

بسم الله الرحمن الرحيم



دانشگاه صنعتی شریف  
دانشکده مهندسی کامپیووتر  
فاز دوم پروژه درس سامانه‌های یادگیری ماشین

عنوان:  
قسمت طراحی و آموزش مدل  
SentiMovie پروژه

مدرس:  
دکتر علی زارع‌زاده

اعضای گروه:  
محمدحسین موثقی‌نیا (۰۰۰۰۰۹۱۹)  
مهدی منوچهری (۰۰۰۲۱۱۵۹۲)  
حمید مهتدی جعفری (۰۰۱۲۱۲۳۴۶)

# ۱ مقدمه

در این فاز از پروژه ابتدا یک فرآیند توسعه به کمک wandb طراحی شده است و سپس مدل‌های یادگیری ماشین مختلفی با کانفیگ‌های مختلف داده بر روی دادگان بررسی شده است. در نهایت برای آن‌ها یک مجموعه هایپرپارامتر بهینه به دست آمده و عملکرد مدل نهایی با جزئیات مورد بررسی قرار گرفته است؛ که شامل بررسی معیارهای عملکرد مختلف می‌باشد و همچنین بررسی Calibration Curve و تحلیل خطا نیز انجام شده است.

## ۲ مانیتورنیگ و مدیریت توسعه با استفاده از wandb

به منظور مدیریت فرآیند مانیتورینگ اجرای آزمایش‌ها و روند توسعه پروژه یک ساختار منسجم و یکسان در نوت‌بوک wandb-flow قرار داده شده است که شامل تعریف و استفاده از wandb [۱] برای مانیتورینگ پروژه می‌باشد. همچنین نام متغیرها نیز در این نوت‌بوک مشخص شده است تا تمامی بخش‌ها در ادامه از همین نام گذاری و روش لاغ گرفتن استفاده کنند تا یکپارچگی لاغ‌ها حفظ شود. این نوت‌بوک به عنوان یک تمپلیت در توسعه مدل‌های مختلف مورد استفاده قرار گرفته است.

## ۳ بهینه‌سازی هایپرپارامترها با استفاده از wandb

به منظور بهینه‌سازی هایپرپارامترهای مدل کاندید، یک فرآیند ساختارمند با استفاده از wandb توسعه داده شده است تا به صورت منظم ترکیبی از هایپرپارامترها را تست کند. برای این منظور یک تابع run-model نوشته شده که در این تابع ابتدا دادگان فراخوانی شده و سپس دیتاست<sup>۱</sup> و دیتالودر<sup>۲</sup> برای آن‌ها ساخته می‌شود و پس از آن مدل، بهینه‌ساز، زمانبند مدیریت نرخ آموزش<sup>۳</sup>، وزن‌دهی کلاس‌ها ساخته می‌شوند و در نهایت همه این موارد به تابع train ارسال شده تا فرآیند آموزش انجام شود. در نهایت نیز بهترین مدل انتخاب شده و به صورت جداگانه بر روی دادگان تست مورد بررسی قرار می‌گیرد. کدهای مربوط به این بخش در نوت‌بوک wandb-flow قابل دسترس است.

<sup>1</sup>Dataset

<sup>2</sup>Dataloader

<sup>3</sup>Learning-rate scheduler

## ۴ بررسی مدل‌ها

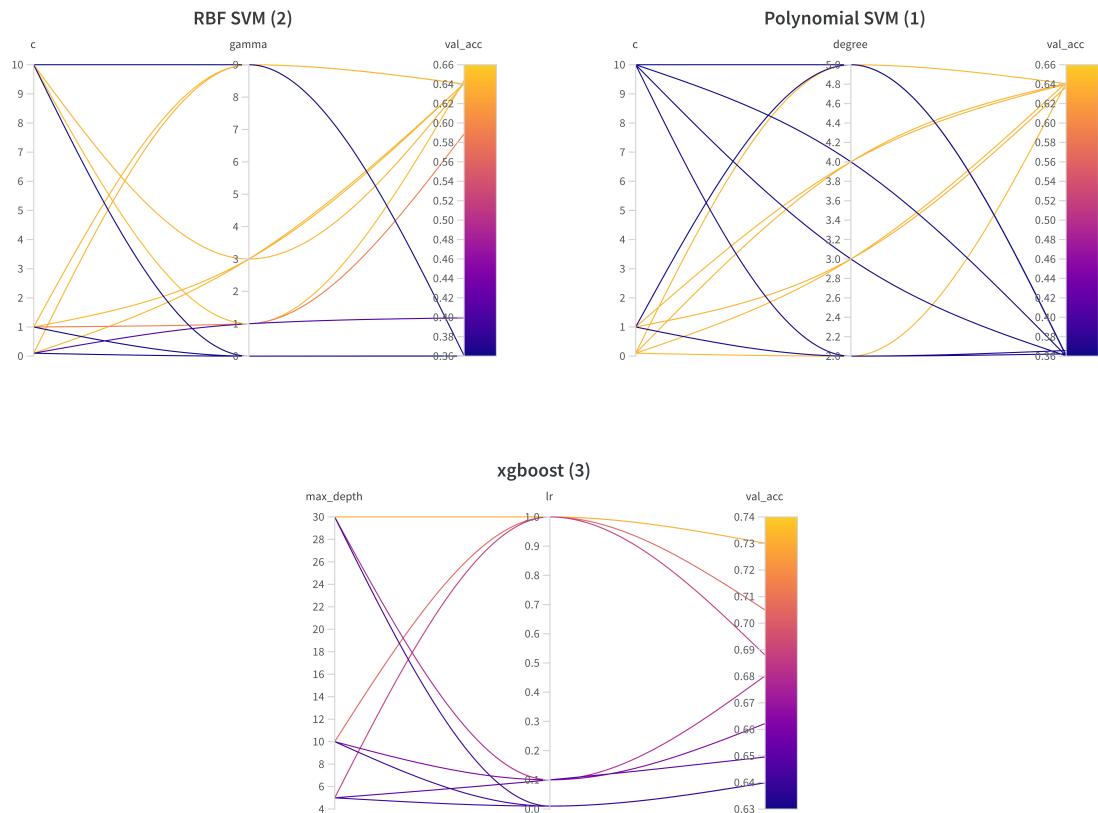
به منظور بررسی عملکرد مدل‌های مختلف انتخاب مدل بهینه، مدل‌های یادگیری ماشین متفاوتی بر روی دادگان مورد بررسی قرار گرفت. مدل کاملاً رندم و مدل رندم براساس احتمال هر کلاس، مدل SVM خطی، چندجمله‌ای و برپایه رادیان، مدل Logistic Regression مورد بررسی به عنوان مدل‌های پایه قرار گرفته است. به مدل‌ها به جز دو روش رندم، بردار tf-idf دادگان به عنوان ورودی داده شده است. به عنوان مدل‌های کاندید سه مدل برپایه ترنسفورمر BERT، ROBERTA و distil-BERT از پیش آموزش دیده روی متون، مورد استفاده قرار گرفته است. که در نهایت مدل distil-BERT با توجه به این که سبک‌تر بوده و دقت مناسبی نیز به دست آورده است، به عنوان مدل نهایی انتخاب شده است. برای این مدل‌ها به عنوان ورودی هم دادگان نرمال شده استفاده شده است و هم دادگان بدون نرمال‌سازی. نتایج با جزئیات در جدول ۱ آمده است.

جدول ۱: بررسی معیارهای عملکرد مختلف در مدل‌های ساده یادگیری ماشین و مدل‌های پیچیده به منظور انتخاب مدل بهینه. در مدل‌های پایه مدل Logistic regression عملکرد بهتری داشته است و در میان مدل‌های مبتنی بر ترنسفورمر، مدل distil-BERT بر روی دادگان غیر نرمال شده عملکرد بهتری داشته است. در مدل BERT دوم که بر روی دادگان غیر نرمال بررسی شده است با توجه به محدودیت زمانی امکان ذخیره‌سازی و بررسی بقیه معیارهای عملکرد نشده است.

Model	Data	Accuracy	F1	Precision	Recall
Random	main-normal	0.49	0.50	0.53	0.49
Random with prob	main-normal	0.54	0.54	0.54	0.54
Linear SVM	tf-idf	0.48	0.47	0.58	0.48
Poly SVM	tf-idf	0.64	0.52	0.63	0.64
RBF SVM	tf-idf	0.64	0.51	0.64	0.64
Logistic Regression	tf-idf	<b>0.81</b>	<b>0.80</b>	<b>0.80</b>	<b>0.81</b>
Random Forest	tf-idf	0.74	0.73	0.74	0.74
xgboost	tf-idf	0.73	0.71	0.72	0.73
BERT	main-normal	0.66	0.63	0.74	0.72
BERT	not-normal	0.78	-	-	-
Roberta	main-normal	0.75	0.73	0.82	0.78
distil-BERT	not-normal	<b>0.89</b>	<b>0.91</b>	<b>0.91</b>	<b>0.92</b>

## ۱-۴ مدل‌های پایه

به منظور بررسی بهتر مدل‌های پایه یک جست و جوی کامل روی فضایی از هایپرپارامترهای آنها صورت گرفته است که جزئیات کد مربوطه در فایل model/Baseline.ipynb قرار دارد. همچنین تمامی فرآیند با استفاده از wandb لاگ گرفته شده است که در [این لینک](#) قابل مشاهده است. همچنین برخی از نمودارهای مربوط به جست و جوی فضای هایپرپارامتر در شکل ۱ قابل مشاهده است.

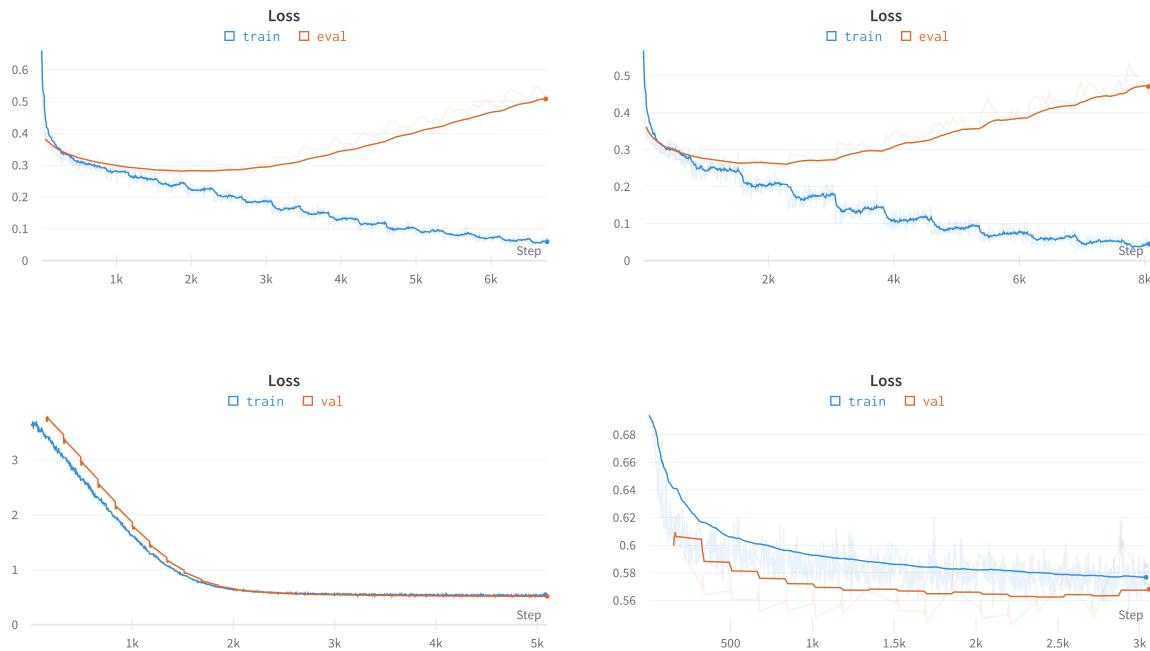


شکل ۱: سه نمونه از جست و جوی فضای هایپرپارامترها (جزئیات بیشتر در [این لینک](#) )

## ۲-۴ مدل‌های مبتنی بر ترنسفورمر

با توجه به این که در مقالات مختلف نشان داده شده است که مدل‌های مبتنی بر ترنسفورمر از جمله Roberta و BERT عملکرد بهتری نسبت به مدل‌های ساده یادگیری ماشین دارند، با این فرض سه مدل مختلف را بررسی کردیم. مدل BERT از پیش آموزش دیده شده، مدل Roberta از پیش آموزش دیده شده و مدل distil-BERT از پیش آموزش دیده شده. برای همه این موارد نیز از هاب Huggingface استفاده شده است. همانطور که در جدول ۱ اشاره شد، مدل BERT و Roberta عملکرد جالبی از خود نشان ندادن که احتمالی که ما می‌دهیم این است که با استی لایه‌های بیشتری از آن‌ها را از حالت فریز خارج می‌کردیم و احتمالاً در این صورت نتایج بهتری به دست می‌آمد. دلیل دیگری که این مسئله را نشان می‌دهد این است که مدل‌ها underfit می‌شوند و به دلیل محدودیت منابع و زمان به دنبال مدل نسبتاً ساده‌تری رفتیم که هم سبک‌تر باشد و هم بتوانیم آزمایش‌های بیشتری بر روی آن انجام دهیم و تلاش کنیم تا اول از همه روی دادگان overfit شویم تا مطمئن باشیم که مدل به اندازه کافی امکان یادگیری دارد و سپس تلاش کنیم تا با بررسی هایپرپارامترها و موارد دیگر با استفاده از این مدل به generalization مناسب برسیم. به همین

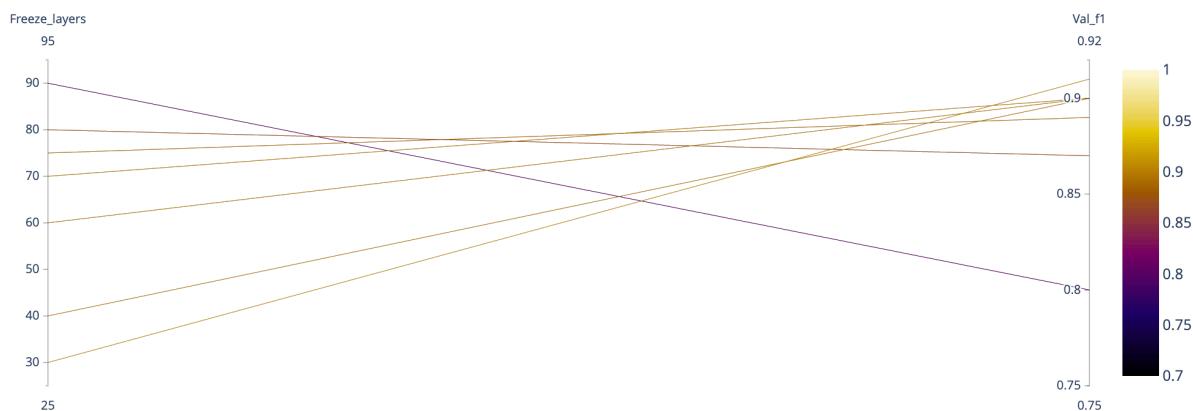
منظور از مدل distil-BERT استفاده کردیم و نهایتاً توانستیم با فریز کردن ۳۰ لایه اول آن بهترین نتایج را به دست آوریم. (با توجه به محدودیت منابع و زمان امکان اجرای جست و جوی فضای حالات هایپرپارامترها در این مدل‌ها نبود و به صورت دستی جست و جویی نسبی صورت گرفته است). نکته قابل ذکر این است که نتایج زمانی بهتر شد که هیچ کدام از نرم‌السازی‌های روی دادگان انجام نشده است. البته نیاز به زمان بیشتر و آزمایشات بیشتری هست تا بتوانیم دقیق‌تر بررسی کنیم که چه ترکیبی از هایپرپارامترها بهترین عملکرد را خواهند داشت. اما در حال حاضر بهتری ترکیب، استفاده از دادگان بدون نرم‌السازی و مدل distil-BERT بوده است. (تمامی فرآیندهای آموزش و بررسی عملکرد همه مدل‌ها با استفاده از wandb مانیتور شده است). به منظور ذکر کدن برخی از فرآیندهای آموزش مدل در این گزارش، چهار مورد از این فرآیندها در شکل ۲ قابل مشاهده است.



شکل ۲: نمودار تابع هزینه مدل‌های مختلف ترنسفورمری. (راست بالا) نمودار تابع هزینه برای مدل distil-bert با ۸۰ لایه فریز شده می‌باشد که همانطور که مشاهده می‌شود هم مقدار هزینه روی دادگان آموزش بسیار کمتر از دو نمودار پایین شده است و هم توانسته ایم به حالت overfit بررسیم که این به معنی توانایی مدل در یادگیری ویژگی‌ها است. (چپ بالا) این نمودار نیز مرتبط با مدل distil-bert با ۳۰ لایه فریز شده می‌باشد که مثل نمودار قبل توانایی مدل در یادگیری را نشان می‌دهد و همچنین این مدل دقیق‌تر بیشتری را در دادگان اعتبارسنجی (0.89) به دست آورده است. (راست پایین) این نمودار مربوط به مدل BERT با دو لایه FC در آنها برای آموزش می‌باشد که نتوانسته است بر روی دادگان overfit شود و حداقل دقیقی که بدست آورده است 0.78 بوده و همانطور که در نمودار نیز مشخص است مقدار هزینه از 0.5 کمتر نشده است در حالی که در نمودارهای بالا به زیر 0.1 نیز رسیده ایم. (چپ پایین) نمودار تابع هزینه مرتبط با مدل Roberta می‌باشد که همانند مدل BERT این مدل هم از چند لایه متصل در آنها برای آموزش استفاده شده است و باز هم به مقدار قابل توجهی تفاوت دقیق با مدل بهینه داشته است.

### ۳-۴ بهترین مدل

با توجه به این که بهترین نتایج را مدل distil-BERT داشته است؛ با جزئیات بیشتری نتایج این مدل را بررسی کردیم. به دلیل محدودیت منابع نتوانستیم یک جست و جوی منسجم در فضای هایپرپارامترها انجام دهیم ولی به صورت کلی یکی از مهمترین پارامترها تعداد لایه‌های فریز شده بود که در نمودار ۳ قابل مشاهده است. بنابراین مدلی که در ادامه مورد استفاده قرار گرفته است، مدل distil-BERT با ۳۰ لایه فریز شده در هنگام آموزش می‌باشد. همچنین به منظور بررسی میزان اطمینان مدل به ازای صحت پیش‌بینی‌های مدل، از نمودار کالیبریشن استفاده کردیم که بر روی دادگان اعتبارسنجی بررسی شده است (شکل ۴). همانطور که مشخص است مدل توانسته است به میزان مناسبی متناسب با انتظار یک مدل بهینه عمل کند.

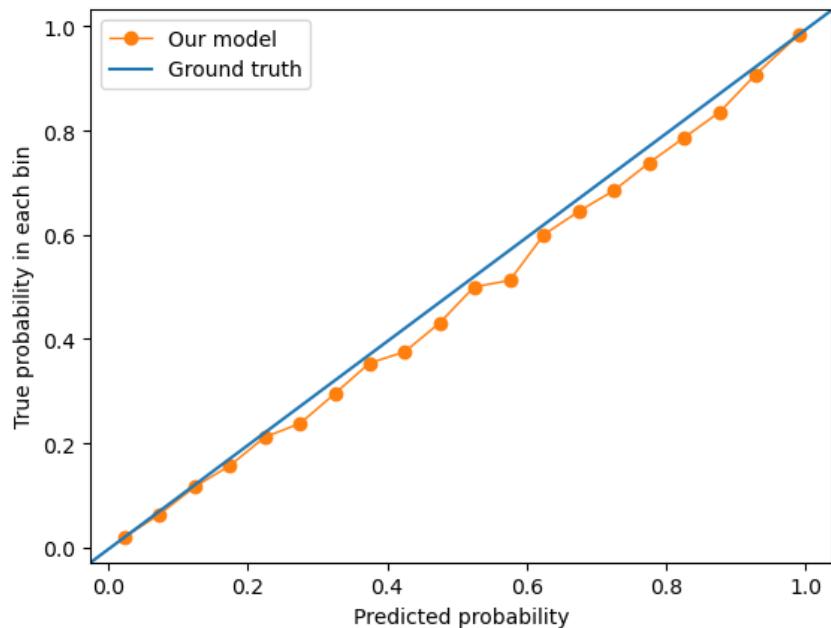


شکل ۳: تاثیر تعداد لایه‌های فریز شده از مدل distil-BERT در میزان عملکرد آن. بهترین مدل با ۳۰ لایه فریز شده به مقدار ۰.۹۱ در معیار عملکرد f1 دست پیدا کرده است.

### ۵ تحلیل عملکرد مدل

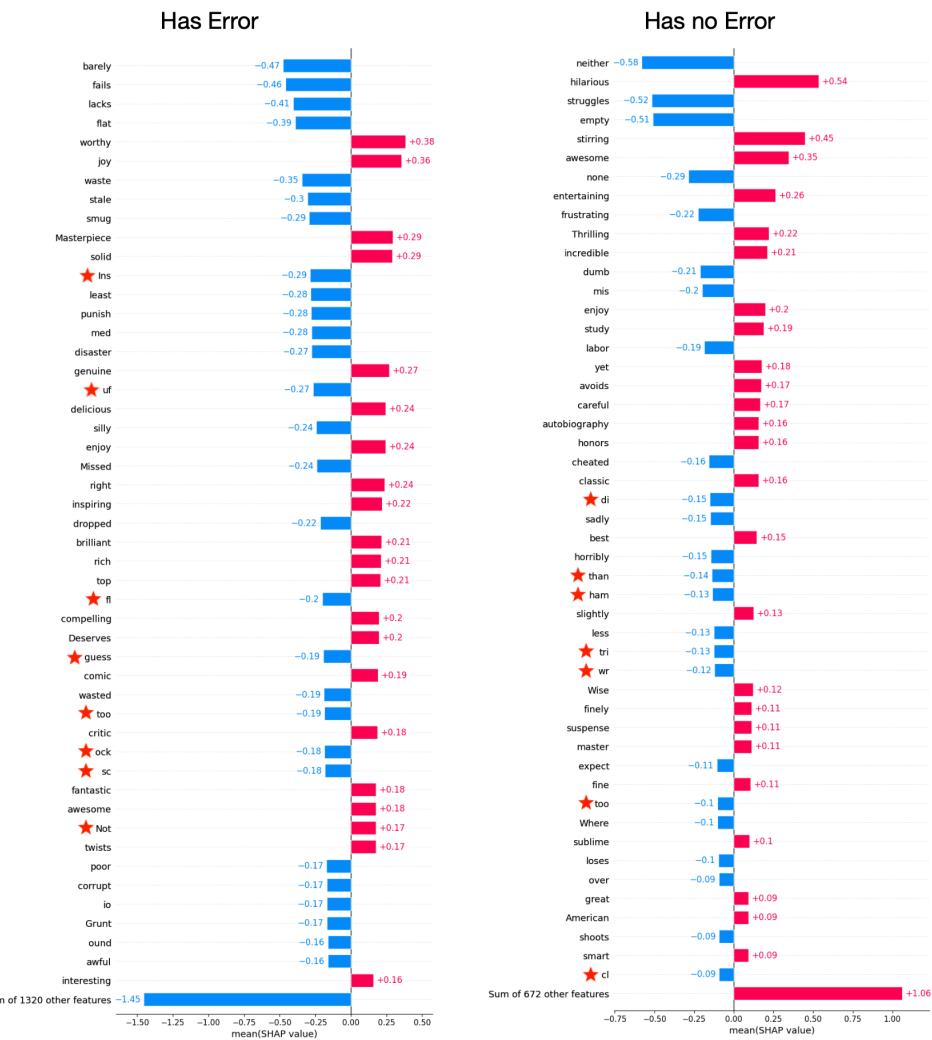
در این بخش بهترین مدل به دست آمده در مرحله قبل را انتخاب کرده و برای ۱۰۰ نمونه که مدل اشتباه کرده است و ۵۰ نمونه که مدل درست حدس زده است (از دادگان اعتبارسنجی) به کمک معیارهای Shapely [۲] مدل را تفسیر کردیم. ابتدا به بررسی میزان وزنی که به لغات در تصمیم‌گیری در دسته درست و دسته غلط داده است پرداختیم؛ همانطور که در شکل ۵ نیز مشخص است برخی لغات که با ستاره قرمز مشخص شده اند، هم در دسته ای که مدل درست پیش بینی کرده است و هم در دسته ای که مدل اشتباه پیش بینی کرده است، به اشتباه وزن مثبت یا منفی به آنها داده است. یعنی لغاتی هستند که به صورت کلی وزن خنثی دارند ولی مدل به آنها وزنی بسیار زیادی در جهت مثبت یا منفی داده است. البته که به صورت کلی همانطور

Calibration plot for validation data



شکل ۴: نمودار کالیبریشن (bins = 20) برای دادگان اعتبارسنجی، نمودار نارنجی احتمالات مربوط به مدل می باشد و نمودار آبی بهترین چیزی که باید باشد.

که مشخص است لغات را به خوبی تشخیص داده اما در بعضی لغاتی که اشاره شد اشتباه داشته است که این مسئله یکی از دلایل بروز خطا در تصمیم گیری مدل بوده است. که لازمه این مورد این است که پاکسازی های اولیه دادگان انجام شود ولی نه به صورت قبلی و با دقت خیلی بیشتر و بسیاری از جزئیات بایستی که در متن بماند و از طرف دیگر می توان از آگمنتیشن نیز برای حل این مسئله استفاده کرد. همچنین برای تمامی این نمونه ها، نمودار دیگری نیز رسم شده است که ضمیمه شده و به صورت خاص چند مودر از آن در شکل ۶ قابل مشاهده است که شامل مواردی است که مدل اشتباه پیش بینی کرده است. مبتنی بر این نمودار و بررسی متعدد نمونه ها می توان به این نتیجه رسید که بعضی از متون چند قسمتی هستند و علاوه بر این بعضی مواقع از معانی کنایی برای بیان هدف استفاده شده است. فایل های کامل html موارد بالا که شامل دادگان درست model/best-model/error-label1.html پیش بینی شده و اشتباه پیش بینی شده است در فایل های model/best-model/true-label1.html و شده و درست پیش بینی شده اند.



شکل ۵: لغات پر تکرار در متن به همراه وزنی که به آن‌ها توسط مدل داده شده است. شکل سمت راست، برای دادگان درست پیش‌بینی شده و شکل سمت چپ، برای دادگان اشتباه پیش‌بینی شده است. برای مواردی که به نظر اشتباه‌ها وزن در نظر گرفته شده بود، ستاره قرمز درج شده است.

## ۶ تحلیل خطای

بر روی این ۱۵۰ نمونه‌ای که در قسمت ۵ به آن اشاره شد به صورت مفصل تحلیل خطای انجام دادیم که با کمک بررسی احتمالات کلاس‌های پیش‌بینی شده، متن نظرات و نمودارهای به دست آمده در بخش قبل بوده است و در نهایت ۷ خطای احتمالی یافتیم که در آن‌ها تکرار شده بود. به همین منظور ابتدا بر روی ۱۰۰ نمونه که مدل اشتباه حدس زده بود و ۵۰ نمونه که درست حدس زده بود بررسی کردیم تا ببینیم چه مقدار به صورت دقیق این اrorها تکرار شده‌اند. همچنین ۵۰ نمونه دیگر از مواردی که مدل اشتباه کرده بود را به جز این ۱۰۰ مورد، به صورت تصادفی انتخاب کردیم و بر روی آن‌ها نیز بررسی کردیم. نتیجه به این صورت بود که تقریباً ۴ مورد از اrorهایی که پیدا کرده بودیم تکرار زیادی در نمونه‌های اشتباه حدس زده شده داشتند؛ که عبارت اند از:



شکل ۶: برخی از مواردی که مدل اشتباه کرده است با جزئیات تاثیر هر کدام از کلمات در این تصمیم گیری بررسی شده است که در بخش تحلیل خطا یکی از مواردی که مبتنی بر آن خطاهای را یافته ایم با استفاده از این نمودار بوده است.

۱. دو بخشی بودن متن: به این معنی که یک بخش از متن کاملاً مثبت است و بخش دیگری از متن کاملاً منفی است در نهایت مدل نمی‌تواند مفهوم کلی متن را به درستی استخراج کند که مثبت است یا منفی.

۲. وجود کلمات در تضاد با بار کلی جمله: برای مثال در یک متنی که هدف کلی مثبت است، از لغات متعددی که بار منفی دارند استفاده شده است.

۳. رفرنس دادن به فیلم‌ها یا مضماین مرتبط با فیلم: با توجه به این که مدل BERT‌ای که استفاده کردیم روی یک فضای عمومی متن آموزش دیده است، امکان فهمیدن بسیاری از مواردی که ممکن است در زمینه فیلم و سریال به آن‌ها اشاره شود را ندارد و به همین دلیل درصورتی که به آن‌ها رفرنس داده شود و این رفرنس داد در تصمیم گیری مثبت یا منفی بودن نظر کلیدی باشد، مدل نمی‌تواند به درستی مضمون را درک کند.

۴. خنثی بودن: برخی از نظرات نه لزوماً مثبت بودند و نه منفی که این مسئله باعث می‌شد که مدل نتواند نسبت به آن‌ها به خوبی عمل کند. دلیل این که ما به سراغ مثبت و منفی رفتیم این بود که دیتاست مناسبی برای فیلم و سریال پیدا نکرده بودیم که شامل ۳ دسته مثبت و منفی و خنثی باشد. اما در ادامه می‌توان تلاش کرد که با برچسب زنی یا استفاده از ایده‌های semi-supervised دسته خنثی را نیز اضافه کرد.

در [این جدول](#) که برای تحلیل خطا استفاده شده است ، نسبت هر کدام از خطاهای و بررسی نمونه‌ها با جزئیات کامل قابل مشاهده است. (این جداول در قسمت `doc/phase2/error-analysis` نیز در دسترس اند.)

## مراجع

- [1] L. Biewald, “Experiment tracking with weights and biases,” 2020. Software available from [wandb.com](https://wandb.com).
- [2] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Advances in Neural Information Processing Systems 30* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds. ), pp.4765–4774, Curran Associates, Inc., 2017.