به نام خدا

گزارش پروژه پایانی مبانی رایانش نرم

دسته بندی قطبیت روی توئیت ها با شبکه عصبی MLP

محمد میرزانژاد

شماره دانشجویی ۸۳۰۵۹۶۰۲۷

رشته علوم تصمیم و مهندسی دانش

استاد : جناب آقای دکتر ویسی

فهرست

۱- توضیح مسئله

در این پروژه ، هدف دسته بندی قطبیت روی توئیت هاست که به دو دسته مثبت و منفی تقسیم میشوند. دیتاست مورد استفاده برای آموزش ، دیتاست Kaggle[۱] است که توئیت ها را با برچسب های مثبت و منفی علامت گذاری کرده است. داده ها در این دیتاست شامل ایموجی[۲] ، هشتگ[۳] و url هستند که باید به فرم استاندارد جهت پردازش تبدیل شوند. همچنین باید از توئیت ها ویژگی هایی مثل unigram و bigram استخراج شود که جهت ارائه به شبکه عصبی مناسب باشند.

۲- داده ها

فایل داده آموزش یک فایل csv است که شامل سه مولفه tweet_id، دسته بندی قطبیت (۱ به عنوان مثبت و ۰ به عنوان منفی) و متن tweet است. فایل داده تست نیز شامل دو مولفه tweet_id و متن tweet است. همچنین جهت بررسی دقت پیش بینی شبکه یک فایل csv حاوی مقادیر هدف مربوط به قطبیت صحیح توئیت های تست نیز ایجاد شده است. داده ها مخلوطی از کلمات ، ایموجی ها ، علائم ،URL و ارجاع به کاربران[۴] است . کلمات و ایموجی ها میتوانند قطبیت یک توئیت را مشخص کنند اما میتوان از URL ها و ارجاع به کاربران صرف نظر کرد. همچنین ممکن است کلمات مخلوطی از کلمات با املای غلط ، سجاوندهای اضافی و یا کلمات با تکرار یک یا چند حرف باشند که نیاز به پیش پردازش و استانداردسازی دارند.

دیتاست آموزش شامل ۱۰۰۰۰ توئیت و دیتاست تست شامل ۱۰۰۰ توئیت که از دیتاست Kaggle که خود شامل ۱۵۷۸۶۲۷ توئیت است، انتخاب شده است. در ذیل آمار استخراج شده از مجموعه داده آموزش آورده شده است.

Tweets => Total: ۱۰۰۰۰

Positive: ۴۱۸۷, Negative: ۵۸۱۳

User Mentions => Total: ۲۲۷۴, Avg: ۰.۲۲۷۴, Max: ۱۲

URLs => Total: ۶۸۹, Avg: ۰.۰۶۸۹, Max: ۴

Emojis => Total: ۱۴۱, Positive: ۷۵, Negative: ۶۶, Avg: ۰.۰۱۴۱, Max: ۲

[۱] http://thinknook.com/twitter-sentiment-analysis-training-corpus-dataset-/۲۲-۰۹-۲۰۱۲
[۲] Emoticon
[۳] Hashtag
[۴] User Mention

Words => Total: ۱۱۳۸۸۵, Unique: ۱۴۲۰۳, Avg: ۱۱.۳۸۸۵, Max: ۳۱, Min: ۰

Bigrams => Total: ۱۰۳۹۱۱, Unique: ۶۲۶۰۳, Avg: ۱۰.۳۹۱۱

۳- پیاده سازی

۳-۱ پیش پردازش

توئیت های خام که ویژگیهای مختلف از قبیل retweet ، ایموجی ، ارجاع کاربر و غیره دارند، جهت استفاده در شبکه عصبی باید نرمال سازی شوند. در ابتدا یک سری عملیات پیش پردازش عمومی انجام میشود :

- تبدیل توئیت به حروف کوچک
- تبدیل ۲ یا تعداد بیشتر نقطه به فاصله
- حذف فاصله ها و علائم نقل قول (" و ') از انتهای توئیت
- جایگزینی ۲ یا تعداد بیشتر کاراکتر فاصله (space) با یک تک فاصله

۳-۱-۱ URL

هایپرلینک ها معمولا در بحث دسته بندی اسناد مورد توجه قرار نمیگیرند لذا در توئیت ها با واژه URL جایگزین میشوند. Regular Expression مورد استفاده جهت تطبیق URL ها به این شکل است : ((www\.[\S]+)|(https?://[\S]+))

۳-۱-۲ ارجاع کاربر

ارجاع کابران به یکدیگر را با USER_MENTION جایگزین میکنیم. عبارت باقاعده جهت تطبیق به این شکل است : @[\S]+

۳-۱-۳ ایموجی ها

کاربران جهت بیان احساسات از شکلک های مختلف استفاده میکنند. در این پروژه سعی شده تا رایج ترین شکلک های مورد استفاده لحاظ شوند و به شرح جدول ذیل جایگزین شوند. شکلک ها با دو عبارت مثبت و منفی EMO_POS و EMO_NEG جایگزین میشوند.

Emoticon	Type	Regular expression	Replacement
:) , :) , :-) , (: , (: , (-: , :’)	Smile	(:\s?\)|:-\)|\(\s?:|\(-:|:\’\))	EMO_POS
:D , : D , :-D , xD , x-D , XD , X-D	Laugh	(:\s?D|:-D|x-?D|X-?D)	EMO_POS
;-) , ;) , ;-D , ;D , (; , (-;	Wink	(:\s?\(|:-\(|\)\s?:|\)-:)	EMO_POS
<۳ , :*	Love	(<۳|:*)	EMO_POS
:-(, : (, :(,): ,)-:	Sad	(:\s?\(|:-\(|\)\s?:|\)-:)	EMO_NEG
: ,(, :’(, :"(	Cry	(:,\(|:\’\(|:"\()	EMO_NEG

۳-۱-۴ Hashtag

هشتگ ها کلمات بدون فاصله ای هستند و با علامت # شروع میشوند که برای ذکر و ترویج یک موضوع روز یا مورد بحث توسط کاربران به طور مداوم استفاده میشوند. تمام هشتگ ها را با کلمه اصلی آن Hastag بدون علامت # جایگزین میکنیم . برای مثال #hello را با hello جایگزین میکنیم. عبارت منظم جهت جایگزینی این مورد ، (\S+)# است.

۳-۱-۵ Retweet

Retweetها توئیت هایی هستند که توسط سایر کاربران فرستاده و به اشتراک گذاشته میشوند. این دسته از توئیت ها با RT شروع میشوند که ما در اینجا آن را حذف میکنیم. بعد از پیش پردازش ، خود توئیت ها به ترتیب زیر پردازش و روی آن ها اصلاحاتی صورت میگیرد:

- حذف نقطه گذاری ها و علائم سجاوندی [’"?!,.():;]
- تبدیل تکرار ۲ بار یا تعداد بیشتر حروف به یک حرف. برای مثال I am sooooo happppppy به I am so happy تبدیل میشود.
- حذف – و ' . برای مثال t-shirt و their's به tshirt و theirs تبدیل میشوند.
- بررسی معتبر بودن کلمه، یعنی کلمه با یکی از حروف الفبا شروع شود و ادامه یابد و در ادامه آن عدد یا نقطه (.) و یا زیرخط[۵] (_) بیاید.

[۵] Underscore

۳-۲ استخراج ویژگی Feature Extraction

در این پروژه دو نوع ویژگی از داده ها استخراج میشود که عبارتند از unigram و bigram . در ابتدا یک توزیع فراوانی [۶] از unigram و bigram های موجود در دیتاست ایجاد میشود و سپس رایج ترین و فراوان ترین bigram ها وunigram ها برای تحلیل انتخاب میشوند. در ابتدا به تعریف مفاهیم unigram و bigram میپردازیم.

تعریف Unigram و Bigram :

N-gram ها مدل های زبانی مبتنی بر احتمالات هستند که از n-۱ کلمه قبل جهت پیش بینی وقوع کلمه بعد استفاده میکنند. در واقع هدف این است که اگر زبانی T کلمه متمایز داشته باشد احتمال اینکه کلمه x بعد از کلمه y بیاید چقدر است. در Unigram ، احتمال وقوع x در یک متن با توجه تکرار وقوع آن در مجموعه ای از داده های مورد بررسی (مثلا دیتای آموزش) تخمین زده میشود. مثلا احتمال کلی وقوع بعضی کلمات مثل popcorn از کلمکاتی مثل unicorn بیشتر است. در Bigram احتمال وقوع x با توجه وقوع و حضور کلمه قبلی ، تخمین زده میشود . مثلا mythical unicorn محتمل تر از mythical popcorn است . یعنی هر گاه کلمه قبلی mythical باشد احتمال وقوع کلمه unicorn بیشتر است. در واقع در bigram ، احتمال یک دنباله کلمات برابر است با ضرب احتمال شرطی bigram های آن . به عنوان مثال :

$$P(the,mythical,unicorn) = P(unicorn|mythical)\ P(mythical|the)\ P(the|<start>)$$

۳-۲-۱ Unigram در پروژه

ساده ترین و رایج ترین ویژگی مورد استفاده برای دسته بندی اسناد ، حضور انفرادی کلمات و علائم در متن است. کلمات را به صورت انفرادی از مجموعه آموزش استخراج میکنیم و یک توزیع فراوانی از این کلمات را بدست می آوریم.در مجموع ۱۴۲۰۳ کلمه منحصر به فرد از مجموعه دیتای آموزش استخراج شد. برای مجموعه داده ۱۰۰۰۰ تایی از توئیت ها ، جهت ساخت Vocabulary از ۲۰۰ کلمه با توزیع فراوانی بالا در حالت دسته بندی sparse vector استفاده شده است و در حالت دسته بندی dense vector از ۱۰۰۰ کلمه اول با توزیع فراوانی بالا استفاده شده است.

[۶] Frequency Distribution

۳-۲-۲ Bigram در پروژه

Bigram ها زوج کلماتی هستند که پشت سر هم در اکثر متون می آیند. تعداد ۶۲۶۰۳ Bigram منحصر به فرد از دیتاست استخراج شد که از این تعداد ما ۱۰۰۰ Bigram اول با میزان تکرار بالا را جهت تشکیل vocabulary استفاده کردیم.

۳-۳ ارائه ویژگی Feature Representation

بعد از استخراج unigram ها و bigram ها ، هر توئیت را به صورت یک بردار ویژگی (Feature Vector) و به یکی از دو فرم نمایش sparse وdense بسته به نوع دسته بندی نمایش میدهیم.

۳-۳-۱ فرم نمایش Sparse Vector

در صورتی که فقط از unigram استفاده شود ، sparse vector مربوط به نمایش هر توئیت به طول ۱۵۰۰ خواهد بود ودر صورت استفاده توام از unigram و bigram ، به طول ۲۵۰۰ خواهد بود . به هر unigram یا bigram با توجه به رتبه آن (توزیع فراوانی آن) یک index یکتا داده میشود. بردار ویژگی یک توئیت در اندیس هایی از unigram و bigram که در آن توئیت حضور دارند ، مقدار مثبت و در سایر جاها مقدار صفر دارد و به همین علت است که sparse نامیده میشود. این مقدار مثبت اندیس های unigram یا Bigram بستگی به نوع ویژگی ای دارد که ما در نظر میگیریم یعنی میتواند منعکس کننده حضور یا تکرار آنها باشد که در ذیل شرح داده میشود:

- حضور (Presence) : در این نوع نمایش ، بردار ویژگی در اندیس های Unigram و Bigram که در توئیت وجود دارند ، عدد ۱ را در بر دارد.
- تکرار (frequency) : در این حالت بردار ویژگی میزان تکرار unigram و bigram در اندیس های مرتبط در یک توئیت را منعکس میکند و در صورت عدم وقوع حاوی عدد صفر است. یک ماتریس از بردار های term-frequency ، برای کل مجموعه داده آموزش ساخته میشود و هر تکرار کلمه (term frequency) با استفاده از شاخص idf [۷] ، مقدارد دهی میشود و کلمات مهمتر مقدار بیشتری دریافت میکنند. شاخص idf یک کلمه به صورت زیر تعریف میشود :

$$idf(t) = \log\left(\frac{1+n_d}{1+df(d,t)}\right)+1$$

[۷] inverse-document-frequency

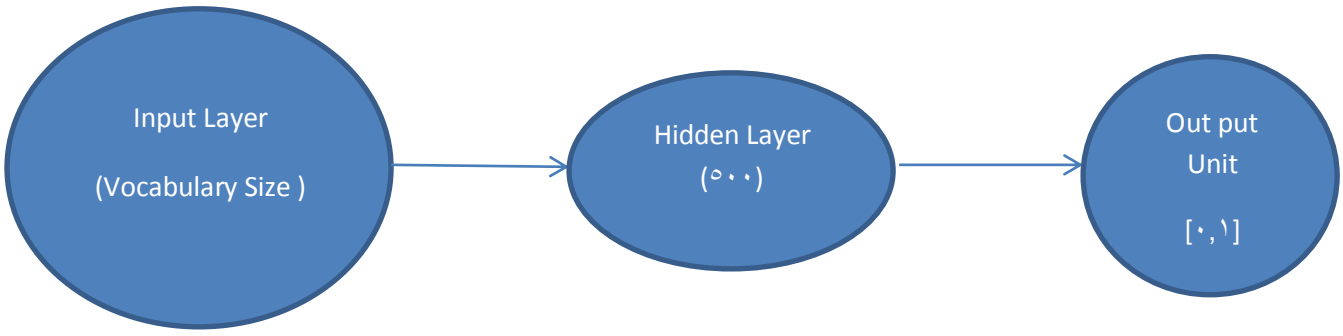
که nd تعداد کل متن ها و df(d,t) تعداد متونی است که کلمه t در آن ها حضور دارد.

۲-۳-۳ فرم نمایش Dense Vector

برای فرم نمایش dense ، از vaocabulary متشکل از Unigram ها به اندازه ۱۰۰۰ ، یعنی ۱۰۰۰ لغت با تکرار بالا در دیتاست . در واقع به هر کلمه با توجه به رتبه آن یک عدد صحیح انتساب دادیم ، یعنی به رایج ترین کلمه با بیشترین تکرار عدد ۱ اختصاص یافت و به دومین پرتکرار ترین کلمه رتبه ۲ و الی آخر ... . در نهایت هر توئیت با برداری حاوی این اندیس ها ارائه میشود که dense vector نام دارد.

۳-۴ شبکه عصبی MLP

با استفاده از کتابخانه های TensorFlow و Keras ، شبکه عصبی با یک لایه مخفی و ۵۰۰ نرون مخفی پیاده سازی شد. خروجی شبکه یک مقدار واحد است که به تابع سیگموید فرستاده میشود و مقداری بین ۰ تا ۱ تولید میشود که در واقع احتمال مثبت یا منفی بودن توئیت را نشان میدهد. تعداد epoch های اجرا شده برای شبکه ۱۰۰۰تکرار است. ساختار شبکه در ذیل آمده است :



۳-۵ روند اجرای برنامه

نیازمندیهای پروژه :

برنامه با زبان Python ۳.۵.۲ نوشته شده است. جهت اجرای برنامه ، کتابخانه های زیر باید دانلود و نصب شوند که همگی از طریق اجرای دستور pip install از command line قابل نصب و بهره برداری هستند.

-TensorFlow

-Keras

-NLTK

-Numpy

۳-۵-۱ ماژول preprocess

در ابتدا باید ماژول preprocess.py جهت انجام عملیات پیش پردازش ذکر شده در ۳-۱ روی داده آموزش و آزمون اجرا شود . خروجی این ماژول دو فایل train-processed.csv و test-processed.csv است.

۳-۵-۲ ماژول stats

این ماژول وظیفه تولید Unigram و bigram ها را بر حسب فرم نمایش تعداد تکرار در توئیت را بر عهده دارد . همچنین vocabulary کلمات منحصر به فرد نیز توسط این فایل تولید میشود . در واقع سه فایل train-processed-unique.txt حاوی vocabulary کلمات و دو فایل train-processed-freqdist و train-processed-freqdist-bi به ترتیب نماینده Unigram و bigram های تولید شده از روی داده آموزش هستند.

۳-۵-۳ ماژول neuralnet

در این ماژول در ابتدا bigram و Unigram های با تکرار بالا و با رتبه بالا انتخاب میشوند پس از پردازش توئیت ها و استخراج بردارهای ویژگی ابتدا شبکه با مجموعه داده آموزش ، Train میشود و مدل ساخته میشود. سپس داده های تست جهت پیش بینی به مدل ساخته شده ارسال میشوند و

نتایج پیش بینی در فایل prediction.csv ذخیره میشوند. همچنین دقت پیش بینی در خروجی درج میشود.

۴- نتایج

شبکه پس از آموزش روی ۱۰۰۰۰ توئیت روی یک مجموعه ۱۰۰۰ تایی از توئیت ها تست شد که نتایج ذیل را در بر داشت :

فرم نمایش مبتنی بر تکرار کلمات		فرم نمایش مبتنی بر حضور کلمات		
Unigram + Bigram	Unigram	Unigram + Bigram	Unigram	
۷۶.۹	۷۶.۴	۷۷.۱	۷۶.۲	دقت پیش بینی

۵- منابع

[۱] Tan, Chade-Meng & Wang, Yuanfang & Lee, Chan-Do. (۲۰۰۲). The Use of BiGrams to Enhance Text Categorization. Information Processing & Management. ۳۸. ۵٤٦-۵۲۹. ۱۰.۱۰۱٦/S٤٥٧۳-۰۳۰٦(۰۱).۰-۰۰۰٤٥

[۲] Qasem, Mohammed & Thulasiram, Ruppa & Thulasiram, Parimala. (۲۰۱۵). Twitter sentiment classification using machine learning techniques for stock markets. ۸۳٤-۸٤۰. ۱۰.۱۱۰۹/ICACCI.۲۰۱۵.۷۲۷۵۷۱٤.

[۳] Mansour R., Hady M.F.A., Hosam E., Amr H., Ashour A. (۲۰۱۵) Feature Selection for Twitter Sentiment Analysis: An Experimental Study. In: Gelbukh A. (eds) Computational Linguistics and Intelligent Text Processing. CICLing ۲۰۱۵. Lecture Notes in Computer Science, vol ۹۰٤۲. Springer, Cham