محمد جواد رنجبر

۸۱۰۱۰۱۱۷۳

پروژه امتیازی درس پردازش گفتار:

تشخیص احساسات زبان فارسی با استفاده از یادگیری عمیق

دکتر ویسی

بهار ۱۴۰۳

فهرست

فهرست شکل‌ها

فهرست جدول‌ها

چکیده:

در سال‌های اخیر با پیشرفت روزافزون مدل‌های یادگیری ماشین و به خصوص مدل‌های یادگیری عمیق تمرکز زیادی به زمینه‌ی پردازش صوت، صورت گرفته است. تشخیص احساسات از گفتار از یکی مهم‌ترین ابزار انتقال و درک معنی در گفتار است با این وجود مدل‌های موجود برای این کار همچنان جا برای بهتر کردن دارد. مدل‌های از پیش آموزش دیده با روش self-supervised به طور مداوم نتایج پیشرفته‌ای را در زمینه پردازش زبان طبیعی (NLP) ارائه داده اند. با این حال، شایستگی آنها در زمینه تشخیص احساسات گفتار (SER) هنوز به بررسی بیشتر نیاز دارد. در این پژوهش قصد پیاده‌سازی مدل تشخیص احساسات برای زبان فارسی با استفاده از روش‌های یادگیری عمیق داریم که با استفاده از مدل‌های بزرگ برای وظیفه‌ی تشخیص احساسات آن‌ها را تنظیم دقیق می‌کنیم.

مقدمه:

در سال‌های اخیر، پردازش گفتار به عنوان یکی از زمینه‌های کلیدی در هوش مصنوعی و یادگیری ماشین به طور فزاینده‌ای مورد توجه قرار گرفته است. پردازش گفتار نه تنها امکان تبدیل گفتار به متن را فراهم می‌کند بلکه ابزارهای پیشرفته‌تری نظیر تشخیص احساسات از گفتار نیز ارائه می‌دهد. تشخیص احساسات از گفتار یکی از مهم‌ترین ابزارهای انتقال و درک معنی در ارتباطات انسانی است که می‌تواند در بهبود تعاملات انسانی-کامپیوتری و بهبود کارایی سیستم‌های خودکار نقش بسزایی ایفا کند.

اهمیت تشخیص احساسات از گفتار را می‌توان در کابردهای متعددی مشاهده کرد. این تکنولوژی می‌تواند در حوزه‌هایی نظیر خدمات مشتری، تحلیل احساسات در رسانه‌های اجتماعی، سیستم‌های توصیه‌گر، و حتی در حوزه پزشکی و روان‌شناسی برای ارزیابی حالت‌های روحی و روانی افراد مورد استفاده قرار گیرد. با وجود پیشرفت‌های چشم‌گیر در این زمینه، مدل‌های موجود همچنان نیازمند بهبود و ارتقاء هستند تا بتوانند با دقت و کارایی بیشتری احساسات را از گفتار تشخیص دهند.

علاوه بر این، به دلیل مشکلات موجود در جمع‌آوری داده‌ها، مجموعه داده‌های عمومی اغلب تعداد کافی گوینده ندارند تا بتوانند تنوعات شخصی در بیان احساسات را به‌طور مناسب پوشش دهند. به همین دلیل، برخی از تکنیک‌های رایج یادگیری عمیق که به SERوارد شده‌اند، شامل یادگیری انتقالی، یادگیری چندوظیفه‌ای، سیستم‌های چندوجهی، و معماری‌های مدل قدرتمندتر می‌باشند.

مجموعه داده:

در این پروژه قصد داریم با استفاده از مدل‌های آموزش دیده روی دیتاست‌های بزرگ برای وظیفه‌ی تشخیص احساسات روی مجموعه‌داده‌ی ShEMO آموزش دهیم، این مجموعه‌داده شامل ۳۰۰۰ صوت نیمه طبیعی، معادل ۳ ساعت و ۲۵ دقیقه داده‌های گفتاری است که از نمایشنامه های رادیویی آنلاین استخراج شده است. ShEMO نمونه‌های گفتاری ۸۷ فارسی زبان بومی را برای پنج احساس اساسی شامل خشم، ترس، شادی، غم و تعجب و همچنین حالت خنثی پوشش می‌دهد.

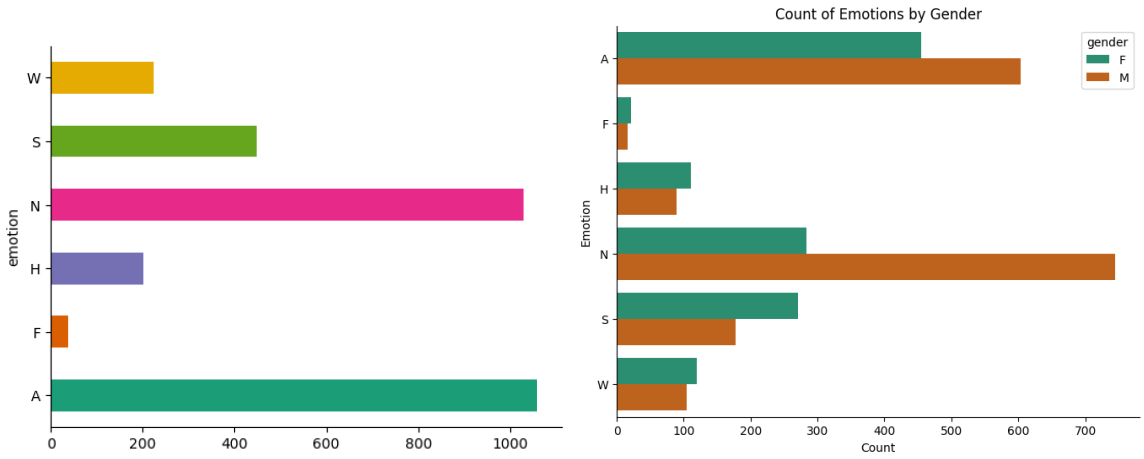
در این مجموعه داده توزیع جنسیت افراد به صورت زیر می‌باشد:

جنسیت	تعداد
مرد	۱۷۳۷

زن	۱۲۳۶

جدول1 توزیع جنسیت

با توجه به جدول بالا تعداد صوت برای مردها بیشتر از زنان است که احتمالا باعث ایجاد مقداری اشتباه در مدل بشود با این حال این موضوع اهمیت کمتری نسبت به توزیع نابرابر در احساسات مختلف دارد که به صورت زیر می‌باشد:



شکل1 توزیع داده‌ها در کلاس‌های مختلف

احساس	خشم (A)	ترس (F)	خوشحال (H)	غم (S)	تعجب (W)	خنثی (N)
تعداد	۱۰۵۹	۳۸	۲۰۱	۴۴۹	۲۲۵	۱۰۲۸

جدول2 توزیع احساسات

با توجه به توزیع کلاس‌های مختلف، مشخص است که مدل نهایی در یادگیری کلاس‌هایی که تعداد کمتری نمونه دارند مشکل خواهد داشت، برای رفع این مشکل چند راه‌حل پیشنهاد می‌شود:

۱. **توازن کلاس‌ها با استفاده از تکنیک‌های بازنمونه‌گیری:**
   - **Oversampling**: افزایش تعداد نمونه‌ها در کلاس‌های با تعداد کمتر از طریق تکرار یا تولید نمونه.
   - **Undersampling**: کاهش تعداد نمونه‌ها در کلاس‌های با تعداد بیشتر به منظور تعادل با کلاس‌های کمتر.
۲. **استفاده از الگوریتم‌های خاص برای داده‌های نامتعادل:**
   - استفاده از مدل‌هایی که به طور خاص برای مقابله با توزیع نامتعادل داده‌ها طراحی شده‌اند، مانند الگوریتم‌های جنگلی تصادفی (Random Forest) با وزن‌دهی کلاس‌ها یا الگوریتم‌های مبتنی بر هزینه.
۳. **تنظیم وزن کلاس‌ها در مدل:**
   - در تنظیمات مدل، به کلاس‌هایی که تعداد کمتری نمونه دارند وزن بیشتری داده شود تا مدل بیشتر بر روی آنها تمرکز کند.
۴. **استفاده از معیارهای ارزیابی مناسب:**

- به جای معیارهای ارزیابی معمول مانند دقت(Accuracy) ، از معیارهایی مانندF1-Score ، AUC-ROC، Precisionو Recall استفاده شود که بهتر قادر به ارزیابی عملکرد مدل در داده‌های نامتعادل هستند.

۵. **استفاده از داده‌های افزایشی:**

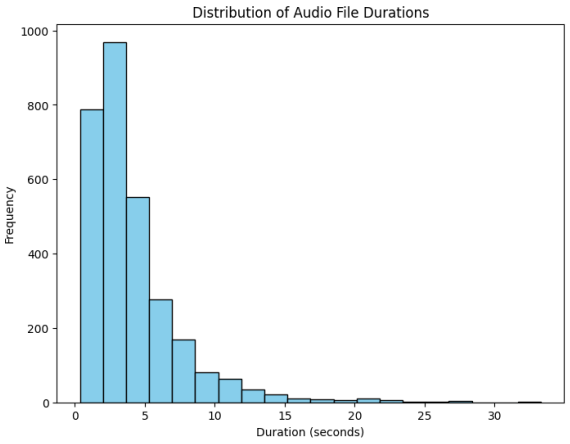
- جمع‌آوری داده‌های بیشتر برای کلاس‌هایی که تعداد نمونه‌های کمتری دارند تا به تعادل بهتری برسند.

۶. **تولید داده‌های مصنوعی:**

- استفاده از تکنیک‌های تولید داده‌های مصنوعی مانند GAN (شبکه‌های مولد متخاصم) برای ایجاد نمونه‌های جدید از کلاس‌های کم تعداد.

با استفاده از این راه‌حل‌ها، می‌توان به بهبود عملکرد مدل در مواجهه با توزیع نامتعادل داده‌ها کمک کرد و دقت پیش‌بینی‌ها را در کلاس‌های با تعداد کمتر نمونه بهبود بخشید. روش oversampling و under sampling برای این پروژه به کار گرفته شده است که در نوت بوک‌های مختلف قابل مشاهده است.

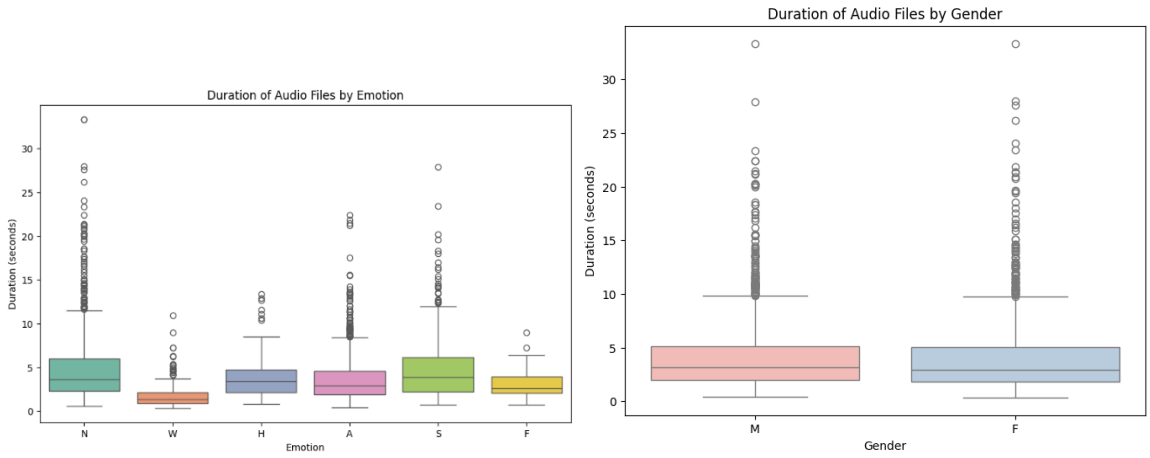
علاوه بر این توزیع مدت زمان صوت به صورت زیر می‌باشد:



شکل ۲ هیستوگرام مدت زمان صوت‌ها

بیشتر صوت‌ها دارای طول حدود ۵ ثانیه می‌باشد، با توجه به این موضوع می‌توان همه صوت‌ها را به طول کمتر از ۵ ثانیه دراورد که

تغییر خاصی در نتیجه نهایی نخواهد داشت.



شکل ۳ باکس پلات طول صوت‌ها، کلاس و جنسیت

همچنین توزیع طول کلاس‌های مختلف با توجه به کلاس‌ها به صورت بالا می‌باشد که اطلاعات مفید زیادی در اختیار ما نمی‌گذارد.

مدل‌های استفاده شده:

HuBERT:

مدل HuBERT (Hidden-Unit BERT) یک مدل یادگیری نمایش گفتار خودنظارتی است که توسط Facebook AI توسعه داده شده است. این مدل برای یادگیری نمایش‌های صوتی گفتار بدون نیاز به داده‌های برچسب‌گذاری شده طراحی شده است، که آن را به‌ویژه در سناریوهایی که داده‌های حاشیه‌نویسی شده کمیاب یا در دسترس نیست، مفید می‌سازد.

مفاهیم کلیدی و معماری:

۱. **یادگیری خودنظارتی:**
   - HuBERT با استفاده از رویکرد یادگیری خودنظارتی آموزش دیده است، به این معنا که یاد می‌گیرد تا بخش‌هایی از داده‌های ورودی را خودش پیش‌بینی کند. به‌طور خاص، بخش‌های ماسک شده صوتی را از بخش‌های بدون ماسک پیش‌بینی می‌کند، مشابه روشی که BERT کلمات ماسک شده در یک متن را پیش‌بینی می‌کند.
۲. **استراتژی ماسک‌گذاری:**
   - در طول آموزش، بخش‌های تصادفی از ورودی صوتی ماسک می‌شوند و مدل برای پیش‌بینی این بخش‌های ماسک شده بر اساس بخش‌های بدون ماسک آموزش می‌بینند. این امر مدل را مجبور به یادگیری نمایش‌های معنادار از صدا می‌کند.
۳. **واحدهای پنهان:**

- به جای پیش‌بینی مستقیم سیگنال صوتی خام، HuBERT فریم‌های صوتی را با استفاده از الگوریتم خوشه‌بندی k-means به واحدهای گسسته (واحدهای پنهان) خوشه‌بندی می‌کند. این واحدهای پنهان به عنوان هدف برای وظیفه پیش‌بینی استفاده می‌شوند.

٤. **معماری مدل:**
   - HuBERT از معماری مبتنی بر ترنسفورمر استفاده می‌کند، که برای ثبت وابستگی‌های بلندمدت در داده‌های متوالی مانند گفتار مناسب است. مدل ترنسفورمر ویژگی‌های صوتی را پردازش کرده و نمایش‌هایی ارائه می‌دهد که برای وظایف پایین‌دستی مختلف مفید هستند.

فرآیند آموزش:

١. **استخراج ویژگی:**
   - صوت خام ابتدا با استفاده از یک استخراج‌کننده ویژگی به دنباله‌ای از بردارهای ویژگی تبدیل می‌شود، مانند ضرایب کپسترال فرکانس مل (MFCC) یا بانک‌های فیلتر.

٢. **خوشه‌بندی:**
   - این بردارهای ویژگی سپس با استفاده از خوشه‌بندی k-means به واحدهای پنهان خوشه‌بندی می‌شوند. خوشه‌ها واحدهای فونتیک گسسته در گفتار را نشان می‌دهند.

٣. **پیش‌بینی ماسک شده:**
   - در طول آموزش، برخی از بردارهای ویژگی ماسک می‌شوند و مدل برای پیش‌بینی واحدهای پنهان این بخش‌های ماسک شده آموزش می‌بیند.

کاربردها:

- **تشخیص گفتار** HuBERT :می‌تواند برای وظایف تشخیص خودکار گفتار (ASR) تنظیم شود، که در آن بهبودهای قابل توجهی نسبت به مدل‌های قبلی نشان داده است.
- **شناسایی گوینده** :نمایش‌های یادگرفته شده توسط HuBERT می‌توانند برای شناسایی گویندگان در یک کلیپ صوتی استفاده شوند.
- **تولید گفتار** :همچنین می‌تواند در سیستم‌های تبدیل متن به گفتار (TTS) به‌کار رود تا گفتاری طبیعی‌تر تولید کند.

مزایا:

- **کارآیی داده** HuBERT :از مقادیر زیادی از داده‌های صوتی بدون برچسب بهره می‌برد و آن را در سناریوهایی که داده‌های برچسب‌گذاری شده محدود است، بسیار کارآمد می‌سازد.
- **بهبود عملکرد** :نشان داده شده است که در وظایف مختلف پردازش گفتار از مدل‌های قبلی پیشی می‌گیرد.
- **همه‌کاره بودن** :نمایش‌های یادگرفته شده عمومی هستند و می‌توانند برای طیف وسیعی از کاربردهای مرتبط با گفتار تطبیق داده شوند.

Wav۲vec:

wav2vec نیز یک مدل یادگیری خودنظارتی برای تعبیه صوت است که توسط Facebook AI توسعه داده شده است. این مدل به‌طور خاص برای پردازش و درک گفتار از داده‌های صوتی خام طراحی شده است و مانندHuBERT ، به برچسب‌گذاری گسترده داده نیاز ندارد.

مفاهیم کلیدی و معماری:

۱. **یادگیری خودنظارتی:**
   - wav2vec با استفاده از یادگیری خودنظارتی آموزش می‌بیند که به مدل اجازه می‌دهد تا نمایش‌های معنایی از داده‌های صوتی را بدون برچسب‌گذاری دستی فراگیرد. این مدل با پیش‌بینی بخش‌های ماسک شده از سیگنال صوتی، نمایش‌های غنی از اطلاعات را یاد می‌گیرد.
۲. **استراتژی ماسک‌گذاری:**
   - بخش‌های تصادفی از سیگنال صوتی ورودی ماسک می‌شوند و مدل برای پیش‌بینی این بخش‌های ماسک شده بر اساس بخش‌های بدون ماسک آموزش می‌بیند. این فرآیند مدل را مجبور به یادگیری وابستگی‌های طولانی مدت در داده‌های صوتی می‌کند.
۳. **مدل‌سازی چندمرحله‌ای:**
   - wav2vec در دو مرحله اصلی آموزش می‌بیند: مرحله پیش‌آموزش و مرحله تنظیم نهایی. در مرحله پیش‌آموزش، مدل با استفاده از یادگیری خودنظارتی نمایش‌های عمومی از داده‌های صوتی یاد می‌گیرد. در مرحله تنظیم نهایی، مدل برای وظایف خاصی مانند تشخیص گفتار آموزش دیده و بهینه می‌شود.
٤. **معماری مدل:**
   - wav2vec از یک شبکه عصبی کانولوشنی (CNN) برای استخراج ویژگی‌های سطح پایین از سیگنال صوتی خام و یک ترنسفورمر برای مدل‌سازی وابستگی‌های طولانی مدت استفاده می‌کند. این ترکیب به مدل اجازه می‌دهد تا نمایش‌های معنایی قدرتمندی از داده‌های صوتی ایجاد کند.

فرآیند آموزش:

۱. **استخراج ویژگی:**
   - صوت خام ابتدا با استفاده از شبکه عصبی کانولوشنی به بردارهای ویژگی تبدیل می‌شود. این بردارهای ویژگی نماینده ویژگی‌های سطح پایین صوت هستند.
۲. **پیش‌آموزش خودنظارتی:**
   - بخش‌هایی از بردارهای ویژگی ماسک می‌شوند و مدل برای پیش‌بینی این بخش‌های ماسک شده بر اساس بردارهای بدون ماسک آموزش می‌بیند. این فرآیند مدل را قادر می‌سازد تا نمایش‌های معنایی عمومی از داده‌های صوتی یاد بگیرد.
۳. **تنظیم نهایی:**

- پس از پیش‌آموزش، مدل برای وظایف خاصی مانند تشخیص خودکار گفتار (ASR) تنظیم می‌شود. این مرحله شامل آموزش مدل با داده‌های برچسب‌گذاری شده برای بهینه‌سازی عملکرد آن در وظایف خاص است.

کاربردها:

- **تشخیص گفتار** wav2vec :به‌طور گسترده‌ای برای وظایف ASR استفاده می‌شود و بهبودهای قابل توجهی در دقت و کارآیی نشان داده است.
- **پردازش گفتار** :این مدل می‌تواند برای وظایفی مانند شناسایی گوینده، تشخیص احساسات، و تبدیل گفتار به متن استفاده شود.
- **فهم گفتار** :نمایش‌های یادگرفته شده توسط wav2vec می‌توانند برای تحلیل و درک معنایی داده‌های صوتی به‌کار روند.

مزایا:

- **کاهش نیاز به داده‌های برچسب‌گذاری شده** :با استفاده از یادگیری خودنظارتی، wav2vecمی‌تواند از مقادیر زیادی از داده‌های صوتی بدون برچسب بهره ببرد و نیاز به برچسب‌گذاری دستی را کاهش دهد.
- **بهبود عملکرد** :این مدل در بسیاری از وظایف پردازش گفتار، به‌ویژه تشخیص گفتار، عملکرد قابل توجهی نشان داده است.
- **انعطاف‌پذیری و همه‌کاره بودن** wav2vec :می‌تواند به راحتی برای وظایف مختلف پردازش گفتار تنظیم شود و نمایش‌های معنایی عمومی ایجاد کند که در طیف گسترده‌ای از کاربردها مفید هستند.

روش‌های کلاسیک:

در روش‌های کلاسیک تشخیص احساسات از صوت، به جای استفاده از ترنسفورمرها و شبکه‌های عصبی پیچیده، از ویژگی‌های استخراج شده مستقیم از سیگنال‌های صوتی و الگوریتم‌های یادگیری ماشین سنتی استفاده می‌شد. برخی از ویژگی‌های رایج که در این روش‌ها استفاده می‌شد عبارتند از:

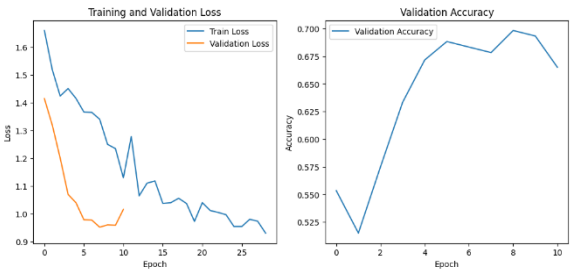
۱. **ویژگی‌های زمانی** :شامل ویژگی‌هایی مانند مدت زمان سکوت، طول گفتار، نرخ تغییرات سیگنال و الگوهای زمان‌بندی در گفتار.
۲. **ویژگی‌های فرکانسی** :شامل ویژگی‌هایی مانند توزیع انرژی در باندهای فرکانسی مختلف، طیف قدرت، و پارامترهای ملو-فرکانسی کپسترال.(MFCC)
۳. **ویژگی‌های آماری** :شامل میانگین، واریانس، چولگی و کشیدگی توزیع‌های مختلف ویژگی‌های زمانی و فرکانسی.

نتایج:

مدل Hubert base

این مدل که برای وظایف خاص خود آموزش دیده است، می‌تواند به خوبی برای وظایف مختلفی مانند تشخیص گفتار و تحلیل صداها به کار رود. در اینجا برای وظیفه تشخیص احساسات زبان فارسی آن را تنظیم دقیق می‌کنیم. نتایج آن به صورت زیر می‌باشد:

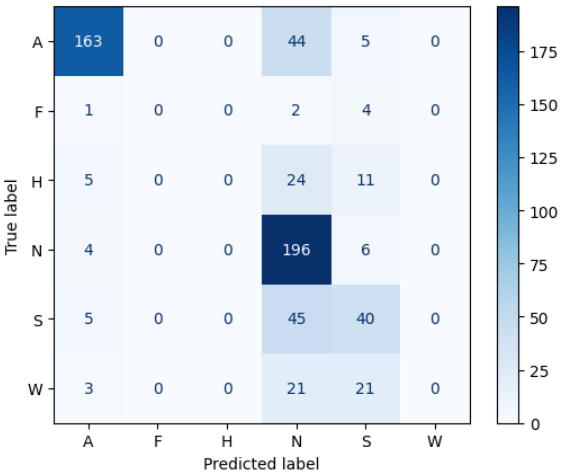
این مدل را برای ۱۰ دوره آموزش می‌دهیم و نمودار خطا و دقت در طول آموزش به شکل زیر می‌باشد:



شکل ۴عملکرد Hubert در حین آموزش

همچنین عملکرد مدل روی داده‌های تست به شکل زیر می‌باشد:

Classification Report:	precision	recall	f1-score	support
A	0.90	0.77	0.83	212
F	0.00	0.00	0.00	7
H	0.00	0.00	0.00	40
N	0.59	0.95	0.73	206
S	0.46	0.44	0.45	90
W	0.00	0.00	0.00	45
accuracy			0.67	600
macro avg	0.33	0.36	0.34	600
weighted avg	0.59	0.67	0.61	600



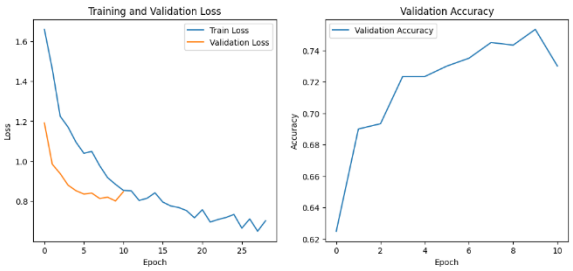
شکل ۵ عملکرد Hubert بر روی داده‌های تست

مشخص است که مدل به کلاس‌هایی که داده‌های کمتری داشتن underfit شده است و با وجود دقت ۶۰٪ درصد همچنان تعدادی از کلاس‌ها را تشخیص نداده است که با توجه به توزیع نابرابر داده‌ها این اتفاق پیش‌بینی می‌شد.

مدل Wav۲vec er

این مدل روی مجموعه داده‌ی تشخیص احساسات انگلیسی آموزش دیده است و به احتمال زیاد برای زبان فارسی نیز عملکرد مناسبی خواهد داشت. تنها کافی است تا روی مجموعه داده‌های فارسی نیز آموزش داده شود تا دقت آن افزایش یابد. در اینجا برای وظیفه تشخیص احساسات زبان فارسی آن را تنظیم دقیق می‌کنیم. نتایج آن به صورت زیر می‌باشد:

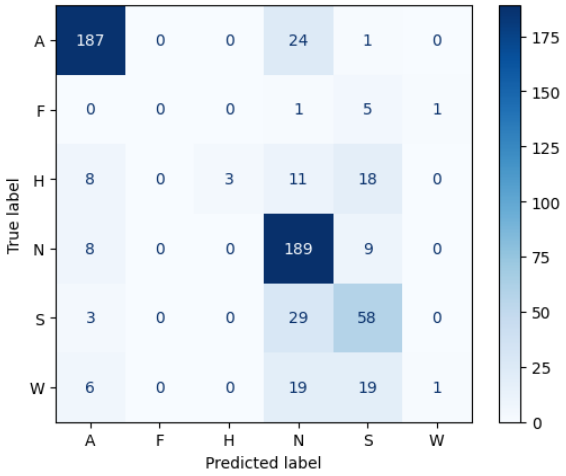
این مدل را برای ۱۰ دوره آموزش می‌دهیم و نمودار خطا و دقت در طول آموزش به شکل زیر می‌باشد:



شکل ۶ عملکرد مدل Wav2vec

همچنین عملکرد مدل روی داده‌های تست به صورت زیر می‌باشد:

Classification Report:	precision	recall	f1-score	support
A	0.88	0.88	0.88	212
F	0.00	0.00	0.00	7
H	1.00	0.07	0.14	40
N	0.69	0.92	0.79	206
S	0.53	0.64	0.58	90
W	0.50	0.02	0.04	45
accuracy			0.73	600
macro avg	0.60	0.42	0.41	600
weighted avg	0.73	0.73	0.68	600



شکل ۷ عملکرد مدل Wav2vec روی داده‌های تست

در اینجا نیز با وجود عملکرد کلی بهت مدل روی داده‌ها همچنان، مدل روی کلاس‌هایی که دادههای کمتری دارند مشکل دارد و دقت پایینی در آن کلاس‌ها نشان می‌دهد.

روش‌های کلاسیک:

ابتدا ویژگی‌های زیر را برای صوت‌ها استخراج می‌کنیم:

MFCC: **MFCC**ها نمایشی از طیف توان کوتاه مدت یک صدا هستند که معمولاً در پردازش گفتار و صدا استفاده می‌شوند.

آن‌ها با گرفتن تبدیل فوریه از یک سیگنال، نگاشت توان‌های طیف به مقیاس مل و سپس گرفتن تبدیل کسینوس گسسته از لگاریتم

این توان‌ها مشتق می‌شوند.

کروما: ویژگی‌های کرومای به دوازده کلاس مختلف زیر و بمی مربوط می‌شوند و اغلب برای شناسایی هارمونی و آکوردها در موسیقی استفاده می‌شوند. آن‌ها نمایانگر انرژی هر یک از ۱۲ کلاس زیر و بمی (دو، دو دیز، ر، ...، سی) در یک فریم مشخص هستند.

مل اسپکتروگرام: مل اسپکتروگرام یک اسپکتروگرام است که در آن فرکانس‌ها به مقیاس مل تبدیل می‌شوند که بیشتر با ادراک شنوایی انسانی همخوانی دارد. این ویژگی نمایشی زمان-فرکانس از سیگنال صوتی ارائه می‌دهد.

حال مدل‌های قدیمی یادگیری ماشین به همراه یک شبکه‌ی عصبی ساده را برای وظیفه تشخیص احساسات آموزش می‌دهیم:

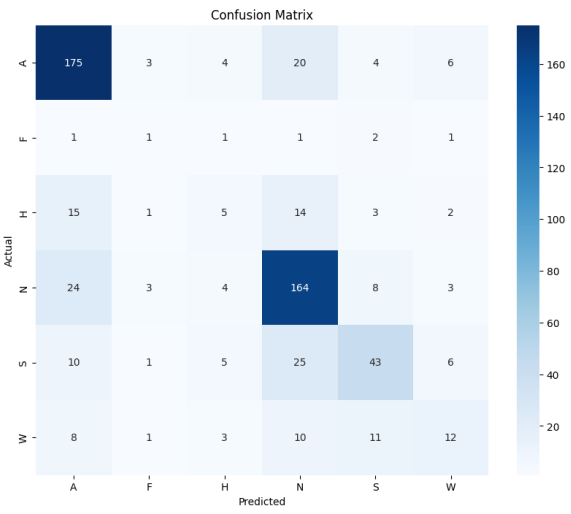
SVM:

نتایج SVM خطی به صورت زیر می‌باشد:

```
Test Accuracy: 0.6666666666666666
Classification Report:
              precision    recall  f1-score   support

           A       0.75      0.83      0.79       212
           F       0.10      0.14      0.12         7
           H       0.23      0.12      0.16        40
           N       0.70      0.80      0.75       206
           S       0.61      0.48      0.53        90
           W       0.40      0.27      0.32        45

    accuracy                           0.67       600
   macro avg       0.46      0.44      0.44       600
weighted avg       0.64      0.67      0.65       600
```

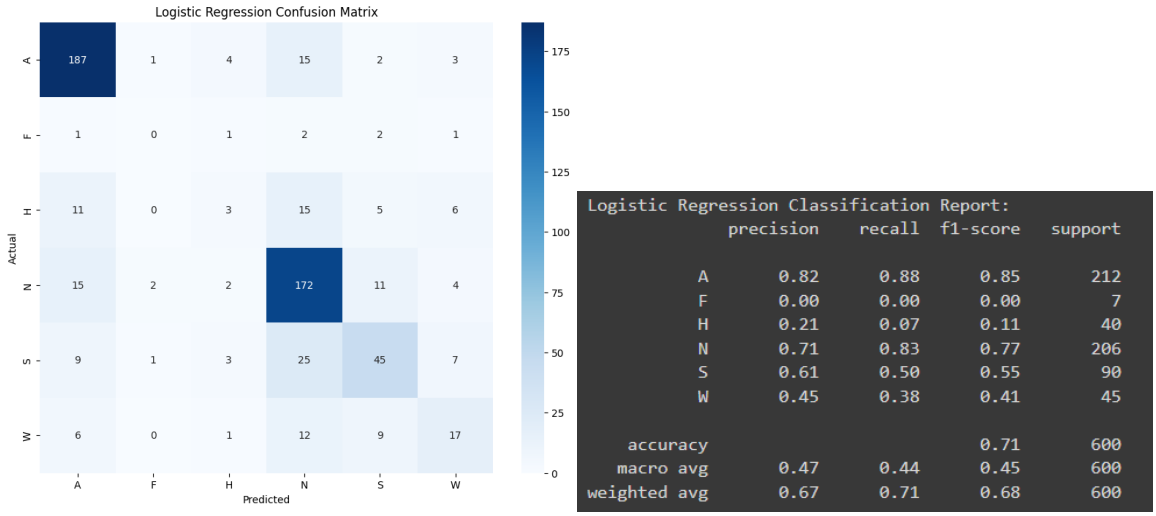


شکل ۸ عملکرد SVM روی داده‌ها تست

مشخص است که این مدل نسبت به مدل شبکه‌ی عصبی عملکرد کلی بدتری دارد، با این حال این مدل با وجود سادگی، تعمیم‌پذیری بیشتری داشته و در کلاس‌هایی که داده‌ها خیلی کم بوده‌اند نیز تا حدی بهتر از شبکه‌ی عصبی عمل کرده است.

Logistic regression:

نتایج آن به صورت زیر می‌باشد:



Logistic Regression Classification Report:

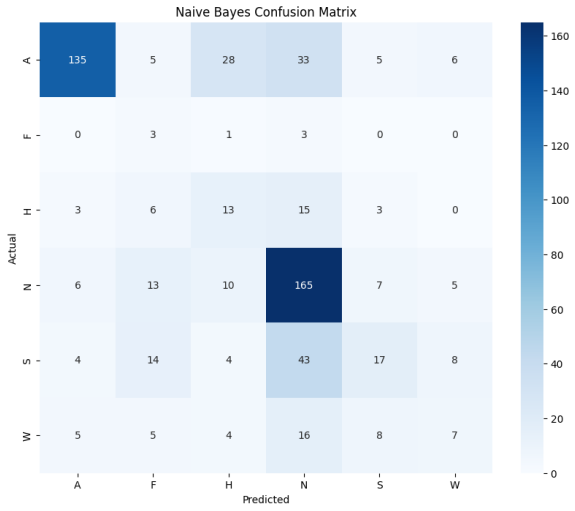
	precision	recall	f1-score	support
A	0.82	0.88	0.85	212
F	0.00	0.00	0.00	7
H	0.21	0.07	0.11	40
N	0.71	0.83	0.77	206
S	0.61	0.50	0.55	90
W	0.45	0.38	0.41	45
accuracy			0.71	600
macro avg	0.47	0.44	0.45	600
weighted avg	0.67	0.71	0.68	600

شکل ۹ عملکرد Logistic regression روی داده‌های تست

مشخص است که این مدل نسبت به مدل شبکه‌ی عصبی عملکرد کلی بدتری دارد، با این حال این مدل با وجود سادگی، تعمیم‌پذیری بیشتری داشته و در کلاس‌هایی که داده‌ها خیلی کم بوده‌اند نیز تا حدی بهتر از شبکه‌ی عصبی عمل کرده است.

Naïve Bayes:

عملکرد آن روی داده‌های تست به صورت زیر می‌باشد:



Naive Bayes Classification Report:

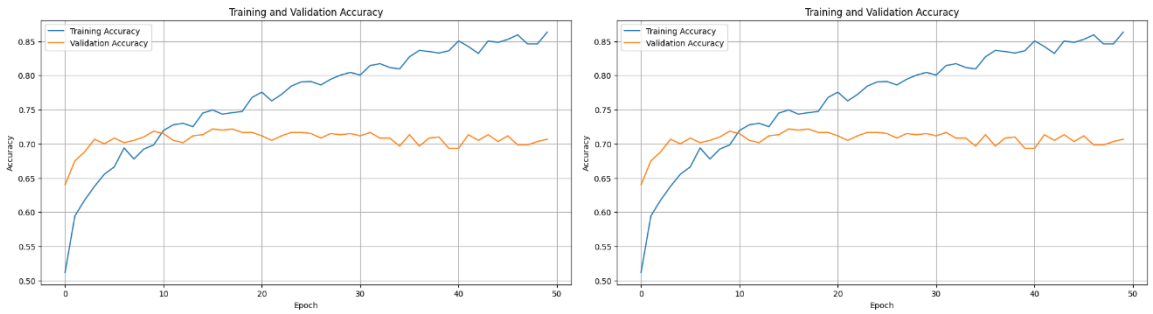
	precision	recall	f1-score	support
A	0.88	0.64	0.74	212
F	0.07	0.43	0.11	7
H	0.22	0.33	0.26	40
N	0.60	0.80	0.69	206
S	0.42	0.19	0.26	90
W	0.27	0.16	0.20	45
accuracy			0.57	600
macro avg	0.41	0.42	0.38	600
weighted avg	0.62	0.57	0.57	600

شکل ۱۰ عملکرد Naive Bayes روی داده‌های تست

عملکرد آن نسبت به مدل‌های قبلی بدتراست با این حال همچنان تعمیم‌پذیری بهتری نسبت به شبکه‌ی عصبی دارد.

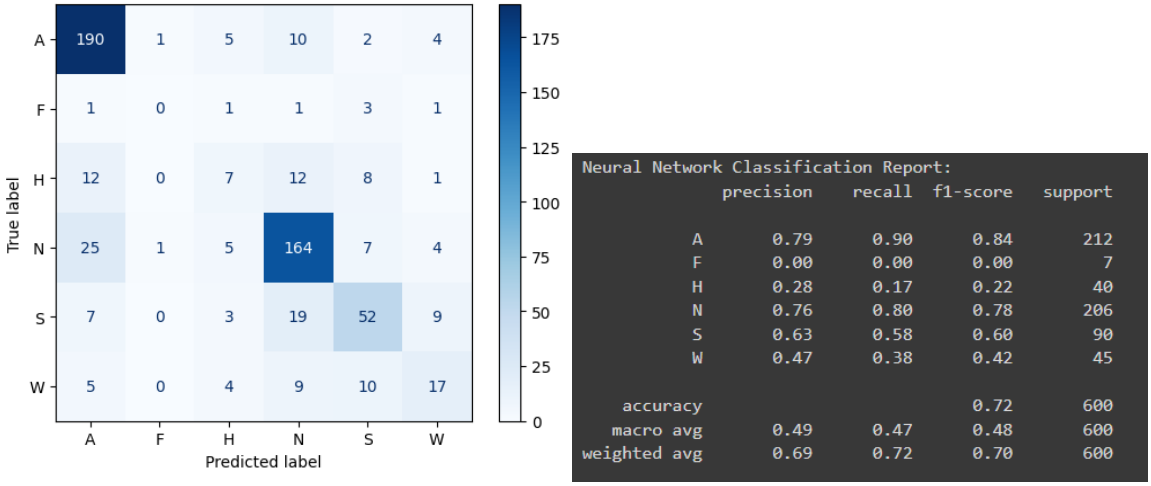
مدل شبکه‌ی عصبی ساده:

این مدل را برای ۵۰ دوره آموزش می‌دهیم:



شکل ۱۱عملکرد MLP در حین آموزش

عملکرد این مدل روی داده‌های تست به صورت زیر می‌باشد:



شکل ۱۲ عملکرد MLP روی داده‌های تست

این مدل نیز عملکرد بهتری نسبت به مدل‌های ساده داشت و همچنین تعمیم‌پذیری کمتری روی یک سری از کلاس‌ها و داشت ولی همچنان از مدل‌های پیچیده‌ی تراسنفورمری بهتر عمل می‌کند.

نتایج با oversampling

با oversample کردن داده‌ها عملکرد مدل‌های یادگیری عمیق تا مقداری روی داده‌های کلاس‌های اقلیت بهتر می‌شود، با این حال با oversample کردن به صورتی داریم مدل‌ها را به توزیع داده‌های خودمان overfit می‌کنیم و احتمالا تعمیم‌پذیری این مدل روی داده‌های خارج از این توزیع بدتر خواهد بود.

همچنین مدل‌های کلاسیک تقریبا عملکرد مشابهی با حالت معمولی خواهند داشت(البته در این حالت مقداری کاهش عملکرد را نیز توقع داریم چون داریم به کلاس‌هایی که تعداد کمتری دارند احتمال بیشتری می‌دهیم.)

نتایج با under sampling

با توجه به اینکه این مجموعه داده به صورت کلی کوچک است، مشخصا با under sample کردن داده‌ها شبکه‌ی عصبی که همیشه به تعداد زیادی داده نیاز دارد، در همه‌ی کلاس‌ها به شدت بدتر عمل خواهد کرد. همچنین عملکرد مدل‌های کلاسیک یادگیری ماشین نیز با under sample کردن داده‌ها بدتر خواهد شد و دقت و تعمیم‌پذیری خود را از دست می‌دهند.

نتایج با اضافه کردن مجموعه‌داده‌های انگلیسی

مشخصا عملکرد مدل در تمام حالت‌ها بهتر می‌شود، با این حال تعدادی از کلاس‌های اقلیت در سایر مجموعه داده‌ها نیز کلاس اقلیت بوده‌اند و لذا تشخیص مدل در این کلاس‌ها همچنان دارای مشکل است.

نتایج با حذف کردن کلاس‌های اقلیت

با توجه به کمبود داده در تعدادی از کلاس‌ها منطقی به نظر می‌رسد که این کلاس‌ها را حذف کنیم تا جز odd حساب شوند، با حذف این کلاس‌ها مدل به کلاس‌هایی که داده به اندازه کافی دارند، بهتر فیت شده و دقت در هریک از این کلاس‌ها افزایش می‌یابد، با این حال ما تعمیم‌پذیری مدل روی کلاس‌های اقلیت را از دست داده‌ایم.

نتیجه‌گیری

مدل‌های یادگیری عمیق به روش‌های متنوع آموزش دید که تمام نوت‌بوک‌های مربوط به آن موجود می‌باشد. با توجه به قابلیت‌های مدل‌های از پیش‌آموزش دیده، استفاده‌ی آن‌ها در این وظایف عملکرد خوبی را نشان داد و از مدل‌های کلاسیک بهتر بود با این حال، با توجه به حجم دیتاست فارسی، مشخص است که باید مجموعه‌داده‌ی بزرگتر و با تنوع بیشتری برای این کار جمع‌آوری شود. علاوه بر این هیرویستک‌هایی مانند تجمیع یک مدل تشخیص احساسات از متن و مدل صوت هم می‌تواند به دقت مدل کمک کند. به

صورت کلی مدل Wav2vec که قبلا روی دیتاست انگلیسی آموزش دیده است با تنظیم دقیق روی داده‌های فارسی در بین همه‌ی مدل‌ها بهترین نتیجه را داشت.