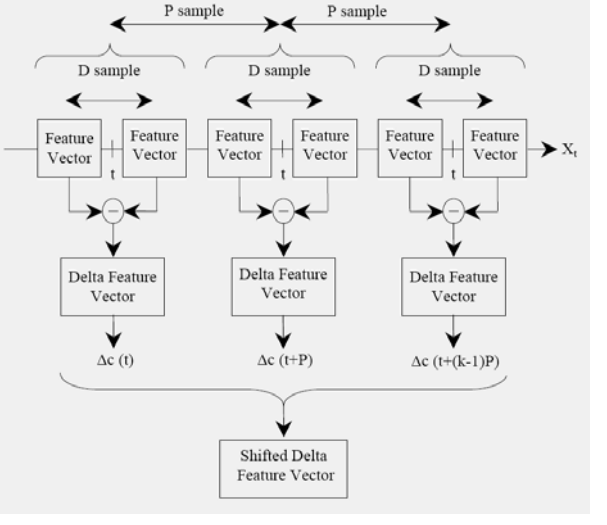


شکل ۵- نحوه ایجاد دو دلتا کپسترال

$$\Delta c(t,i) = c(t+iP+d) - c(t+iP-d) \quad (۵)$$

$$SDC(t) = \begin{pmatrix} \Delta c(t,0) \\ \Delta c(t,1) \\ \vdots \\ \Delta c(t,k-1) \end{pmatrix} \quad (۶)$$

بردارهای ویژگی SDC با استفاده از چهار پارامتر N-d-P-k مشخص می‌شوند که در این راستا با توجه به آزمایش‌های انجام شده قبلی [۳]، از ترکیب ۷-۳-۱-۷ استفاده شد. جزئیات بیشتر در این خصوص، در مرجع شماره [۳] آمده است. شکل۶ نحوه محاسبه بردارهای ویژگی SDC را در حالت کلی نشان می‌دهد.



شکل ۶- نحوه محاسبه بردارهای ویژگی SDC

۴- مدل آمیزه‌های گوسی و نشانه‌گذاری

تابع چگالی آمیزه گوسی مجموعی وزن‌دار از M تابع چگالی جزء است که با رابطه‌ی زیر داده می‌شود:

$$p(\vec{x}|\lambda) = \sum_{i=1}^{M} w_i\, p_i(\vec{x}) \quad (۷)$$

در رابطه‌ی بالا $\vec{x}$ یک بردار تصادفی D بعدی، $p_i(\vec{x})$ توابع چگالی جزء w_i وزن‌آمیزه‌ها می‌باشند. هر تابع چگالی جزء، یک تابع گوسی D متغیره به شکل زیر است:

$$p_i(\vec{x}) = \frac{1}{(2\pi)^{D/2}\left|\sum_i\right|^{1/2}} \exp\left\{-\frac{1}{2}(\vec{x}-\vec{\mu}_i)'\sum_i^{-1}(\vec{x}-\vec{\mu}_i)\right\} \quad (۸)$$

در رابطه بالا $\vec{\mu}_i$ بردار میانگین و $\sum_i$ ماتریس کواریانس جزء iام می‌باشد. وزن آمیزه‌ها به گونه‌ای است که شرط $\sum_{i=1}^{M} w_i = 1$ را برآورده کند تا اینکه آمیزه گوسی بدست آمده یک تابع چگالی احتمال باشد. تقریباً همانند تحلیل فوریه، که با بسط سیگنال‌های سینوسی، سیگنال را بهتر توصیف می‌کنند، مدل آمیزه‌های گوسی نیز با ترکیب چند متغیره گوسین‌ها تمام اطلاعات را به‌طور خلاصه در فضا نمایش می‌دهد.

این مدل تکنیک‌های نیمه پارامتری برای تخمین تابع چگالی احتمال از اطلاعات برچسب‌گذاری شده یا برچسب‌گذاری نشده هستند. آموزش مدل آمیزه‌های گوسی معمولاً با دسته‌بندی k-mean با تکرار چندین مرحله از الگوریتم EM[11] کامل می‌شود. رینولدز بدون بیان علت و احتمالاً براساس تجربه ادعا کرده است که عموماً ۵ تکرار برای همگرایی پارامترها توسط الگوریتم EM کافی است[۱۰]. ولی زو و جردن مدعی شده‌اند که اگر ساختار داده‌ها به گونه‌ای باشد که آمیزه‌های GMM به‌صورت دور از هم واقع‌شده باشند، همگرایی کندتر خواهد بود[۱۱].

در حالت کلی و مخصوصاً برای آموزش تعداد زیادی آمیزه‌هایی که باید طیف وسیعی از کلاس‌های صوتی را پوشش دهد، مقداردهی اولیه صحیح می‌تواند نرخ همگرایی را بالاتر برده و همچنین باعث شود تا الگوریتم EM در ماکزیمم‌های محلی به تله نیفتاده و به ماکزیمم عمومی نزدیک شود[۱۲و۱۳].

یکی از روش‌های مناسب بدین منظور، استفاده از الگوریتم خوشه‌بندی K-means می‌باشد. این الگوریتم طی مراحل زیر داده‌ها را در M خوشه، خوشه‌بندی می‌کند که M همان تعداد آمیزه‌های GMM است.

۴-۱- الگوریتم K-means clustering

K-Means یکی از ساده‌ترین الگوریتم‌های یادگیری بدون نظارت است که مسائل کلاسترینگ معروف را حل می‌کند. این

الگوریتم از یک شیوه ساده برای کلاسیفای کردن یک مجموعه داده در یک تعداد ازپیش مشخص‌شده (k) کلاستر، استفاده می‌کند.

ایده اصلی تعریف k مرکز برای هر یک از کلاسترها می‌باشد .این مراکز بایستی با دقت زیاد انتخاب شوند، زیرا مراکز مختلف، نتایج مختلف را به‌وجود می آورند. بنابراین بهترین انتخاب قراردادن آنها (مراکز) در فاصله هر چه بیشتر از یکدیگر می‌باشد.

قدم بعدی تخصیص هر الگو به نزدیک‌ترین مرکز می‌باشد. وقتی همه نقاط به مراکز موجود تخصیص داده شدند، مرحله اول تکمیل شده است و یک گروه‌بندی اولیه انجام شده است. در این مرحه نیاز داریم که k مرکز جدید برای کلاسترهای مرحله قبل محاسبه کنیم. بعد از تعیین k مرکز جدید، مجدداً داده‌ها را به مراکز مناسب تخصیص می‌دهیم. این مراحل را آنقدر تکرار می‌کنیم که دیگر k مرکز، جابجا نشوند. این الگوریتم تلاش می‌کند که یک تابع هدف[۱۲] را که تابع Squared error می‌باشد، مینیمم کند:

$$J = \sum_{j=1}^{k}\sum_{i=1}^{k}\left\| x_i^{(j)} - c_j \right\|^2 \qquad (۹)$$

در رابطه فوق $\left\| x_i^{(j)} - c_j \right\|^2$، یک معیار فاصله بین نقاط داده $x_i^{(j)}$ و مرکز کلاستر c_j می‌باشد و j مشخص‌کننده فاصله n نقطه داده از مراکز کلاستر مربوطه می‌باشد.

این الگوریتم از مراحل زیر تشکیل شده است:

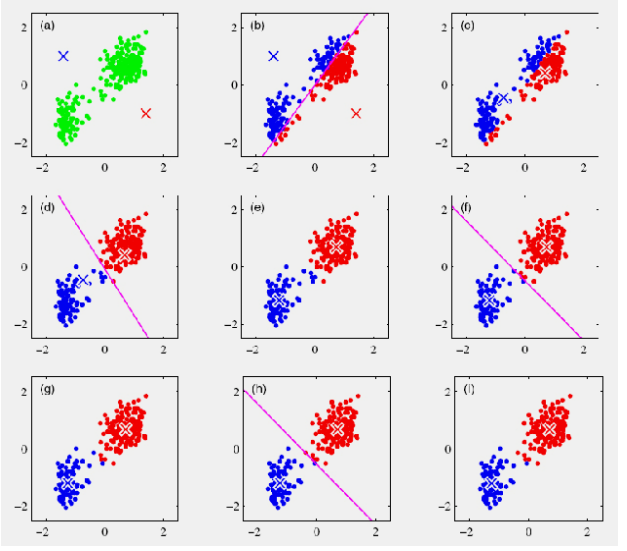
۱ـ مشخص‌کردن مراکز این نقاط که معرفی‌کننده مراکز گروه‌های اولیه می‌باشند.

۲ـ تخصیص هر الگو به گروهی که نزدیکترین مرکز به الگوی مربوطه را دارد.

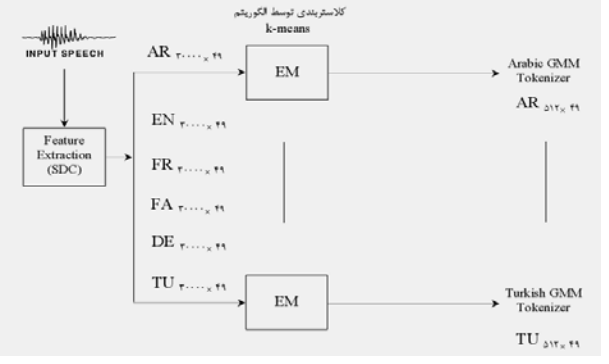
۳ـ وقتی که تمام الگوها تخصیص داده شدند، موقعیت k مرکز دوباره محاسبه می‌شود.

اگر چه ثابت شده است که الگوریتم همیشه پایان می‌پذیرد، الگوریتم k-mean، لزوماً جواب بهینه را پیدا نمی‌کند. این الگوریتم دارای حساسیت زیادی به مراکز کلاستر اولیه است که به‌صورت تصادفی انتخاب می‌شوند. برای کاهش این تأثیر می‌توان الگوریتم را چندین بار اجرا کرد. شکل ۷ عملکرد دسته‌بندی توسط الگوریتم k-mean را نشان می‌دهد.

فرض کنید به ازای هر زبان ۳۰ فایل برای آموزش Tokenizer در نظر بگیریم با فرض این‌که از هر فایل بتوان ۱۰۰۰ بردار آموزشی استخراج کرد، با یکجا جمع‌کردن این بردارها بردار ورودی با ابعاد ۴۹×۳۰۰۰۰ باشد این بردار توسط EM کلاستربندی شده و در خروجی یک GMM با مرتبه مدل ۵۱۲ خواهیم داشت که ۵۱۲ تعداد centerها می‌باشد. کلاستربندی یا خوشه‌بندی توسط الگوریتم k-means مقداردهی اولیه می‌شود. الگوریتم تولید Tokenizer به‌ازای هر ۶ تا زبان تکرار می‌شود. شکل ۸ نحوه ایجاد نشانه‌گذارها را نشان می‌دهد.



شکل ۷- نحوه دسته‌بندی توسط الگوریتم k-mean



شکل ۸- نحوه ایجاد نشانه‌گذارها

مرحله بعدی پس از تولید Tokenizer ساخت مدل‌های زبانی (Language models) می‌باشد. در این مرحله نیز ما برای آموزش از ۶۰ فایل صوتی ۳۰ ثانیه‌ای برای هر زبان استفاده کردیم که با فرض این‌که از هر فایل بتوان ۱۰۰۰ بر دار آموزشی استخراج کرد، با یکجا جمع‌کردن این بردارها بردار ورودی با ابعاد ۴۹×۶۰۰۰۰ خواهد بود که پس از گذشتن از Tokenizer تبدیل به ۶۰۰۰۰ عدد صحیح خواهد شد که متعلق به بازه ۱ تا ۵۱۲ می‌باشند. در این مرحله بررسی می‌گردد که هر بردار به کدام آمیزه نزدیک‌تر است؟ به‌ازای یک ورودی از هر زبان پس از گذشتن از Tokenizer شش

مدل زبانی خواهیم داشت به‌عنوان مثال منظور از LMAR-FA یعنی مدل زبانی که به‌ازای ورودی عربی و عبور از Tokenizer فارسی ایجاد شده است. ابعاد تمام مدل‌های زبانی ۵۱۲×۵۱۲ می‌باشد.

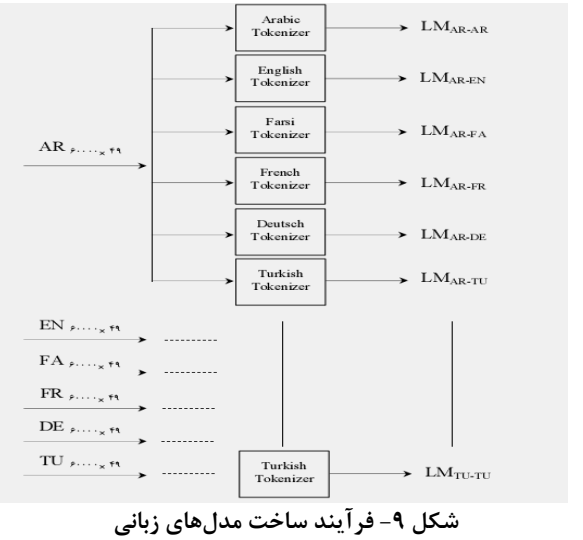
به‌ازای هر زبان شش Tokenizer داریم که منجر به تولید شش مدل زبانی خواهد شد. در نهایت ما به‌ازای ۶ زبان ورودی ۳۶ مدل زبانی خواهیم داشت. برای توصیف نحوه درست‌کردن مدل زبانی فرض می‌کنیم مجموعه اعداد خروجی Tokenizer یکی از زبان‌ها به‌ازای یک زبان ورودی به‌صورت[… 11 8 5 1] باشد (این اعداد نشان‌دهنده اندیس آمیزه گوسی است که به‌ازای یک بردار ورودی بیشترین امتیاز را آورده است). ابتدا باید احتمال وقوع uni-gramها و سپس احتمال وقوع Bi-gramها را محاسبه نمائیم.

منظور از uni-gramها یعنی محاسبه احتمال وقوع تک تک آمیزه‌های گوسی و منظور از Bi-gram یعنی احتمال وقوع دو آمیزه گوسی به‌صورت پشت سر هم. برای این کار از فرمول زیر (مدل interpolated bigram) که توسط تورس [۲و۳] درکارهای قبلی استفاده شده است ، به‌صورت زیر استفاده شد:

$$\hat{p}(a|b) = \lambda_2\, p(a|b) + \lambda_1 p(a) + \lambda_0 \qquad (۱۰)$$

(a و b نماینده اندیس آمیزه‌ها بوده و نشان‌دهنده احتمال وقوع آمیزه a به شرط این‌که قبل از آن آمیزه b آمده باشد را نشان می‌دهد).

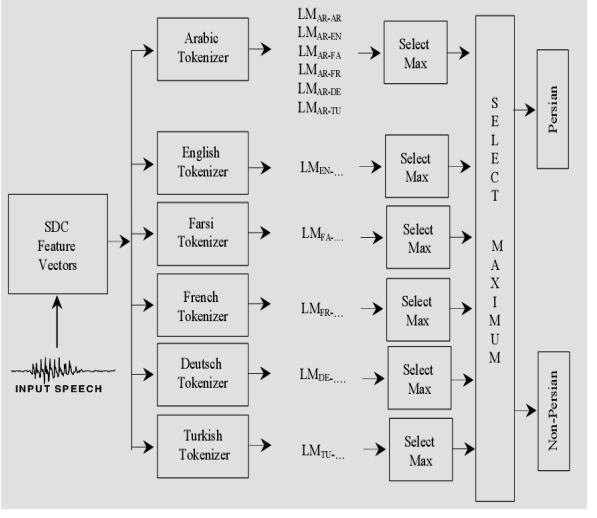
در شکل ۹ فرآیند ساخت مدل‌های زبانی نشان داده شده است.



شکل ۹- فرآیند ساخت مدل‌های زبانی

در مرحله تست هم به‌ازای یک ورودی پس از این‌که ورودی Tokenize شد، احتمال interpolated bigram محاسبه و روی کل ورودی میانگین گرفته می‌شود تا امتیاز تکه صحبت ناشناس بدست آید. امتیاز بدست‌آمده به هر کدام که نزدیک‌تر باشد سیستم آن را انتخاب می‌کند. در واقع شباهت هر نشانه‌گذار وابسته به دنباله سمبل‌ها به‌وسیله مدل زبانی ارزیابی می‌شود.

شکل ۱۰ یک سیستم نشانه‌گذار گوسی را در حالت کلی نشان می‌دهد. سیستم شامل مجموعه‌ای از نشانه‌گذارهای گوسی می‌باشد.



شکل ۱۰- ساختار سیستم گوسی نشانه‌گذار

۵- ترکیب مدل آمیزه‌های گوسی با نشانه‌گذارها و شبکه‌های عصبی

شبکه‌های عصبی مصنوعی که اغلب شبکه‌های عصبی نامیده می‌شوند یک مدل ریاضی یا یک مدل محاسباتی بر پایه شبکه‌های عصبی زیستی است [۸] که شامل مجموعه‌ی به‌هم پیوسته‌ای از نرون‌ها، می‌باشد. در بیشتر موارد شبکه عصبی سیستم تطبیقی است که ساختار خود را طبق اطلاعات داخلی یا خارجی که در روند آموزش از شبکه می‌گذرد تغییر می‌دهد. ما از شبکه‌های عصبی به عنوان پردازشگر پسین به منظور افزایش کارایی سیستم مدل آمیزه‌های گوسی همراه با نشانه‌گذار استفاده کردیم.

ایده بعدی استفاده از شبکه‌های عصبی به‌عنوان پردازشگر پسین از اینجا ناشی می‌شود که در سیستم پایه مدل آمیزه گوسی با نشانه‌گذار فقط بیشترین امتیاز نشانه‌گذار فارسی وابسته به مدل‌های زبانی معیار تصمیم می‌باشد در صورتی که ما می‌توانیم از نشانه‌گذارهای زبان‌های دیگر و امتیازات مدل‌های زبانی مربوطه استفاده کنیم تا امتیازهای زبان‌های غیرهدف را داشته باشیم تا شباهت بین زبان‌ها را داشته باشیم تا ماژولی (شبکه عصبی) را به منظور تفکیک زبان خاصی آموزش دهیم.

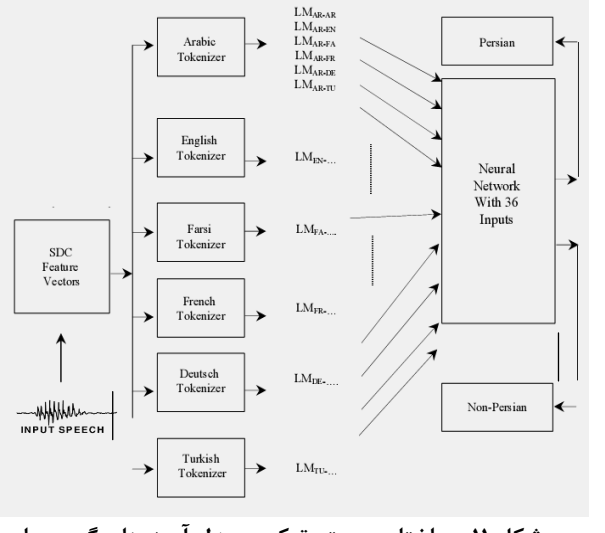
همان‌طور که در شکل ۱۱ مشاهده می‌شود پس از استخراج بردارهای ویژگی، بردارها از داخل نشانه‌گذارها عبور می‌کنند. دنباله‌ای از سمبل‌ها توسط هرمدل زبانی ارزیابی می‌شود تا درکل یک بردار ۳۶ بعدی بدست آید. برای تشخیص زبان فارسی به جای استفاده از امتیازات ۶ مدل زبانی (سیستم پایه گوسی با نشانه‌گذار) ما ۳۶ امتیاز را وارد یک شبکه عصبی می‌کنیم که قبلاً توسط اطلاعات ۶ زبان آموزش داده شده و برچسب‌گذاری شده‌اند.

برچسب زبان فارسی را یک و برچسب سایر زبان‌ها را صفر در نظر می‌گیریم. به دو علت انتظار می‌رود تا عملکرد بهتری داشته باشیم: دلیل اول استفاده از نشانه‌گذارها و امتیازهای همه مدل‌های زبانی و دوم اضافه‌کردن ماژول آموزش اضافی به سیستم (شبکه عصبی).

فرآیندی که در اینجا برای تشخیص زبان فارسی ذکر شد می‌تواند به هر یک از۶ زبان قید شده اعمال شود تا یک سیستم تشخیص‌دهنده کامل داشته باشیم. شکل ۱۱ الگوریتم پیشنهادی را برای تشخیص زبان فارسی نشان می‌دهد.

۶–آزمایش‌ها و نتایج

در این بخش نتایج آزمایش‌ها و مقایسه دو سیستم گوسی نشانه‌گذار با شبکه عصبی و گوسی نشانه‌گذار بدون شبکه عصبی نشان داده شده است. سیستم گوسی با نشانه‌گذار توسط تورس در روی دیتا بیس CallFriend و OGI ارزیابی شد[۲و۳].



شکل ۱۱- ساختار سیستم ترکیبی مدل آمیزه‌های گوسی با نشانه‌گذار و شبکه عصبی برای تشخیص زبان فارسی از سایر زبان‌ها

۶–۱– بانک اطلاعات صحبت و شرایط آموزش و تست

بانک اطلاعات صحبت استاندارد، می‌تواند بهترین بانک اطلاعاتی باشد که بتوان در این پروژه استفاده کرد تا بدین‌وسیله بتوان سیستم پیاده‌سازی شده در این پروژه را با سیستم‌های پیاده‌سازی شده دیگر در زمینه تشخیص زبان مقایسه کرد. از جمله بانک‌های اطلاعاتی استاندارد می‌توان OGI و NIST را نام برد. لازم به ذکر است که ما برای آموزش سیستم بانک اطلاعات قدیمی و ۲۲ زبانه OGI را تهیه نمودیم اما به علت پاره‌ای از مشکلات نتوانستیم از آن استفاده کنیم.

یکی از ایرادهای بانک اطلاعات OGI عدم تمایز تعدادی فایل‌های صوتی زبان‌های مختلف بود به‌طوری که مثلاً در داخل فایل‌های صوتی فارسی، چند فایل صوتی به زبان دیگر وجود داشت. یا وجود فایل‌های به طول زمانی یک ثانیه که سیستم قادر به آموزش آن نبود. به همین دلیل بانک اطلاعات صحبت استفاده شده در این پروژه، از چندین کانال تلویزیونی ماهواره‌ای متنوع تهیه گردید. هیچ نویز پس زمینه‌ای یا آهنگ و یا صحبت هم‌زمان در این پایگاه داده وجود ندارد.

طول زمانی همه صحبت‌های ذخیره شده۳۰ ثانیه بوده و اکثراً یک گوینده و یا در بعضی موارد دو گوینده در یک فایل صحبت کرده‌اند. فایل‌های مورد نظر با فرمت مونو و ۱۶ بیتی با فرکانس ۸ کیلو هرتز ذخیره شده‌اند. برای هر زبان ۲۰۰ فایل صوتی۳۰ ثانیه‌ای ذخیره گردید که از این تعداد،۴۰ فایل صوتی برای آموزش نشانه-گذار،۶۰ فایل صوتی برای آموزش مدل زبانی،۳۰ فایل صوتی برای آموزش شبکه عصبی و۷۰ فایل برای تست بکار برده شد.

نشانه‌گذارها با مرتبه مدل۵۱۲ و تنظیمات۷-۳-۱-۷ برای ایجاد بردارهای SDC انتخاب شدند. همچنین پارامترهای ایجاد مدل زبانی طبق کارهای قبلی انجام شده به‌صورت زیرانتخاب شدند[۳].

$$\lambda_0 = 0.01 \qquad \lambda_1 = 0.333 \qquad \lambda_2 = 0.666$$

شبکه عصبی استفاده شده در این کار pure line activation در لایه‌های مخفی و لایه‌های خروجی انتخاب گردید و با الگوریتم Levenberg-Marquardt backpropagation با خطای مجذور میانگین اندازه‌گیری شده عملکرد آموزش داده شد.

۶–۲– انواع خطاها

در حالت کلی سیستم‌های تشخیص دهنده دو کلاسی، با استفاده از دو نوع خطا بررسی می‌شوند. نوع اول نرخ خطاهای از دست دادن [۱۳] جواب صحیح و نوع دوم نرخ خطاهای آژیر اشتباه [۱۴] می‌باشد نرخ خطاهای از دست دادن عبارت است از احتمال این که زبان ادعا

شده در مرحله تست زبان فارسی باشد و سیستم آن را اشتباه تشخیص دهد که با Emiss نشان داده می‌شود.

نرخ خطاهای آژیر اشتباه عبارت است از حالتی که زبان ادعا شده در مرحله تست فارسی نباشد ولی سیستم، آن را به اشتباه فارسی تشخیص دهد، که با Efa نشان داده می‌شود که هر دو معیار فوق به‌صورت زیر محاسبه می‌شوند:

$$E_{miss} = \frac{n_{miss}}{n_{target}} \qquad (۱۱)$$

$$E_{fa} = \frac{n_{fa}}{n_{impostor}} \qquad (۱۲)$$

در روابط فوق n_{miss} تعداد دفعاتی است که در آن زبانی که ادعا شده فارسی می‌باشد توسط سیستم به اشتباه، فارسی تشخیص داده نمی‌شود. n_{target} تعداد دفعاتی است که در مرحله تست زبان فارسی به سیستم وارد می‌شود. n_{fa} تعداد دفعاتی است که درآن زبان، ادعا شده فارسی نمی‌باشد، اما توسط سیستم به اشتباه فارسی تشخیص داده می‌شود. nimposter تعداد دفعاتی می‌باشد که در مرحله تست زبان غیرفارسی به سیستم وارد می‌شود.

۶-۳- نرخ خطای برابر[۱۵]

هدف از استفاده و معرفی کردن نرخ خطای برابر این است که به وسیله نرخ خطای برابر می‌توان با مخلوط‌کردن نرخ خطای از دست دادن و آژیر اشتباه، یک نقطه‌کار واحد برای سیستم بدست آورد که به‌صورت EER نمایش داده می‌شود. نقطه‌کاری که نرخ خطای برابر به ما می‌دهد، نرخ خطای ازدست‌دادن و آژیر اشتباه با هم برابر می‌شوند. نرخ خطای برابر با تنظیم مقدار آستانه تصمیم‌گیری قابل استخراج است.

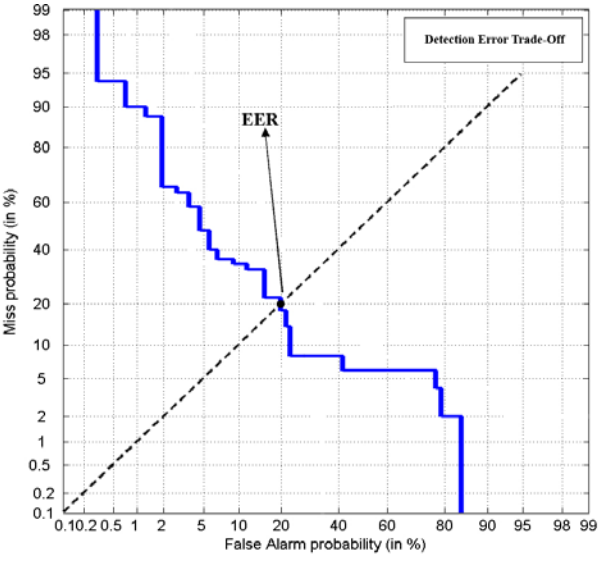
۶-۴- منحنی خطای DET[۱۶]

امروزه در مقالات گوناگون در مورد تشخیص زبان، معیار اصلی مقایسه بین نتایج مختلف را می‌توان منحنی خطای DET دانست که این نمودار اولین بار توسط مارتین در مقاله سال ۱۹۹۷ ارائه شد[۹]. نمودار DET را می‌توان مصالحه بین نرخ خطاهای ازدست‌دادن و آژیر اشتباه در نظرگرفت. نمودار DET دارای مقیاس احتمال غیرخطی می‌باشد و درحالتیکه توزیع احتمالات خطاها به‌صورت گوسی باشد منحنی‌های مصالحه حاصله به‌صورت خطوط راست خواهد بود.

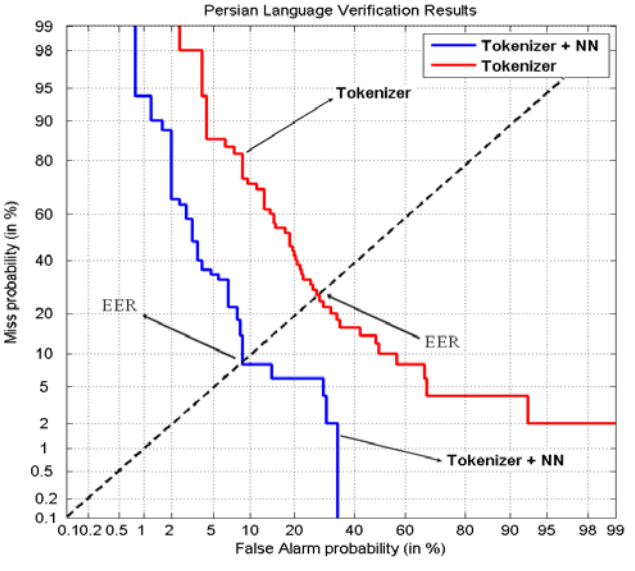
یکی از مزیت‌های استفاده از نمودار DET به این موضوع برمی‌گردد که فاصله بین منحنی‌ها، تفاوت عملکرد سیستم‌ها را نشان می‌دهد. در شکل ۱۲ نرخ خطای برابر و نمودار DET نشان داده شده است. منحنی خطای DET سیستم پایه و عملکرد سیستم پیشنهادی در بهترین حالت در شکل ۱۳ آمده است که عملکرد عالی سیستم پیشنهادی را نشان می‌دهد.

شکل ۱۴ مقایسه نرخ خطای برابر دو سیستم را نشان می‌دهد. همان‌طور که در شکل ۱۴ نشان داده شده است میزان خطای سیستم نشانه‌گذار فاقد شبکه عصبی ۲۶٫۴٪ است که با افزودن شبکه عصبی این خطا به ۸٫۴٪ کاهش می‌یابد.

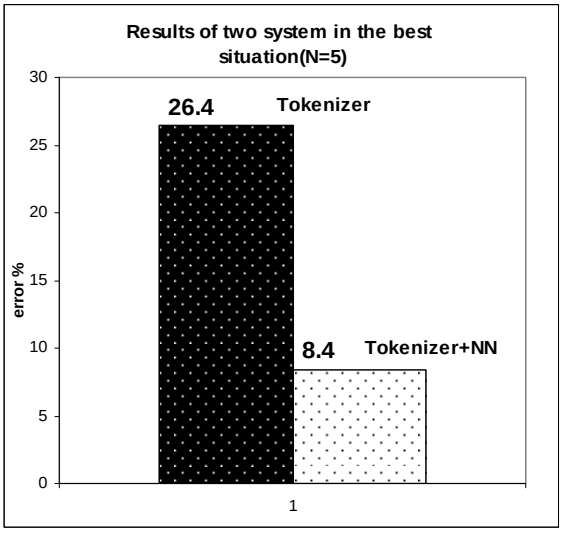
به تعداد سلول‌های مخفی در این کار مقادیر مختلفی اختصاص داده شد و بهترین نتیجه (۸/۴٪ خطا) در تعداد سلول ۵ رخ داد.



شکل ۱۲- منحنی خطای DET و نرخ خطای برابر

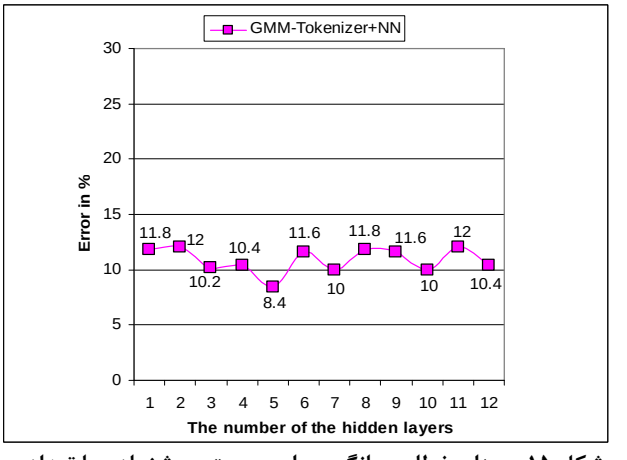


شکل ۱۳- مقایسه نرخ خطای برابر دو سیستم پایه و سیستم پیشنهادی در بهترین حالت (N=5) در منحنی خطای DET



شکل ۱۴- مقایسه خطای دو سیستم با ۵ سلول مخفی

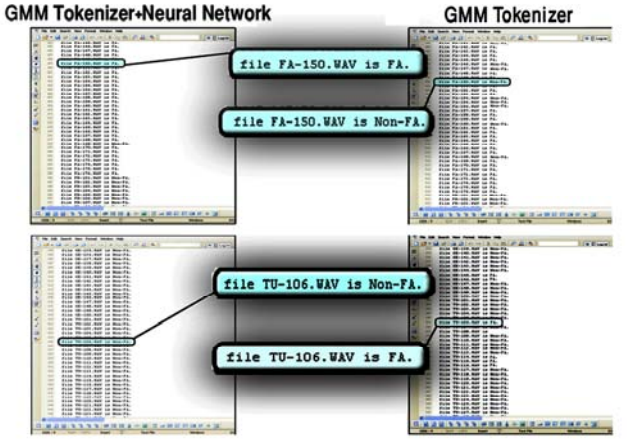
آزمایش‌ها نشان می‌دهند که با افزایش تعداد سلول‌های مخفی نتیجه بهتر نیست. میزان خطای میانگین برای سیستم پیشنهادی با تعداد سلول‌های مخفی مختلف در شکل ۱۵ نشان داده شده است.



شکل ۱۵- میزان خطای میانگین برای سیستم پیشنهادی با تعداد سلول‌های مخفی مختلف

۶-۵- مشاهده نتایج استفاده از شبکه عصبی در مرحله تست

شکل ۱۶ فرآیند تست تشخیص زبان فارسی را در محیط نرم افزار MATLAB نشان می‌دهد. سمت راست تصویر، تست فایل FA-150 را نشان می‌دهد که سیستم پایه آن را به اشتباه غیر فارسی تشخیص داده است، در صورتی که این فایل در سمت چپ تصویر توسط سیستم پیشنهادی با شبکه عصبی، فارسی تشخیص داده شده است. یا در قسمت پایین تصویر تست فایل TU-106 که یک فایل ترکی است، توسط سیستم پایه به اشتباه فارسی تشخیص داده شده است در صورتی که سیستم پیشنهادی آن را غیر فارسی تشخیص داده است.



شکل ۱۶- مشاهده نتایج استفاده از شبکه عصبی به عنوان پردازشگر پسین در مرحله تست

۷- نتیجه‌گیری

در این مقاله ما سیستم ترکیبی مدل آمیزه‌های گوسی نشانه-گذار و شبکه‌های عصبی را روی پایگاه داده جدید جمع‌آوری شده با کیفیت عالی برای تشخیص زبان فارسی مورد آزمایش قرار دادیم و نتایج عملکرد عالی الگوریتم را نشان می‌دهند. آنچه که مشخص است شبیه‌سازی تشخیص زبان فارسی از سایر زبان‌ها توسط نرم افزار MATLAB کاری تقریبا زمان‌بر می‌باشد و با صرف زمان بیشتر می‌توان این سیستم پیشنهادی را به یک سیستم تشخیص‌گر در مورد همه زبان‌ها تعمیم داد. همچنین برای کاهش خطای تشخیص می‌توان تعداد فایل‌های صوتی آموزش سیستم را افزایش داد یا پارامترهای ساخت مدل زبانی را بهینه کرد. در ضمن می‌توان با افزایش تعداد فایل‌های صوتی هر زبان می‌توان به نتایجی بهتر در آموزش سیستم پایه دست یافت که مربوط به تحقیقات آینده می‌باشد. البته افزایش فایل‌های صوتی آموزش تا یک مرحله خاصی جوابگو خواهد بود زیرا با افزایش فایل‌های آموزشی تعداد پارامترها از توان یادگیری سیستم بیشتر شده و سیستم فقط نمونه‌های آموزشی را خوب توصیف می‌کند و در مرحله تست قادر به تشخیص بهتر فایل‌های ناشناس نمی‌باشد.

نتیجه مهم دیگر این است که بردارهای ویژگی SDC باعث پیشی‌گرفتن GMM از بقیه مدل‌ها شده است. علت این امر آن است که GMM خود به تنهایی نمی‌تواند تغییرات تدریجی بین قاب‌ها را مدل‌کند و با توجه به این‌که SDC تغییرات تدریجی قاب‌ها

را در بردارد، لذا ترکیب GMM با SDC بهتر از روش‌های قبلی جواب می‌دهد.

لازم به ذکر است که عمدتاً، بیشترین خطای ایجادشده در تشخیص زبان فارسی، زبان عربی و در بعضی از موارد زبان ترکی می‌باشد. علت این امر آن است که نشانه‌گذارها یک مدل n-gram برای توالی نشانه‌ها را پیاده‌سازی می‌نمایند و در واقع توالی n-gram وقتی برابرند که کلمات یکسانی در دو زبان داشته باشیم که معمولاً بین زبان‌های مشابه از جمله عربی ـ فارسی و در بعضی از موارد در فارسی ـ ترکی رخ می‌دهد.

۸– سپاسگزاری

تمام نویسندگان این مقاله از آقای رحیم سعیدی که در زمان انجام این کار و پیشبرد آن ما را همکاری نمودند تشکر می‌نمایند.

۹– مراجع

[1] M. A. Zissman; "**Comparison of four approaches to automatic language identification of telephone speech**", IEEE Transactions on Speech and Audio Processing, vol. 4, 1996.

[2] P. A. Torres-Carrasquillo, D. A. Reynolds, and J. R.Deller Jr.; "**Language identification using Gaussian Mixture Model Tokenization**", In ICASSP, Orlando, Fl., USA, 2002.

[3] P. A. Torres-Carrasquillo, D. A. Reynolds, and J. R.Deller Jr.; "**Approaches to Language Identification using Gaussian Mixture Models and Shifted Delta Cepstral Features**", In ICASSP, Orlando, Fl., USA, 2002.

[4] Torres-Carrasquillo P. A. Gleason-T. P. Campbell W. M.and Reynolds D.A. Singer, E.; **Acoustic,phonetic, and discriminative approaches to automatic language recognition**. Proc. Euro speech in Geneva, Switzerland, ISCA, pages 1345_1348, 1-4 September 2003.

[5] P.A. Torres-Carrasquillo-D.A. Reynolds W.M. Campbell, E. Singe: **Language recognition with support vector machines**. The speaker and Language Recognition Workshop, Toledo Spain,ISCA, pages 41_44, 31 May - 3 June 2004.

[6] R. Tong, B. Ma, D. Zhu, H. Li, and E. S. Chng; "**Integrating acoustic, prosodic and phonotactic features for spoken language identification**", in Proc. ICASSP, 2006, pp. 205–208.

[7] H Li, B Ma; "**A phonotactic language model for spoken language identification**", in Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics,pp. 515-522, 2005.

[8] http://en.wikipedia.org/wiki/Artificial_neural_network

[9] A. Martin, G. Doddington, T. Kamm, M. Ordowski, M. Przybocki: **The DET Curve in Assessment of Detection Task Performance,** Proceedings of Eurospeech**,** (1997) pp. 1895–1898

[10] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn; "**Speaker verification using adapted Gaussian mixture models**", Digital Signal Processing, vol. 10, pp. 19–41, 2000.

[11] Xu L., Jordan M.I., "**On Convergence Properties of the EM Algorithm for Gaussian Mixtures**", Neural Computation, vol. 8, pp. 129-151, 1996.

[12] McKenzie P., Alder M.; "**Initializing the EM algorithm for use in Gaussian mixture modeling**", Technical Report: TR93-14, The University of Western Australia, Center of Intelligence Information processing Systems, Crawley, Australia.(Available from: http://ciips.ee.uwa.edu.au/papers.)

[13] Vuuren S.; "**Speaker Verification in a Time-Feature space**", Ph.D. thesis, Oregon Graduate Institute, March 1999.

۱۰– پی‌نوشت‌ها

[1]- Gaussian Mixture Models
[2]- Tokenizer
[3]- Backend processor
[4]- Support vector machines
[5]– Phone Recognition Language Model
[6]– Parallel Phone Recognition Language Model
[7]– Linear Predictive Cepstral Coefficients
[8]– Mel Frequency Cepstral Coefficients
[9]- Perceptual Linear Predictive Cepstral Coefficients
[10]– Shifted Delta Cepstral
[11]- Estimation Maximum
[12]- Objective Function
[13]- Miss Error Rate
[14]– False Alarm Error Rate
[15]- Equal Error Rate
[16]– Detection Error Trade-off