

**دانشگاه فنی و حرفه ای**

**دانشکده فنی و حرفه ای دختران تهران- دکتر شریعتی**

**پایان­نامه کارشناسی ناپیوسته**

**رشته کامپیوتر گرایش مهندسی حرفه‌ای نرم افزار**

**عنوان:**

**مدل‌سازی جوامع عاطفی و تشخیص احساسات**

**استاد راهنما:**

سرکار خانم دکتر زهرا ولدخانی

نگارش:

زهرا رحیمیان

پاییز 1402

تقدیم به:

آنان که آرزو داشتند وطن جایی برای ماندن بود...

**تشکر و قدردانی**

**سپاسگزارم از استاد گرامی‌ام که من را به سمت یکی از مباحث روز و ارزشمند راهنمایی کردند و همچنین سپاسگزارم از خانواده‌ عزیزم که همواره حامی و پشتیبان من بوده‌اند.**

# چکیده

این پایان‌نامه به هدف شناسایی و تحلیل جوامع عاطفی در شبکه‌های اجتماعی از طریق تکنیک‌های پیشرفته علم داده و ماشین لرنینگ می‌پردازد. با استفاده از دیتاست‌های معتبر و روش‌های نوین پردازش متن، رویکردی نوآورانه برای تحلیل احساسات کاربران توسعه یافته است. یافته‌های تحقیق حاکی از آن است که مدل‌های پیاده‌سازی شده قادر به تشخیص دقیق و اتوماتیک حالات عاطفی متن‌های فارسی هستند. نتایج حاصل از این پژوهش، دلالت‌های مهمی برای طراحی سیستم‌های توصیه‌گر و تجزیه و تحلیل رفتار کاربران در پلتفرم‌های دیجیتال دارد و پیشنهادهایی برای بهبود سیستم‌های موجود و توسعه روش‌های تحلیل متن در زبان فارسی ارائه می‌دهد.

**کلیدواژه:** علم داده، ماشین لرنینگ، تشخیص احساسات، تشخیص جامعه

فهرست نوشتار

فصل اول: «مقدمه»

[1-1 انگیزه 1](#_Toc155972264)

[1-2 هدف 1](#_Toc155972265)

[1-3 رئوس مطالب سایر فصل‌ها 1](#_Toc155972266)

فصل دوم: «تجزیه و تحلیل نیازمندی‌ها»

[2-1 مقدمه 3](#_Toc155972268)

[2-2 نتیجه‌گیری 3](#_Toc155972269)

فصل سوم: «تشخیص جامعه عاطفی در شبکه های اجتماعی»

[3-1 چکیده 5](#_Toc155972272)

[3-2 مقدمه 5](#_Toc155972273)

[3-3 تعریف اجتماعات مشتق شده 6](#_Toc155972274)

[3-4 تحلیل احساسات و شناسایی جوامع 6](#_Toc155972275)

[3-5 آشنایی با مقیاس اکمن 7](#_Toc155972276)

[3-6 استفاده از الگوریتم‌های تشخیص جامعه برای تحلیل رفتار عاطفی کاربران 7](#_Toc155972277)

[3-6-1 فرآیند تشخیص احساس پست 8](#_Toc155972278)

[3-6-2 تحلیل پروفایل کاربر و سنجه تاثیرگذاری 8](#_Toc155972279)

[3-6-3 رفتار احساسی کاربر 10](#_Toc155972280)

[3-6-4 تشخیص جامعه 11](#_Toc155972281)

[3-6-5 نمایش نمودار احساسی 12](#_Toc155972282)

[3-7 نتیجه‌گیری 13](#_Toc155972283)

[فصل](#_Toc155972284) چهارم: «پیاده سازی»

[4-1 مقدمه 15](#_Toc155972285)

[4-2 معرفی دیتاست استفاده شده 15](#_Toc155972286)

[4-3 کدنویسی 16](#_Toc155972287)

[4-3-1 نصب نیازمندی‌ها و اضافه کردن کتابخانه‌های لازم 16](#_Toc155972288)

[4-3-2 وارد کردن دیتاست و خواندن محتوای آن 18](#_Toc155972289)

[4-3-3 پیش پردازش 21](#_Toc155972290)

[4-3-4 بردارسازی متن 28](#_Toc155972291)

[4-3-5 ساخت و ارزیابی مدل 29](#_Toc155972292)

[4-3-6 مصورسازی 34](#_Toc155972293)

[4-3-7 ذخیره‌سازی مدل آموزش داده شده 35](#_Toc155972294)

[4-3-8 استقرار مدل با پکیج Flask 36](#_Toc155972295)

[4-3-9 کدهای HTML, CSS, js 37](#_Toc155972296)

[4-4 اجرای نرم افزار 37](#_Toc155972297)

[4-5 نتیجه‌گیری 39](#_Toc155972298)

[فصل](#_Toc155972299) پنجم: «جمع‌بندی و پیشنهادها»

[5-1 نتیجه گیری 41](#_Toc155972300)

[5-2 پیشنهادهایی برای کارهای آتی 41](#_Toc155972301)

[فهرست منابع 42](#_Toc155972302)

چکیده انگلیسی .................................................................................................................................................... 43

فهرست شکل­ها

[شکل 1-3 معماری سیستم 7](#_Toc155972759)

[شکل 2-3 معماری ابزار تشخیص احساس 8](#_Toc155972760)

[شکل 3-3 تبدیل به گراف خطی 12](#_Toc155972761)

[شکل 1-4 دیتاست کامنت‌های اسنپ فود 15](#_Toc155972762)

[شکل 2-4 نصب Hazm روی سیستم لوکال 16](#_Toc155972763)

[شکل 3-4 نصب و وارد کردن کتابخانه‌های ضروری 17](#_Toc155972764)

[شکل 4-4 وارد کردن دیتاست 18](#_Toc155972765)

[شکل 5-4 بررسی ساختار دیتافریم 19](#_Toc155972766)

[شکل 6-4 نمایش آمار توصیفی دیتاست 20](#_Toc155972767)

[شکل 7-4 بررسی مقادیر یکتای دیتاست 21](#_Toc155972768)

[شکل 8-4 حذف ستون Unnamed 22](#_Toc155972769)

[شکل 9-4 فیلتر کردن سطرها 22](#_Toc155972770)

[شکل 10-4 تبدیل نوع داده 23](#_Toc155972771)

[شکل 11-4 جداسازی نظرات و برچسب‌ها 24](#_Toc155972772)

[شکل 12-4 تقسیم داده‌ها به دو مجموعه آموزشی و تست 24](#_Toc155972773)

[شکل 13-4 نرمال‌سازی داده 25](#_Toc155972774)

[شکل 14-4 توکن‌سازی کلمات 26](#_Toc155972775)

[شکل 15-4 ریشه یابی کلمات 26](#_Toc155972776)

[شکل 16-4 واژه‌یابی 27](#_Toc155972777)

[شکل 17-4 تبدیل کلمات به رشته متنی 27](#_Toc155972778)

[شکل 18-4 بردارسازی متن 28](#_Toc155972779)

[شکل 22-4 مدل‌های ساخته شده 30](#_Toc155972780)

[شکل 23-4 ویژگی‌های XGBoost 32](#_Toc155972781)

[شکل 24-4 ساخت Pipeline 33](#_Toc155972782)

[شکل 25-4 کدهای مصورسازی 34](#_Toc155972783)

[شکل 26-4 نتیجه مصورسازی 35](#_Toc155972784)

[شکل 27-4 ساخت فایل pickle 35](#_Toc155972785)

[شکل 28-4 کد پکیج فلسک 36](#_Toc155972786)

[شکل 29-4 بخشی از کد رابط کاربری 37](#_Toc155972787)

[شکل 30-4 وب‌سایت تشخیص احساس کامنت 38](#_Toc155972788)

[شکل 31-4 وارد کردن نظر مثبت و دیدن نتیجه 38](#_Toc155972789)

[شکل 32-4 وارد کردن نظر منفی و دیدن نتیجه 39](#_Toc155972790)

فصل اول

«مقدمه»

1-1 انگیزه

اهمیت و ضرورت تحقیق در زمینه تحلیل احساسات در داده‌های فارسی زبان از این واقعیت ناشی می‌شود که فضای مجازی به عرصه‌ای برای بیان نظرات و احساسات کاربران تبدیل شده است. با توجه به حجم عظیم داده‌های تولیدی در شبکه‌های اجتماعی، استفاده از تکنیک‌های پیشرفته یادگیری ماشین و پردازش زبان طبیعی برای شناسایی و تحلیل احساسات، نه تنها برای سازمان‌ها و کسب‌وکارها بلکه برای مطالعات اجتماعی و روانشناختی حیاتی است. مطالعات مشابه، اگرچه در سایر زبان‌ها به وفور انجام شده‌اند، اما در حوزه زبان فارسی هنوز جای کار بسیار دارند و با چالش‌هایی مانند کمبود منابع داده‌ای معتبر و ابزار تحلیل مناسب روبرو هستند.

1-2 هدف

این تحقیق به دنبال پر کردن خلأ پردازش زبان طبیعی با زبان فارسی است و با ارائه یک مدل دقیق و کارآمد، در راستای بهبود درک ما از جامعه دیجیتال فارسی‌زبان گام برمی‌دارد.

1-3 رئوس مطالب سایر فصل‌ها

این تحقیق دارای فصل‌های مقدمه، تجزیه و تحلیل نیازمندی‌ها، تشخیص جامعه عاطفی در شبکه‌های اجتماعی، پیاده‌سازی و جمع بندی و پیشنهادات است.

فصل دوم  
«تجزیه و تحلیل نیازمندی‌ها»

2-1 مقدمه

در این پروژه، از ابزارها و محیط‌های توسعه متنوعی برای اجرای بخش‌های مختلف کار استفاده شده. برای اجرای کدهای پکیج فلسک، از محیط توسعه اسپایدر استفاده شده است. برای نوشتن کدهای وب پیج، از محیط توسعه vscode استفاده شده است. به کارگیری نوت‌بوک کولب برای پیاده‌سازی و تحلیل داده‌ها نشان‌دهنده تمایل به استفاده از ابزارهای تحلیل داده در محیط ابری است. استفاده از مقالات دسترسی‌پذیر از گوگل اسکولار برای ترجمه و تحلیل محتوا نیز بیانگر تمرکز بر تحقیق و استفاده از منابع علمی معتبر است. در نهایت، از یک دیتاست‌ معتبر موجود در کگل برای تأمین داده‌های لازم برای تحقیق یک استفاده شده است.

2-2 نتیجه‌گیری

‌در این فصل ابزاهای استفاده شده برای نگارش این پایان نامه و انجام پروژه عملی معرفی شدند. در فصل بعد به بررسی دیتاست و داده‌های استفاده شده پرداخته شده است.

فصل سوم

«تشخیص جامعه عاطفی در شبکه‌های اجتماعی»

3-1 چکیده

بررسی و تحلیل شبکه‌های اجتماعی به‌طور گسترده‌ به عنوان یک حوزه تحقیقی پیچیده و چالش‌برانگیز شناخته شده است. نمونه بارز این موضوع، سازماندهی رأس‌ها در خوشه‌ها است که بسیاری از یال‌ها رأس‌های همان خوشه را به هم متصل می‌کنند و تعداد کمتری یال رأس‌های خوشه‌های مختلف را به هم می‌رسانند. این موضوع شامل جنبه‌ای اساسی است که مربوط به تشخیص اجتماعات کاربران می‌شود. در حوزه‌هایی نظیر جامعه‌شناسی و علوم کامپیوتر که تعاملات و ارتباطات اغلب به صورت گراف‌ها نمایش داده می‌شوند، تشخیص اجتماعات از اهمیت ویژه‌ای برخوردار است. مقیاس عاطفی اکمن نقطه کلیدی است که با آن رفتار عاطفی کاربران در توییت‌هایشان تحلیل می‌شود. در نتیجه، اجتماعات مشتق‌شده با استفاده از سه معیار مختلف ارزیابی می‌شوند، در حالی که نسخه وزن‌دار یک الگوریتم تشخیص اجتماعات مدولاریتی مورد استفاده قرار می‌گیرد. شواهد کافی وجود دارد که نشان می‌دهد روش پیشنهادی ما اجتماعات تأثیرگذاری ایجاد می‌کند.

3-2 مقدمه

در چند سال گذشته، توییتر به همراه سایر رسانه‌های اجتماعی محبوبیت فزاینده‌ای پیدا کرده است. این امر منجر به ایجاد دغدغه‌های پژوهشی عظیم و فرصت‌های جدیدی برای مطالعه تعاملات متقابل گروه‌های مختلف افراد شده است. شناسایی جوامع و تحلیل احساسات دو نمونه از این موارد هستند که به عنوان موضوعات محبوب در بررسی و درک بهتر شبکه‌های اجتماعی شناخته می‌شوند. از یک طرف، شناسایی جوامع در تلاش است تا شبکه‌های اجتماعی را با هدف اصلی شناسایی خوشه‌هایی از کاربران مرتبط و وابسته به یکدیگر تحلیل کند، و از طرف دیگر، تحلیل احساسات سعی دارد تا به رفتار کاربران در سطح احساسی پی ببرد و در نتیجه نگرش آن‌ها را نسبت به موضوعات متنوعی مانند نحوه احساس افراد تعیین کند. تعیین رفتار کاربران در هر یک از جوامع ظهور یافته و همچنین در کل شبکه، یکی از اساسی‌ترین جنبه‌های تحلیل شبکه‌های اجتماعی است و مفهومی پویا برای تجزیه و تحلیل دقیق روش‌های ارتباط کاربران برای ایجاد جوامع اجتماعی می‌باشد. برای توضیح دینامیک‌های اجتماعی تعامل بین گروه‌های افراد، لازم است که ما ساختار جامعه شبکه را مطالعه کنیم.

3-3 تعریف اجتماعات مشتق شده

اجتماعات مشتق‌شده به گروه‌های یا خوشه‌هایی اشاره دارد که از طریق تجزیه و تحلیل داده‌های شبکه‌های اجتماعی به دست آمده‌اند. این اجتماعات می‌توانند مجموعه‌هایی از کاربران باشند که بر اساس رفتار یا ویژگی‌های مشابهشان در شبکه‌های اجتماعی شناسایی شده‌اند. برای سنجش و بررسی دقت و کارآیی این اجتماعات، سه روش مختلف ارزیابی به کار رفته است. این معیارها ممکن است شامل مواردی مانند تراکم ارتباطات درون اجتماع، میزان همپوشانی با اجتماعات واقعی، و یا معیارهای دیگری باشند که برای ارزیابی کیفیت تشخیص اجتماعات در نظر گرفته شده‌اند.

در تحلیل اجتماعات، از یک الگوریتم خاص که بر مبنای مدولاریتی (یک مفهوم در شبکه‌های اجتماعی که برای اندازه‌گیری قدرت یک تقسیم‌بندی شبکه به اجتماعات استفاده می‌شود) کار می‌کند، استفاده شده است. عبارت وزن‌دار به این معنی است که در حین تحلیل داده‌ها، برخی از عوامل یا مشخصات خاص با اهمیت بیشتری در نظر گرفته شده‌اند.

3-4 تحلیل احساسات و شناسایی جوامع

روش‌های شناسایی جامعه کاربران در شبکه‌های اجتماعی نقش مهمی در ارتقاء استراتژی‌های اقتصادی و بازاریابی دارند. این روش‌ها با شناسایی دقیق گروه‌های کاربران مرتبط و تحلیل رفتار و ترجیحات آن‌ها، امکان هدف‌گیری و اجرای طرح‌های تبلیغاتی موثرتر در شبکه‌های خاص را فراهم می‌کنند. بدین ترتیب، ارائه توصیه‌ها و محتواهای سازگار با علایق و نیازهای کاربران میسر می‌شود. علاوه بر این، تحلیل نگرش‌ها و تصمیمات مردم بر اساس محتوای موجود در شبکه‌های اجتماعی اهمیت ویژه‌ای دارد، زیرا این نگرش‌ها می‌توانند تحت تأثیر جو عمومی شکل گرفته و بر تمام جنبه‌های زندگی تأثیرگذار باشند. شناسایی وضعیت احساسی کاربران از طریق تحلیل محتوای تولید شده توسط آن‌ها در شبکه‌های اجتماعی نه تنها یک چالش است بلکه می‌تواند بینش‌های مهمی در مورد وضعیت احساسی یک جامعه یا منطقه ارائه دهد. در نهایت، درک این جنبه‌های احساسی رفتار کاربران به ما کمک می‌کند تا جوامع را با دقت بیشتری شناسایی کرده و ساختارهای اجتماعی متراکم‌تر و معنادارتری را توسعه دهیم.

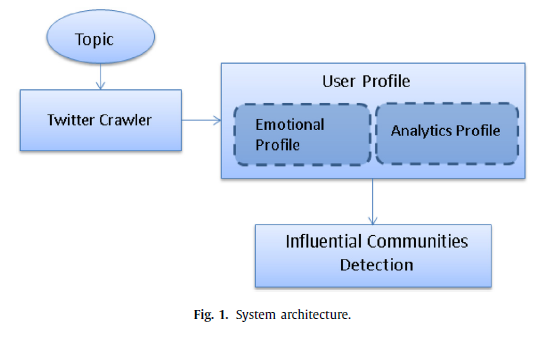
3-5 آشنایی با مقیاس اکمن

مقیاس Ekman ، که توسط پل اکمن، روان‌شناس برجسته، توسعه یافته است، به شناسایی شش هیجان اصلی انسانی می‌پردازد: خشم، تنفر، ترس، شادی، غم، و تعجب. این مقیاس بر این فرض استوار است که این هیجان‌ها در فرهنگ‌های متفاوت سراسر جهان به صورت بنیادین وجود دارند و در بیانات چهره و زبان بدن به طور مشابهی تجلی می‌یابند.

استفاده از مقیاس اکمن تنها به مطالعات روان‌شناختی محدود نمی‌شود. در عرصه‌هایی چون بازاریابی، تحلیل رفتار مصرف‌کننده، و حتی در حوزه‌های پیشرفته‌تری مانند هوش مصنوعی و تحلیل داده‌های بزرگ، از این مقیاس بهره گرفته می‌شود. به خصوص در هوش مصنوعی، این مقیاس برای شناسایی و تحلیل احساسات در متون، صداها، و تصاویر کاربرد دارد.

3-6 استفاده از الگوریتم‌های تشخیص جامعه برای تحلیل رفتار عاطفی کاربران

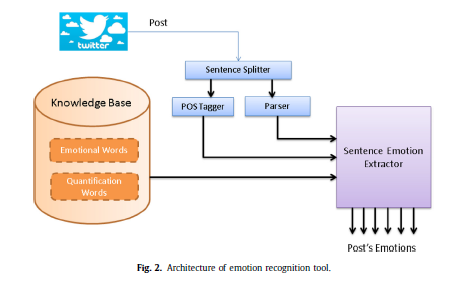
در این بخش، روش معرفی‌شده برای تجزیه و تحلیل و مدل‌سازی گفتگوها در موضوعات خاص در شبکه توییتر ارائه می‌شود. این روش توییت‌های کاربران را تجزیه و تحلیل کرده و محتوای احساسی آن‌ها را بر اساس مقیاس اکمن ششگانه مشخص می‌کند که شامل شش احساس انسانی اساسی: «خشم، انزجار، ترس، شادی، غم، تعجب» می‌شود. پس از آن، نفوذ کاربر در شبکه برای شناسایی جوامع تاثیرگذارتر در شبکه مشخص شده تعیین می‌گردد. جوامع تاثیرگذار استخراج‌شده بر اساس محتوای احساسی پست‌ها و همچنین نفوذ کاربران استفاده می‌شوند و پس از آن به عنوان نمایندگی تعاملات احساسی در شبکه مربوطه در نظر گرفته می‌شوند. معماری سیستم پیشنهادی در شکل زیر نشان داده شده است.



شکل 1-3 معماری سیستم

3-6-1 فرآیند تشخیص احساس پست

در ابتدا، پست‌های توییتر وارد سیستم شده و سپس توسط یک جداکننده جمله به جملات مجزا تقسیم می‌شوند. هر جمله به نوبه خود توسط یک برچسب‌گذار اجزای گفتار مورد تجزیه و تحلیل قرار می‌گیرد که به هر کلمه نقش دستوری می‌دهد. سپس، توسط یک تجزیه‌کننده، ساختار دستوری جملات تجزیه شده و نمودار وابستگی ساخته می‌شود که روابط معنایی و دستوری کلمات را در جمله نشان می‌دهد. این اطلاعات همراه با دانش موجود در پایگاه دانش که شامل کلمات احساسی و کلمات کمیت‌سنجی است، به یک استخراج‌کننده احساسات جمله ارسال می‌شود که قدرت احساسی هر کلمه و وضعیت احساسی کل جمله را تعیین می‌کند. در نهایت، احساسات شناسایی شده از هر جمله جمع‌آوری شده و به صورت فهرستی از احساسات مرتبط با کل پست ارائه می‌شود.



شکل 2-3 معماری ابزار تشخیص احساس

3-6-2 تحلیل پروفایل کاربر و سنجه تاثیرگذاری

در این بخش از تحقیق، روشی برای تخمین نفوذ یک کاربر در شبکه توییتر توضیح داده شده است. این روش که قبلاً تحت عنوان "دنبال‌کننده به دنبال شونده" معرفی شده، شبکه‌ای را تصور می‌کند که در آن کاربران توییتر به عنوان گره‌ها و ارتباطات بین آن‌ها به عنوان لبه‌ها محسوب می‌شوند. معیار نفوذ نباید تنها بر اساس تعداد "دنبال‌کنندگان" یک کاربر باشد، زیرا این اطلاعات می‌توانند به طور بالقوه دنبال‌کنندگانی باشند که محتوای کاربر را دریافت نمی‌کنند. در عوض، نسبت "دنبال‌کنندگان به دنبال شوندگان" (Ff ratio) می‌تواند یک شاخص مناسب‌تری باشد. معیار دیگری که می‌توان در نظر گرفت تعداد توییت‌های یک کاربر است که نشان‌دهنده نفوذ بیشتری بر شبکه مرتبط می‌باشد. علاوه بر این، فرکانس توییت‌های کاربر که نشان‌دهنده کاربرانی است که بیشتر از دیگران توییت می‌کنند نیز مورد استفاده قرار گرفته است.

علاوه بر این، برخی ویژگی‌های اضافی که تعامل بین کاربران مختلف را در توییتر اندازه‌گیری می‌کنند نیز در نظر گرفته شده است. به طور دقیق، تعداد بازتوییت‌ها و پاسخ‌ها نشان می‌دهد که آیا یک کاربر از محتوای دیگر کاربران لذت می‌برد و آن را بازتوییت و یا بحث می‌کند یا خیر. میزان علاقه‌مندی دنبال‌کنندگان یا دیگر کاربران به محتوای یک کاربر خاص می‌تواند از طریق تعداد محبوبیت‌ها و اشاره‌ها مشخص شود؛ این ویژگی‌ها نشان می‌دهند که ممکن است برخی از پست‌های کاربران برای برخی دیگر جالب‌تر باشد. به طور خاص، تعداد کلیک‌ها نشان‌دهنده تعداد دفعاتی است که کاربران یک پست خاص را مشاهده کرده‌اند و نشان‌دهنده نوعی از "پیوند" بین کاربران، بدون استفاده از اشاره‌ها، است. تمام معیارهای ذکر شده ترکیب شده‌اند تا فرمول زیر را تشکیل دهند که با ویژگی‌های پست‌ها سروکار دارد:

در کار حاضر، آخرین توییت‌های کاربر برای محاسبه معیارهای فوق مطابق با API توییتر پردازش شده‌اند. تجربیات ما نشان می‌دهد که ارزش‌های k مورد نظر بر اساس تمام ویژگی‌های ذکر شده و معیارهای ارزیابی شده است و به صورت زیر ارائه شده است:

در مورد نسبت Ff، یک نکته مهم این است که نسبت Ff در یک پایه لگاریتم ۱۰ قرار داده شده است تا از مقادیر بیش از حد جلوگیری شود. علاوه بر این، مقدار ۱ به نسبت اضافه شده است تا از معیار بودن آن به صفر در مواردی که کلیک‌ها، محبوبیت‌ها، اشاره‌ها، پاسخ‌ها یا بازتوییت‌ها صفر هستند جلوگیری شود.

اگر تعداد دنبال‌کنندگان دقیقاً برابر با تعداد دنبال شوندگان باشد، نسبت Ff (دنبال‌کنندگان به دنبال شوندگان) صفر خواهد بود. برای اجتناب از این موضوع، به نسبت کلیک‌ها، محبوبیت‌ها، اشاره‌ها، پاسخ‌ها و بازتوییت‌ها به تعداد کل توییت‌ها، یک واحد اضافه می‌شود. این افزودن یک واحد به نسبت‌های پیشنهادی به منظور جلوگیری از صفر شدن معیار در مواقعی است که مقادیر کلیک‌ها، محبوبیت‌ها، اشاره‌ها، پاسخ‌ها یا بازتوییت‌ها صفر باشند. این رویکرد به اطمینان از آنکه معیار تأثیرگذاری همیشه یک مقدار مثبت دارد و هرگز به صفر نمی‌رسد کمک می‌کند، که برای تجزیه و تحلیل دقیق‌تر تأثیر کاربران در شبکه‌های اجتماعی حیاتی است.

3-6-3 رفتار احساسی کاربر

در این بخش از تحقیق، رفتار احساسی کاربران به صورت دقیق تشریح شده است که شامل دو بخش اصلی است؛ اولین بخش به تجزیه و تحلیل وضعیت احساسی هر پست خاص می‌پردازد و دومین بخش وضعیت احساسی کلی کاربر را مشخص می‌کند.

برای بخش اول، روش تحلیل توییت‌های کاربران را در بازه زمانی سه هفته‌ای مورد بررسی قرار می‌دهد تا وضعیت احساسی هر پست را بر اساس شش احساس اساسی اکمن تعیین کند. این روش به دنبال شناسایی الگوهای احساسی در زمان‌های مختلف است و از بازه زمانی سه هفته‌ای بهره می‌برد تا تغییرات در وضعیت احساسی کاربران را به دست آورد. به علاوه، این روش از یک ابزار خاص استفاده می‌کند که توییت‌ها را به صورت احساسی تفسیر می‌کند و اگر هر توییتی پس از تجزیه و تحلیل احساسی شده باشد، وضعیت احساسی کلی کاربر بر اساس وضعیت احساسی توییت‌هایش تعیین می‌گردد.

در بخش دوم، وضعیت احساسی کلی کاربران بر اساس تفسیر احساسی هر توییت در بازه زمانی مشخص محاسبه می‌شود. این وضعیت احساسی کلی از طریق مشخص کردن اینکه آیا کاربران دارای یک پالس احساسی زنده هستند یا نه تعیین می‌گردد؛ اگر دست کم ۱۰٪ از پست‌های آن‌ها به عنوان احساسی شناخته شوند، آن‌ها دارای وضعیت حساسیتی تلقی می‌شوند، در غیر این صورت به عنوان بی‌طرف تعریف می‌گردند.

به طور کلی، وضعیت احساسی کاربران بر اساس تجزیه و تحلیل توییت‌هایشان در بازه‌ای سه هفته‌ای و با توجه به آستانه حداقل ۱۰٪ از پست‌هایشان که به عنوان احساسی شناخته می‌شوند، تعیین می‌گردد. این روش به منظور فراهم آوردن تعادل بین پست‌های احساسی و غیر احساسی کاربران در نظر گرفته شده و در موارد استثنا، مانند کاربرانی که احساسات شدیدی را نشان می‌دهند یا احساساتی را در بیش از ۵۰٪ از توییت‌هایشان به نمایش می‌گذارند، این آستانه می‌تواند تا ۱۰۰٪ افزایش یابد. این روش به ما امکان می‌دهد تا یک تصویر دقیق و کامل از رفتار و ویژگی‌های کاربران را از طریق بررسی کل فعالیت‌های پست‌های آن‌ها فراهم آوریم.

3-6-4 تشخیص جامعه

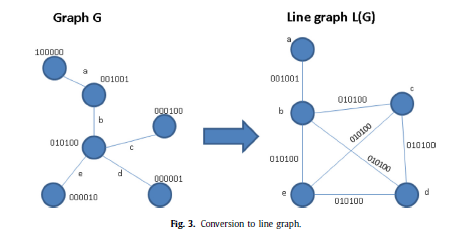
در این بخش از تحقیق، روشی برای شناسایی جوامع تأثیرگذار در شبکه کاربران توییتر شرح داده شده است. انگیزه از این کار ناشی از علاقه به شناسایی جوامع مهم در گراف کاربران توییتر است و این فرآیند با اضافه کردن یک مرحله تبدیل ترانسفورماسیون به عنوان مرحله پیش‌پردازش آغاز می‌شود. برای رسیدن به این هدف، الگوریتم بهینه‌سازی مدولاریتی که جامعه‌های تأثیرگذار را در شبکه توییتر می‌یابد، به کار گرفته شده است. این اطلاعات به صورت گراف دوگانه استخراج شده و در نظر گرفته می‌شود، که به عنوان گراف خطی شناخته می‌شود. در نهایت، الگوریتم مدولاریتی وزن‌دار برای استخراج جوامع مهم به کار برده می‌شود که با گراف اولیه مطابقت داده می‌شود.

روش به صورت جامع در مراحل زیر خلاصه شده است:

1. تبدیل به گراف خطی، که در آن گراف خطی دوگانه یک گراف اولیه است؛ دوگانه به معنای معکوس کردن یال‌ها و گره‌ها است. تبدیل متناظر در شکل ۳ نشان داده شده است. برای دقت بیشتر، کاربران (یعنی گره‌ها) توسط بردار وضعیت احساسی توییت‌هایشان بر اساس مقیاس اکمن نمایش داده می‌شوند، که در آن مقدار ۱ به معنای حضور احساس متناظر و مقدار ۰ به معنای غیاب آن است. از سوی دیگر، یال‌ها بین این گره‌ها رابطه "دنبال کردن" را نشان می‌دهند که می‌توان مشاهده کرد آن‌ها برچسب‌های متفاوتی دارند و به گره‌های مختلف متصل می‌شوند.

2. استفاده از الگوریتم تشخیص جامعه وزن‌دار، که یک روش بهینه‌سازی مدولاریتی است تا جوامع در شبکه توییتر شناسایی شوند.

3. تبدیل به گراف دوگانه اولیه، که برعکس مرحله اولیه تبدیل است.



شکل 3-3 تبدیل به گراف خطی

3-6-5 نمایش نمودار احساسی

در این بخش از تحقیق، نمایش گرافی احساسی و وزن‌دهی به تعاملات احساسی بین دو گره در شبکه مورد بحث قرار گرفته است. هر یک از شش احساس اکمن به وسیله یک یال احساسی بین دو نود محاسبه می‌شود که توسط یال متناظر نشان داده می‌شود. در ابتدا، هر موضوع توییت بر اساس نحوه بیان اطلاعات متناظر در توییت‌ها مشخص می‌شود. بنابراین، گراف پیشنهادی احساسات ارتباطی بین کاربران را که در قسمت موضوعی از ابزار کلی استفاده می‌کنند، نشان می‌دهد. بیشتر دقیق، گراف احساسی یک گراف جهت‌دار وزن‌دار G = (V, E) است که در آن گره‌ها شش احساس انسانی اساسی را که توسط اکمن تعریف شده‌اند، نمایش می‌دهند. مدل خاصی برای تعریف این وزن وجود دارد که در هر یال نشان‌دهنده شدت این دو تعامل احساسی است. وزن به صورت زیر محاسبه می‌شود:

که در آن k تعداد پست‌هایی است که بر روی یک موضوع خاص با احساس emotion1 به لحاظ احساسی نشانه‌گذاری شده‌اند. پارامتر n تعداد پست‌هایی با احساس emotion1 را نشان می‌دهد که در پاسخ به پستی با احساس emotion2 ارسال شده‌اند. پست i پست احساسی را که کاربر ارسال کرده است و پارامتر Inf\_Metric نفوذ کاربری را که پست را ارسال کرده است نشان می‌دهد. این فرمول نشان‌دهنده بخش وزن‌دار از توییت‌های کاربران است که احساس emotion1 را برای پاسخ دادن یا کامنت گذاشتن بر روی پست‌هایی که احساس emotion2 را نشان می‌دهند، حمل می‌کند.

بنابراین، مقدار نزدیک به صفر نشان می‌دهد که پست‌های نشانه‌گذاری شده با emotion2 توسط کاربرانی که پست‌هایی با emotion1 دارند، کامنت نشده‌اند. در مقابل، مقدار نزدیک به ۱۰۰ نشان می‌دهد که پست اولیه نشانه‌گذاری شده با emotion2 به طور گسترده‌ای توسط کاربرانی که emotion1 دارند، کامنت شده است.

3-7 نتیجه‌گیری

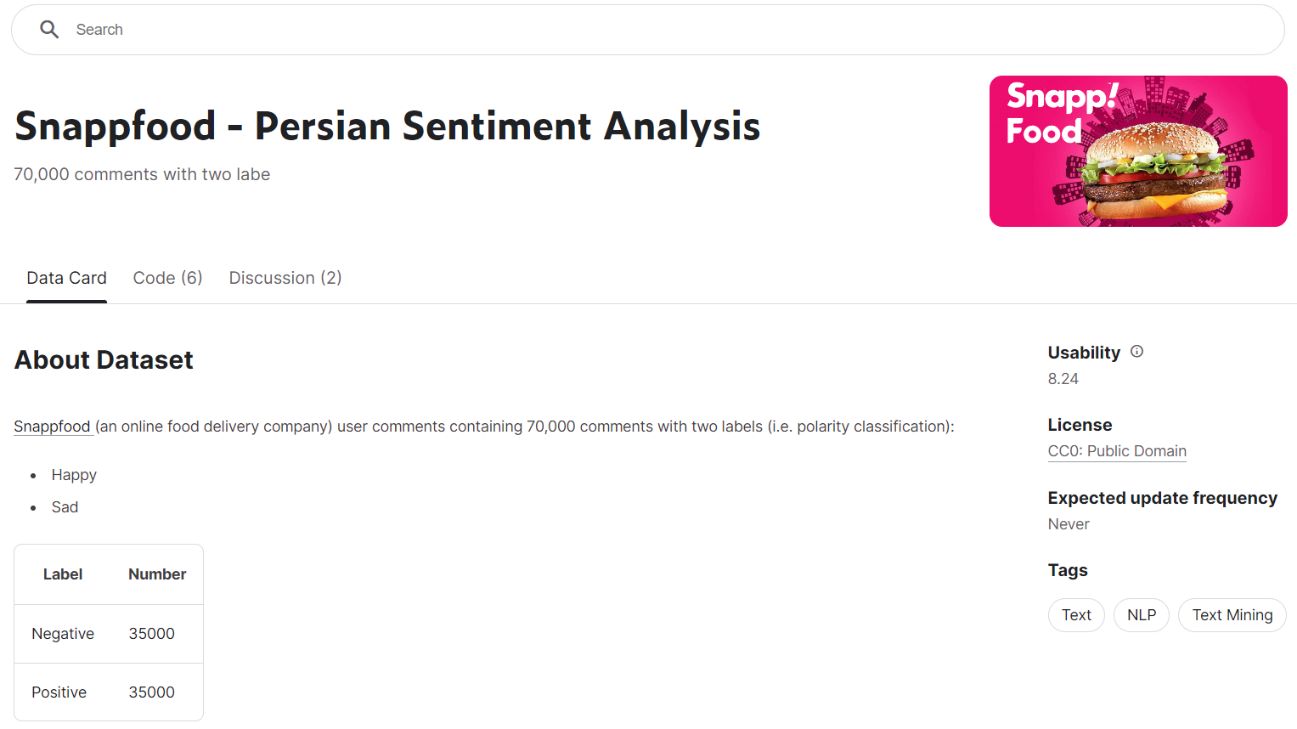
بر اساس مطالعه و تحلیل انجام شده در این تحقیق، می‌توان نتیجه گرفت که الگوریتم‌ها و رویکردهای پیشرفته معرفی شده در این مقاله، قادر به شناسایی و تجزیه و تحلیل دقیق جوامع موجود در شبکه‌های اجتماعی مانند توییتر هستند. با استفاده از مقیاس اکمن و الگوریتم‌های تشخیص جامعه وزن‌دار، ما توانستیم نه تنها جوامع متأثر از تعاملات احساسی کاربران را شناسایی کنیم، بلکه به درک عمیق‌تری از دینامیک‌های احساسی موجود در این جوامع دست یابیم. نتایج حاصل از این تحقیق نشان داد که تعاملات عاطفی کاربران می‌تواند تأثیر بسزایی در شکل‌گیری و پایداری اجتماعات داشته باشد و به ما امکان می‌دهد که با دقت بیشتری به تحلیل رفتارهای کاربران در فضای مجازی بپردازیم. این دستاوردها می‌توانند در زمینه‌های مختلفی مانند بازاریابی دیجیتال، مطالعات اجتماعی و توسعه سیاست‌های اجتماعی کاربردی و مؤثر باشند.

فصل چهارم  
«پیاده‌سازی»

4-1 مقدمه

در فصل قبل روش‌های تحلیل احساس پست و پروفایل کاربر و درنهایت تشخیص جامعه بررسی شد. کامنت یا نظر بخشی از پروفایل کاربر است که می‌تواند حاوی اطلاعات مهمی باشد از این رو در فصل یک پروژه عملی برای تحلیل احساس کامنت‌ها توسعه داده شده است.

4-2 معرفی دیتاست استفاده شده



شکل 1-4 دیتاست کامنت‌های اسنپ فود

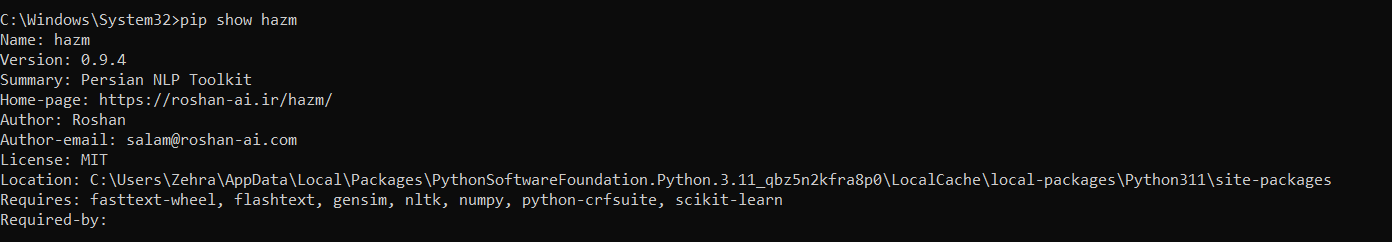
Snappfood - Persian Sentiment Analysis یک دیتاست مربوط به تحلیل احساسات به زبان فارسی است. دیتاست شامل 70,000 نظر است که با دو برچسب تقسیم شده‌اند که این برچسب‌ها نشان‌دهنده‌ی نوع احساس (مثبت یا منفی) در هر نظر هستند. دو برچسب اصلی داده‌ها Happy و Sad هستند که برای دسته‌بندی نظرات به نظرات مثبت و منفی به کار رفته‌اند. هر دو برچسب Sad و Happy دارای تعداد مساوی نظرات هستند، یعنی هر کدام 35,000 نظر، که نشان‌دهنده‌ی توازن داده‌ها در دیتاست است. این دیتاست امتیاز 8.24 را دارد که نشان دهنده خوانایی بالای آن است. همچنین این دیتاست دارای مجوز CC0: Public Domain است که است نشان می‌دهد می‌توان به صورت آزاد و بدون محدودیت از آن استفاده کرد.

4-3 کدنویسی

در این بخش، هر بخش از نوت‌بوک گوگل کولب این پروژه، با جزئیات توضیح داده شده است.

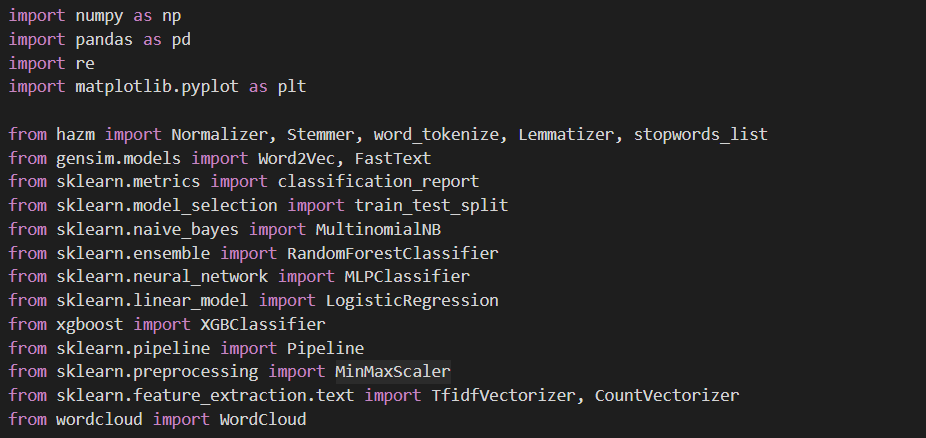
4-3-1 نصب نیازمندی‌ها و اضافه کردن کتابخانه‌های لازم

استفاده از دستور pip install hazm! برای نصب کتابخانه hazm در محیط پایتون است. Hazm یک کتابخانه متن باز برای پردازش زبان طبیعی فارسی است. این کتابخانه امکانات متنوعی را برای کار با متن‌های فارسی فراهم می‌کند، از جمله تجزیه و تحلیل دستوری، استخراج ویژگی‌ها، و تقطیع کلمات. استفاده از Hazm برای پروژه‌هایی که نیاز به تحلیل داده‌های متنی فارسی دارند بسیار مفید است.



شکل 2-4 نصب Hazm روی سیستم لوکال

برای مثال، اگر شما بخواهید احساسات موجود در نظرات فارسی را تحلیل کنید یا نیاز به تقسیم کردن جملات به کلمات و عبارات داشته باشید، `Hazm` ابزار مناسبی برای این کارها است. این کتابخانه به شما کمک می‌کند تا با استفاده از تکنیک‌های پردازش زبان طبیعی، داده‌های متنی فارسی را به راحتی مدیریت و تحلیل کنید.

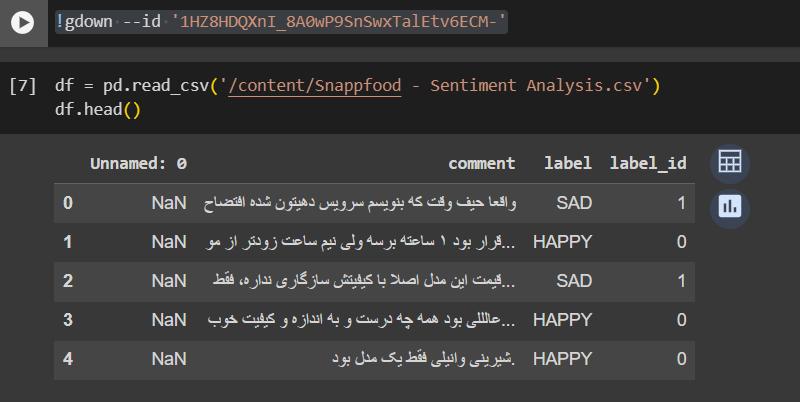


شکل 3-4 نصب و وارد کردن کتابخانه‌های ضروری

در این بخش از کد، مجموعه‌ای از کتابخانه‌های مختلف وارد شده‌اند که هر کدام کاربرد خاصی در پروژه‌های پردازش زبان طبیعی و یادگیری ماشین دارند.

1. numpy (np): کتابخانه‌ای برای محاسبات علمی در پایتون. این کتابخانه برای کار با آرایه‌های بزرگ و چندبعدی مناسب است و توابع متنوعی برای انجام عملیات ریاضی و آماری دارد.
2. pandas (pd): ابزاری برای تجزیه و تحلیل داده‌ها. این کتابخانه برای خواندن، نوشتن و مدیریت داده‌های ساختاریافته (مثل جداول) به کار می‌رود.
3. re: کتابخانه‌ای برای کار با عبارت‌های منظم (Regular Expressions). این کتابخانه امکان جستجو، ویرایش و مدیریت متن‌ها بر اساس الگوهای خاص را فراهم می‌کند.
4. matplotlib.pyplot (plt): کتابخانه‌ای برای ترسیم نمودارها و گراف‌ها. این ابزار برای نمایش داده‌ها به صورت بصری استفاده می‌شود.
5. hazm: این کتابخانه مجموعه‌ای از ابزارها برای پردازش زبان فارسی است، شامل تبدیل‌کننده‌ها (Normalizer, Stemmer, Lemmatizer) و ابزارهایی برای تقسیم جمله به کلمات (word\_tokenize) و لیستی از کلماتی که احساس ندارن و حس خاصی به جمله نمی‌دهند (stopwords\_list).
6. gensim: کتابخانه‌ای برای مدل‌سازی موضوعات و کار با مدل‌هایی مانند Word2Vec و FastText که برای تبدیل کلمات به بردارهای عددی استفاده می‌شوند.
7. scikit-learn: این کتابخانه شامل ابزارهای مختلف برای یادگیری ماشین است، از جمله توابعی برای تقسیم داده‌ها (train\_test\_split)، مدل‌های یادگیری (MultinomialNB, RandomForestClassifier, MLPClassifier, LogisticRegression, XGBClassifier)، پایپ‌لاین‌ها (Pipeline) و ابزارهای پیش‌پردازش مانند MinMaxScaler و TfidfVectorizer.
8. wordcloud: کتابخانه‌ای برای تولید تصاویر WordCloud که در آن‌ها اندازه کلمات بر اساس فراوانی ظاهر شدن‌شان در متن تعیین می‌شود.

4-3-2 وارد کردن دیتاست و خواندن محتوای آن



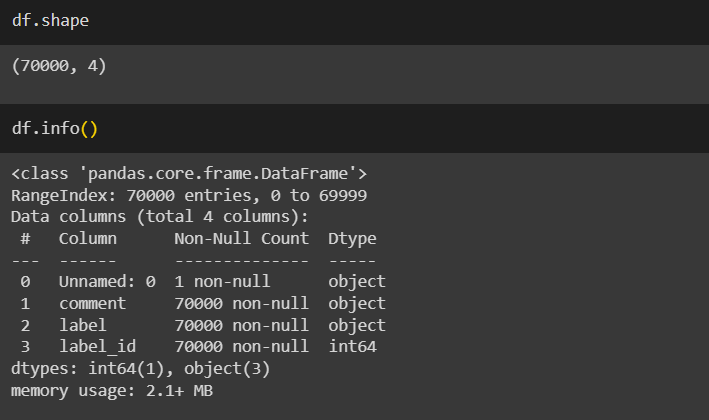
شکل 4-4 وارد کردن دیتاست

دستور gdown URL! در پایتون برای دانلود فایل از Google Drive با استفاده از شناسه مخصوص آن فایل استفاده می‌شود. gdown یک ابزار کمکی است که امکان دانلود مستقیم فایل‌ها از Google Drive را در محیط کدنویسی (مانند جوپیتر نوت‌بوک یا اسکریپت پایتون) فراهم می‌کند.

برای استفاده از این دستور، ابتدا باید gdown را نصب کنید. این می‌تواند با اجرای دستور !pip install gdown در محیط پایتون انجام شود. سپس، با استفاده از دستور gdown به همراه --id و شناسه فایل مورد نظر، می‌توانید فایل را مستقیماً دانلود کنید. این روش برای دسترسی سریع و آسان به فایل‌های ذخیره شده در Google Drive در پروژه‌های کدنویسی بسیار مفید است.

با استفاده از دستور df = pd.read\_csv(URL) ، داده‌های موجود در فایل CSV با نام Snappfood - Sentiment Analysis.csv توسط کتابخانه pandas خوانده می‌شوند. این فایل حاوی داده‌هایی است که برای تحلیل احساسات در نظرات مشتریان اسنپ‌فود استفاده می‌شوند.

با استفاده از دستور df.head()، پنج رکورد اول از دیتافریم df (که حاوی داده‌های خوانده شده از فایل CSV است) نمایش داده می‌شوند. این کار کمک می‌کند تا یک نگاه کلی به ساختار داده‌ها و نوع اطلاعاتی که در دیتافریم وجود دارند، داشته باشیم.



شکل 5-4 بررسی ساختار دیتافریم

توسط این کدها، ساختار DataFrame را بررسی می‌کنیم. دستور df.shape ابعاد DataFrame را نشان می‌دهد، که در این دیتاست نشان دهنده‌ی وجود 70,000 ردیف و 4 ستون است. دیتافریم چهار ویژگی یا مشخصه دارد و داده‌هایی برای 70,000 مورد نظرات کاربران را در خود جای داده است.

دستور df.info یک خلاصه‌ای از DataFrame را فراهم می‌کند، از جمله اطلاعات در مورد هر ستون، تعداد داده‌های غیر تهی (non-null) در هر ستون، و نوع داده‌های هر ستون (Dtype). بر اساس این خروجی، DataFrame ما دارای چهار ستون است:

- ستون اول Unnamed نام دارم.

- ستون دوم با نام comment که شامل 70,000 مقدار غیر تهی (non-null) است و از نوع object است، که معمولاً به متن یا مقادیر مخلوط در pandas اشاره دارد.

- ستون سوم با نام label دارای 70,000 مقدار غیر تهی است و از نوع object است، که برچسب‌های مربوط به حالت احساسی نظرات را نشان می‌دهد.

- ستون چهارم با نام label\_id دارای 70,000 مقدار غیر تهی و از نوع int64 است، که نشان دهنده‌ی یک شناسه عددی برای برچسب‌های احساسی است.



شکل 6-4 نمایش آمار توصیفی دیتاست

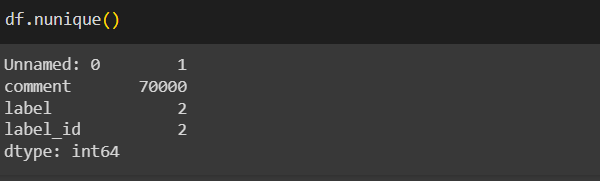
دستور df.describe(include='all') به منظور نمایش آمار توصیفی از همه ستون‌های DataFrame به کار می‌رود، از جمله ستون‌های عددی و غیرعددی.

- ستون Unnamed: 0: این ستون فقط یک مقدار غیرتهی دارد.

- ستون comment: دارای 70,000 مقدار غیرتهی است که همه منحصربه‌فرد هستند. این نشان می‌دهد که هر نظر در دیتاست منحصربه‌فرد است. ستون top یک نمونه نظر را نشان می‌دهد که بیشترین بار تکرار شده است، اما از آنجا که تکرار freq تنها 1 است، می‌توان فهمید که این نمونه نظر منحصربه‌فرد است.

- ستون label: دارای دو مقدار منحصربه‌فرد (unique: 2) است که احتمالاً نشان‌دهنده‌ی دو نوع برچسب (مثبت و منفی) است. مقدار top نشان می‌دهد که برچسب SAD بیشترین تکرار را دارد، ولی از آنجایی که تعداد freq برابر با 35,000 است، این نشان می‌دهد که برچسب‌ها به طور مساوی تقسیم شده‌اند.

- ستون label\_id: این ستون شامل اعداد صحیح است که با توجه به میانگین mean: 0.5 و انحراف معیار std: 0.500004 می‌توان گفت که اعداد صحیح برای نمایش دو برچسب مختلف به کار رفته‌اند. مقادیر کمینه min: 0.0 و بیشینه max: 1.0 این گفته را تایید می‌کنند.



شکل 7-4 بررسی مقادیر یکتای دیتاست

دستور ()df.nunique در محیط پایتون، تعداد مقادیر منحصربه‌فرد در هر ستون از DataFrame را نشان می‌دهد. این تابع می‌تواند برای ارزیابی تنوع داده‌ها و انتخاب روش‌های مناسب برای پیش‌پردازش و تحلیل داده مفید باشد.

4-3-3 پیش پردازش

انجام پیش‌پردازش و تمیزسازی داده‌ها یک مرحله مهم در فرآیند یادگیری ماشین است، زیرا کیفیت و دقت مدل‌های آموزش داده شده به طور مستقیم به کیفیت داده‌های ورودی وابسته است. پیش‌پردازش شامل حذف ویژگی‌های اضافه، حذف یا جایگزینی داده‌های گمشده، تبدیل داده‌ها به فرمت مناسب برای مدل‌سازی، نرمالسازی و استانداردسازی، و حذف نویز از داده‌ها می‌شود.

این اقدامات به حذف تاثیر داده‌های پرت یا غیر مرتبط کمک کرده و اطمینان می‌دهند که مدل بر روی الگوهای واقعی در داده‌ها تمرکز کند و نه بر روی خطاها یا استثنائات. بدون پیش‌پردازش مناسب، مدل‌های یادگیری ماشین ممکن است دچار overfitting شده یا الگوهای نادرستی را یاد بگیرند، که به نتایج نامطلوب و عملکرد ضعیف در هنگام برخورد با داده‌های جدید منجر می‌شود.

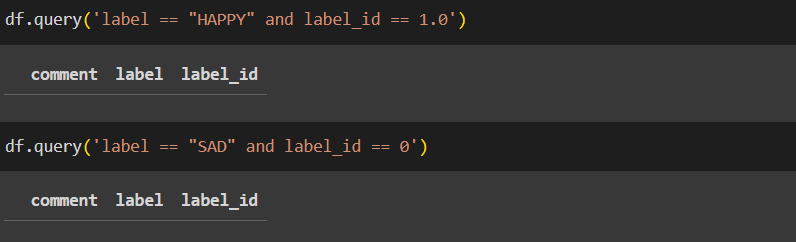


شکل 8-4 حذف ستون Unnamed

در این قسمت از پیش پردازش به دلیل مفید نبودن ستون Unnamed، سه ستون comment ، labelو label\_id از DataFrame اصلی انتخاب می‌شوند و DataFrame جدیدی با این سه ستون ایجاد می‌شود. یعنی به نوعی این سه ستون را حذف می‌کنیم.

سپس، دستور df.dropna(inplace=True) فراخوانی می‌شود که هر ردیفی را که شامل مقادیر تهی (NaN) است، حذف می‌کند. پارامتر inplace=True باعث می‌شود که تغییرات مستقیماً روی DataFrame اعمال شود بدون اینکه نیاز به اختصاص دادن تغییرات به یک متغیر جدید باشد.

در نهایت، ()df.head فراخوانی می‌شود که پنج رکورد اول از DataFrame را برای بررسی نشان می‌دهد. این کار به ما اجازه می‌دهد تا محتوای DataFrame پس از اعمال تغییرات بالا را بررسی کنیم.



شکل 9-4 فیلتر کردن سطرها

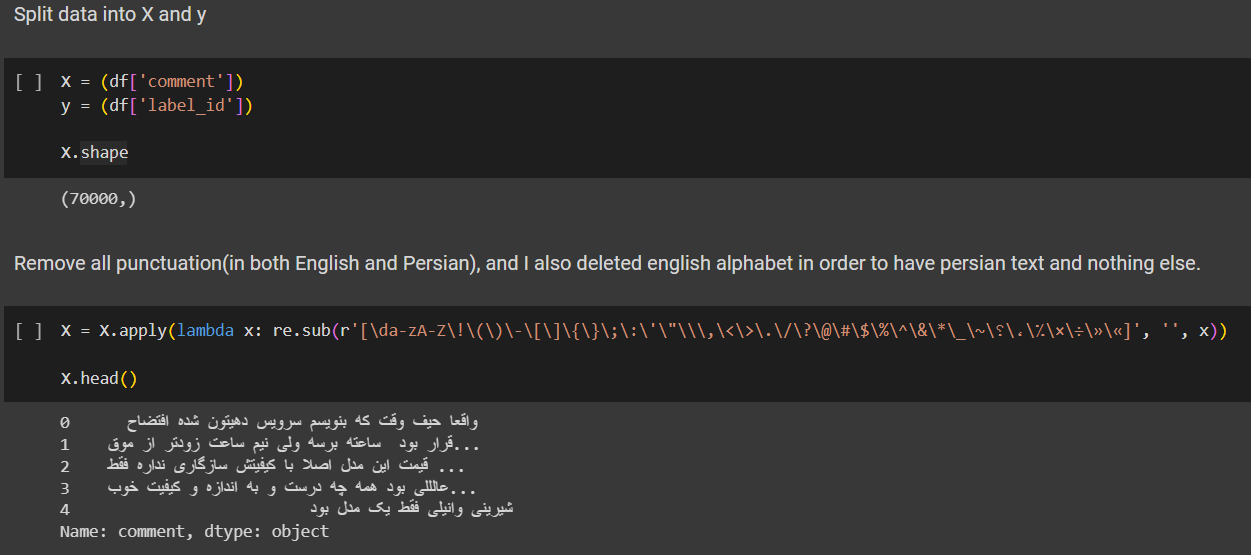
از تابع ()df.query در کتابخانه pandas پایتون برای فیلتر کردن سطرها از یک DataFrame بر اساس شرایط خاص استفاده می‌شود و در اینجا برای چک ناسازگاری‌ در برچسب‌گذاری استفاده شده است.

هدف از این کوئری‌ها بررسی صحت داده‌ها و اطمینان از این است که برچسب‌ها به درستی به label\_id مربوطه اختصاص داده شده‌اند. اگر نتایجی از اولین query برگردانده شوند، نشان دهنده ناهماهنگی در مجموعه داده است که باید برطرف شود، چرا که نظرات HAPPY نباید label\_id 1.0 داشته باشند. دومین query برای بررسی تطابق نظرات SAD با label\_id 0 است، که اگر برچسب‌ها درست باشند نباید استثناء‌ای بازگردانده شود.



شکل 10-4 تبدیل نوع داده

با این کد ستون label\_id از DataFrame به نوع داده‌ای عدد صحیح تبدیل شده و سپس پنج رکورد اول DataFrame نمایش داده می‌شوند. این تغییر نوع داده برای آماده‌سازی داده‌ها جهت استفاده در الگوریتم‌های یادگیری ماشین که نیاز به ورودی‌های عددی دارند، صورت گرفته.

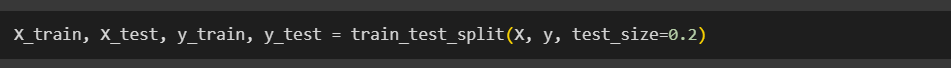


شکل 11-4 جداسازی نظرات و برچسب‌ها

در ادامه داده‌ها به دو بخش تقسیم می‌شوند. X که حاوی نظرات است و y که شامل برچسب‌ها است. این کار قبل از آموزش مدل‌های یادگیری ماشین انجام می‌شود تا مدل هنگام یادگیری برچسب‌ها را نبیند و در ادامه بیش برازش اتفاق نیافتد. X.shape نشان می‌دهد بعد از جداسازی متغیر X دارای 70,000 سطر است.

سپس، علائم نگارشی هم در زبان انگلیسی و هم در فارسی، به همراه حروف الفبای انگلیسی، از متن حذف می‌شوند. هدف این است که تنها متن فارسی باقی بماند و هیچ عنصر دیگری در داده‌ها نباشد. این کار با استفاده از یک تابع lambda و متد ()re.sub انجام می‌شود که یک عبارت منظم (regular expression) را برای شناسایی و حذف کاراکترهای مورد نظر استفاده می‌کند. حذف علائم نگارشی و حروف غیرضروری به کاهش نویز و تمرکز بر محتوای مفید متن کمک کند.

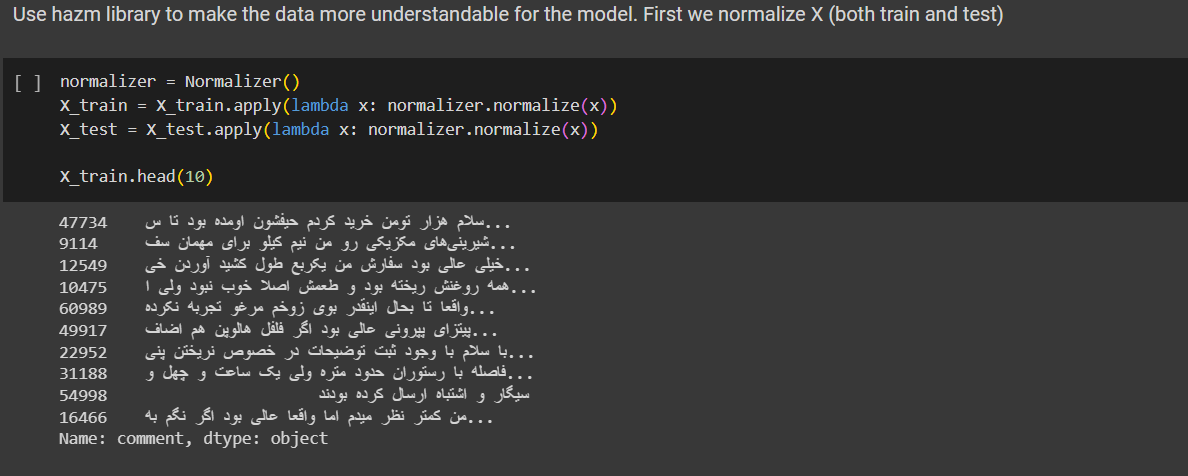
توابع لامبدا در پایتون، که اغلب به عنوان توابع ناشناس شناخته می‌شوند، یک روش مختصر برای تعریف توابع کوچک هستند. این توابع بی‌نام بوده و تنها شامل یک عبارت هستند، که نتیجه آن بلافاصله برگردانده می‌شود. توابع لامبدا معمولاً در مواردی استفاده می‌شوند که نیاز به یک تابع موقتی برای انجام عملیات‌های کوچک است، مانند اعمال یک تابع بر روی هر عنصر یک لیست یا هر سطر یا ستونی در یک DataFrame. به دلیل ساختار مختصر و قابلیت استفاده مجدد، استفاده از توابع لامبدا می‌تواند به کدنویسی تمیزتر و خوانایی بالاتر کمک کند.



شکل 12-4 تقسیم داده‌ها به دو مجموعه آموزشی و تست

تابع train\_test\_split از کتابخانه sklearn.model\_selection برای تقسیم داده‌ها به دو مجموعه آموزشی و تست استفاده می‌شود. در اینجا داده‌ها به دو بخش تقسیم شده‌اند: 80% برای آموزش و 20% برای تست، که با پارامتر test\_size=0.2 تعیین شده است.

چهار متغیر X\_train , X\_test , y\_train و y\_test به ترتیب شامل ویژگی‌های آموزشی، ویژگی‌های تست، برچسب‌های آموزشی و برچسب‌های تست هستند. این تقسیم‌بندی امکان ارزیابی عملکرد مدل یادگیری ماشین را بر روی داده‌هایی که در فرآیند آموزش دیده نشده‌اند، فراهم می‌کند و یک اقدام استاندارد در توسعه مدل‌های پیش‌بینی است.

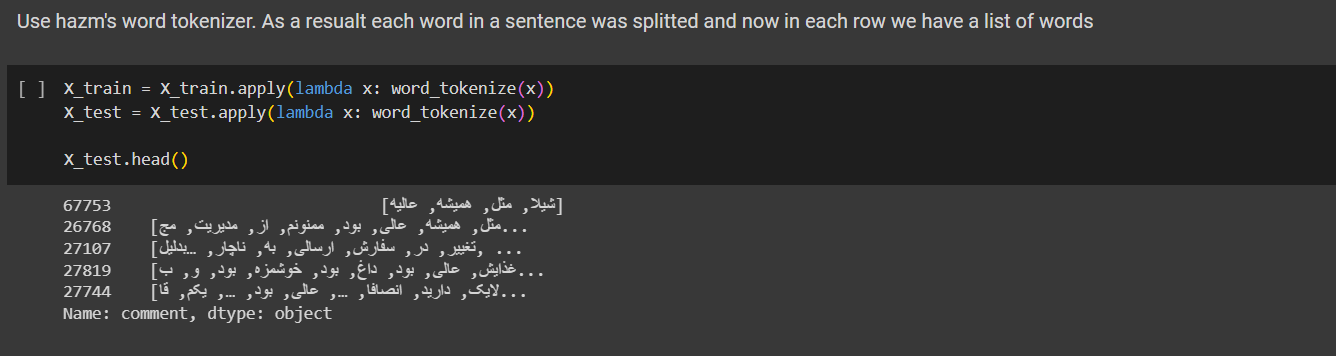


شکل 13-4 نرمال‌سازی داده

در ادامه از کتابخانه hazm برای نرمال‌سازی داده‌های متنی فارسی استفاده می‌شود. hazm یک کتابخانه پردازش زبان طبیعی برای زبان فارسی است که امکانات مختلفی مانند نرمال‌سازی، توکن‌سازی، استخراج ویژگی‌ها، و تجزیه نحوی را ارائه می‌دهد.

در این کد، ابتدا یک شیء Normalizer ایجاد می‌شود. سپس، این نرمال‌ساز برای پیش‌پردازش و نرمال‌سازی متن نظرات در هر دو مجموعه داده‌های آموزشی و تست استفاده می‌شود. عملیات نرمال‌سازی به وسیله تابع ()apply و یک تابع lambda که متد normalize را برای هر متن فراخوانی می‌کند، اجرا می‌شود. در نهایت با نمایش ده سطر اول از X\_train بعد از اعمال نرمال‌سازی، نتیجه نرمال‌سازی را بررسی می‌کنیم.

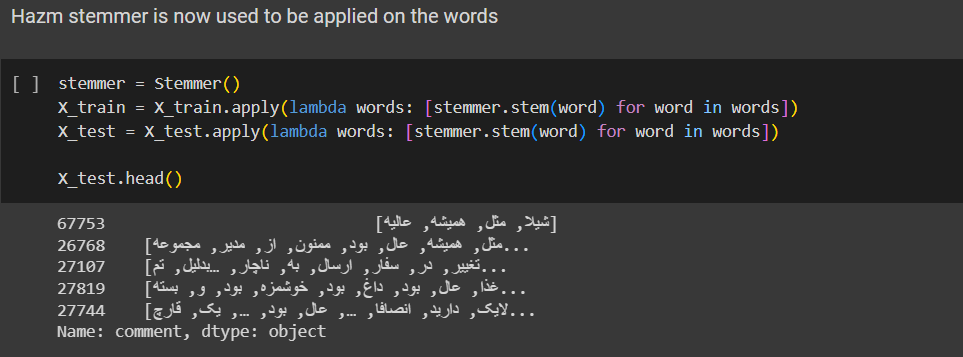
نرمال‌سازی متن ممکن است شامل مواردی مانند حذف فاصله‌های اضافی، تبدیل حروف به شکل استاندارد و یکنواخت‌سازی استفاده از حروف کوچک و بزرگ باشد. این مرحله برای کاهش تنوع در نحوه نوشتاری کلمات و بهبود کارایی مدل‌های یادگیری ماشین بسیار مهم است.



شکل 14-4 توکن‌سازی کلمات

از تابع word\_tokenize کتابخانه hazm برای توکن‌سازی کلمات در داده‌های متنی استفاده می‌شود. توکن‌سازی یکی از مراحل اولیه در پردازش زبان طبیعی (NLP) است و شامل تقسیم کردن متن به واحدهای کوچک‌تری مانند کلمات یا عبارات می‌شود. این فرآیند به مدل اجازه می‌دهد تا بتوانند اطلاعات متنی را به طور موثرتری پردازش کنند.

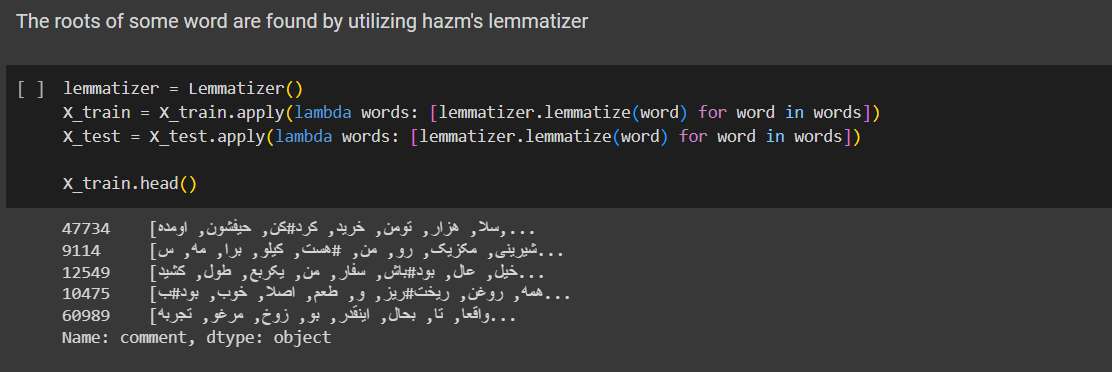
در اینجا متن‌های موجود در ستون comment از هر دو مجموعه داده‌های آموزشی و تست به وسیله تابع apply و یک تابع lambda که word\_tokenize را فراخوانی می‌کند، به لیستی از کلمات تبدیل شده‌اند. نتیجه این است که هر ردیف از ستون comment اکنون شامل یک لیست از کلمات است به جای یک رشته متنی یکپارچه.



شکل 15-4 ریشه یابی کلمات

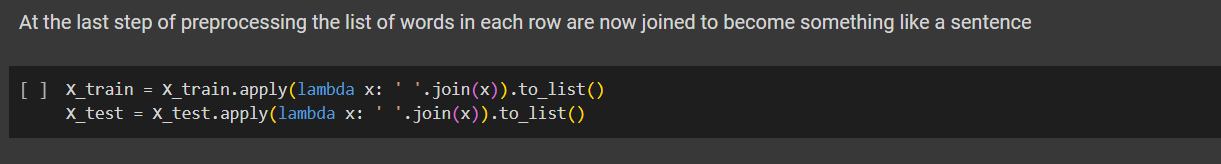
تابع Stemmer در کتابخانه hazm برای ریشه‌یابی (Stemming) فرآیندی است که در آن کلمات به شکل ریشه یا فرم اصلی خود که معمولاً نقش دستوری دارند (مانند فعل یا اسم) تبدیل می‌شوند. این کار به کاهش تنوع و تبدیل کلمات مختلف با همان ریشه به یک شکل استاندارد کمک می‌کند تا مدل‌های یادگیری ماشین بهتر بتوانند اطلاعات را یاد بگیرند و تعمیم دهند.

ابتدا یک شیء Stemmer ایجاد شده است و سپس این ریشه‌یاب برای پردازش لیست‌های کلمات در هر سطر از ستون‌های comment در X\_train و X\_test با استفاده از تابع apply و یک تابع lambda به کار رفته است. در نتیجه هر کلمه در لیست به ریشه خود تبدیل شده است.



شکل 16-4 واژه‌یابی

تابع Lemmatizer در کتابخانه hazm برای انجام واژه‌یابی (lemmatization) استفاده می‌شود. واژه‌یابی فرآیندی است که در آن کلمات به شکل دیکشنری یا واژه‌ی اصلی خود تبدیل می‌شوند، که می‌تواند به شکل جمع یا زمان‌های مختلف فعل باشد. این تکنیک اغلب نسبت به ریشه‌یابی (stemming) دقیق‌تر است زیرا در تلاش است تا واژه‌ها را به شکلی که در زبان مورد استفاده قرار می‌گیرند، تبدیل کند، و نه فقط به یک پایه خام. این مرحله پیش‌پردازش برای کاهش تنوع کلمات و تمرکز بر معنای اصلی آن‌ها به منظور بهبود عملکرد مدل‌ مفید است.

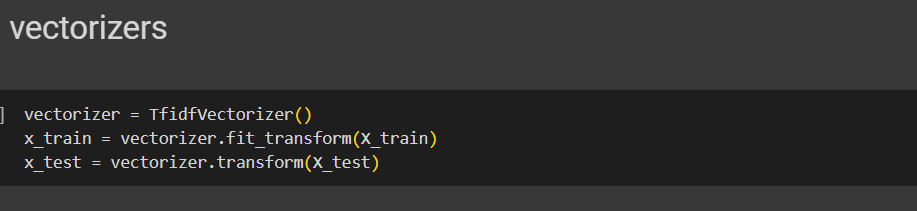


شکل 17-4 تبدیل کلمات به رشته متنی

در آخرین مرحله پیش پرداز لیست کلمات پردازش شده در مراحل قبلی را به رشته‌های متنی تبدیل می‌کنیم. با استفاده از تابع lambda هر لیست از کلمات (x) را به یک جمله با استفاده از متد ().join تبدیل کنیم. این متد کلمات را با فاصله (space character که در اینجا به عنوان آرگومان به ().join داده شده) به هم متصل می‌کند و جملاتی را ایجاد می‌کند که شبیه به جملات اصلی هستند.

این کار برای مدل‌هایی که نیاز به ورودی متنی پیوسته دارند، مانند برخی مدل‌های یادگیری ماشین برای تحلیل احساسات، مفید است. با تبدیل لیست کلمات به جملات، می‌توان ورودی‌هایی را که بیشتر شبیه به داده‌های واقعی هستند به مدل‌ها ارائه کرد.

4-3-4 بردارسازی متن



شکل 18-4 بردارسازی متن

قبل از دادن داده‌ها به مدل باید آنها را به شکل بردار درآورد. برای این کار از یک بردارساز متنی، به نام TfidfVectorizer از کتابخانه scikit-learn استفاده شده است. این بردارساز برای تبدیل متن‌های خام به ماتریسی از ویژگی‌های TF-IDF استفاده می‌شود. TF-IDF مخفف Term Frequency-Inverse Document Frequency است و به کلمات متن بر اساس اهمیت آن‌ها وزن‌دهی می‌کند.

ابتدا یک شی از TfidfVectorizer ایجاد شده است. سپس، این بردارساز با استفاده از داده‌های آموزشی آموزش دیده و سپس داده‌های آموزشی را به بردارهای ویژگی TF-IDF تبدیل کرده است. این کار با استفاده از متد fit\_transform انجام شده.

برای داده‌های تست، فقط تبدیل انجام می‌شود چون مدل باید از واژگانی که قبلاً در مرحله آموزش دیده شده‌اند استفاده کند. این کار با استفاده از متد transform انجام می‌شود.

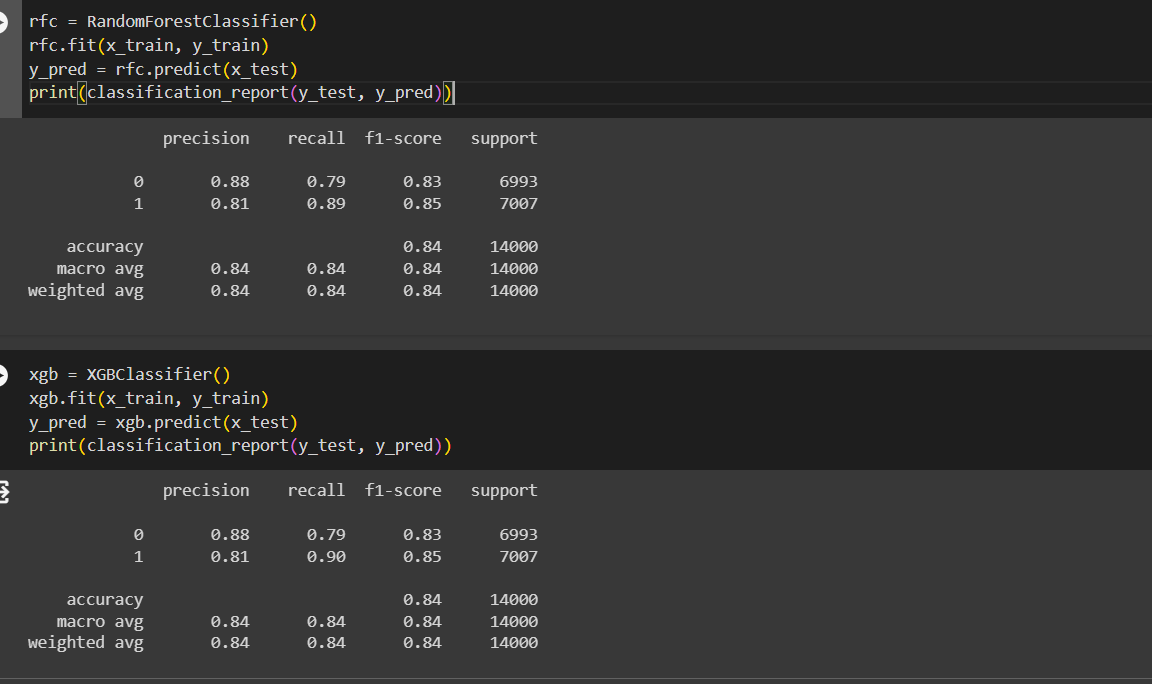
این گام ضروری است تا مدل‌های یادگیری ماشین بتوانند متن‌ها را به عنوان ورودی قابل فهمی در قالب اعداد به جای متن خام دریافت کنند. استفاده از TF-IDF به مدل کمک می‌کند تا بر کلماتی که اطلاعات بیشتری نسبت به سایر کلمات دارند تمرکز کند و از ویژگی‌هایی که به طور مکرر در تمام متون تکرار می‌شوند و اطلاعات کمتری دارند، چشم‌پوشی کند.

در یادگیری ماشین و پردازش داده‌ها، fit\_transform و transform دو متد مرتبط با تبدیل داده‌ها هستند که در کتابخانه‌های مانند scikit-learn استفاده می‌شوند. fit\_transform دو عملیات fit و transform را به طور همزمان انجام می‌دهد. متد fit الگوهای کلیدی (مانند میانگین، انحراف استاندارد، حداکثر، و غیره) را از داده‌های آموزشی یاد می‌گیرد. به عبارت دیگر، متد fit برای محاسبه و ذخیره پارامترهایی است که برای تبدیل داده‌ها لازم هستند. پس از یادگیری الگوها، transform داده‌ها را مطابق با پارامترهای یادگرفته شده تبدیل می‌کند. fit\_transform برای داده‌های آموزشی استفاده می‌شود تا هم الگوها را یاد بگیرد و هم داده‌های آموزشی را تبدیل کند.

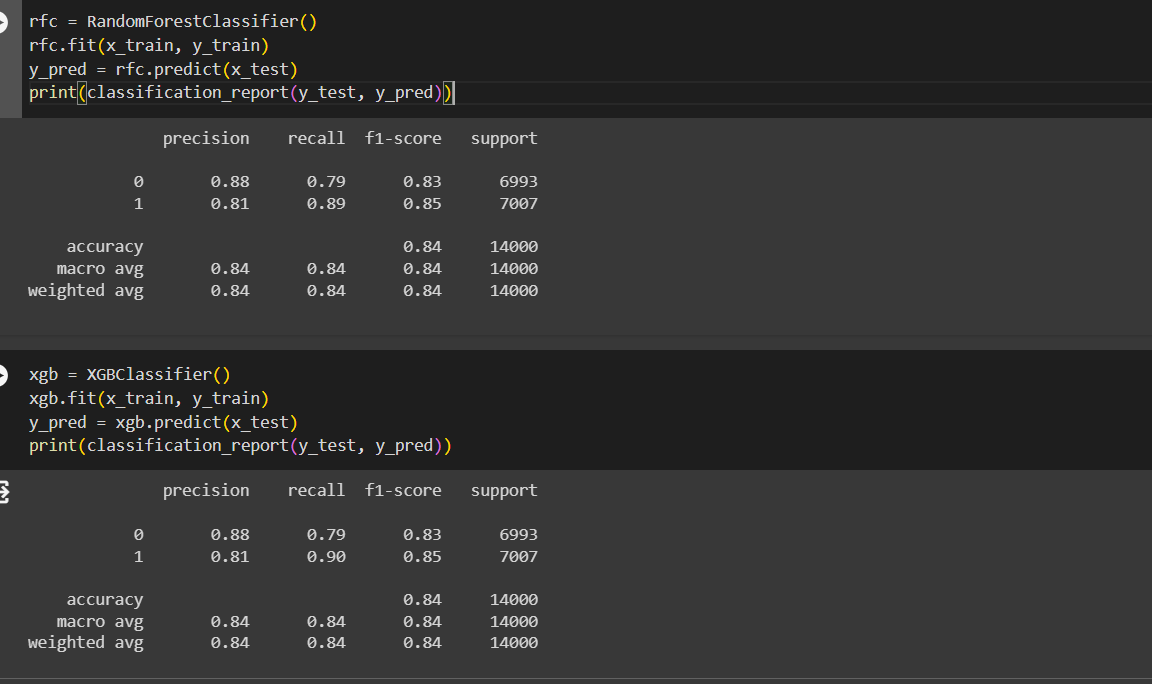
متد transform فقط عملیات تبدیل را با استفاده از پارامترهایی که از قبل توسط fit یاد گرفته شده‌اند، انجام می‌دهد. transform برای داده‌های تست (یا هر داده‌ای جدید) استفاده می‌شود و این اطمینان را می‌دهد که داده‌های تست مطابق با همان الگوها و پارامترهایی که از داده‌های آموزشی یادگرفته شده‌اند، تبدیل می‌شوند و هیچ گونه اطلاعاتی از داده‌های تست در محاسبه پارامترها استفاده نشده است.

به طور خلاصه، fit\_transform را برای آموزش مدل بر روی داده‌های آموزشی و سپس transform را برای تبدیل داده‌های تست یا جدید بدون دگرگونی پارامترهای تبدیل استفاده می‌شوند. این امر به حفظ یکپارچگی مدل و جلوگیری از نشت اطلاعات کمک می‌کند، که می‌تواند منجر به ارزیابی نادرست عملکرد مدل شود.

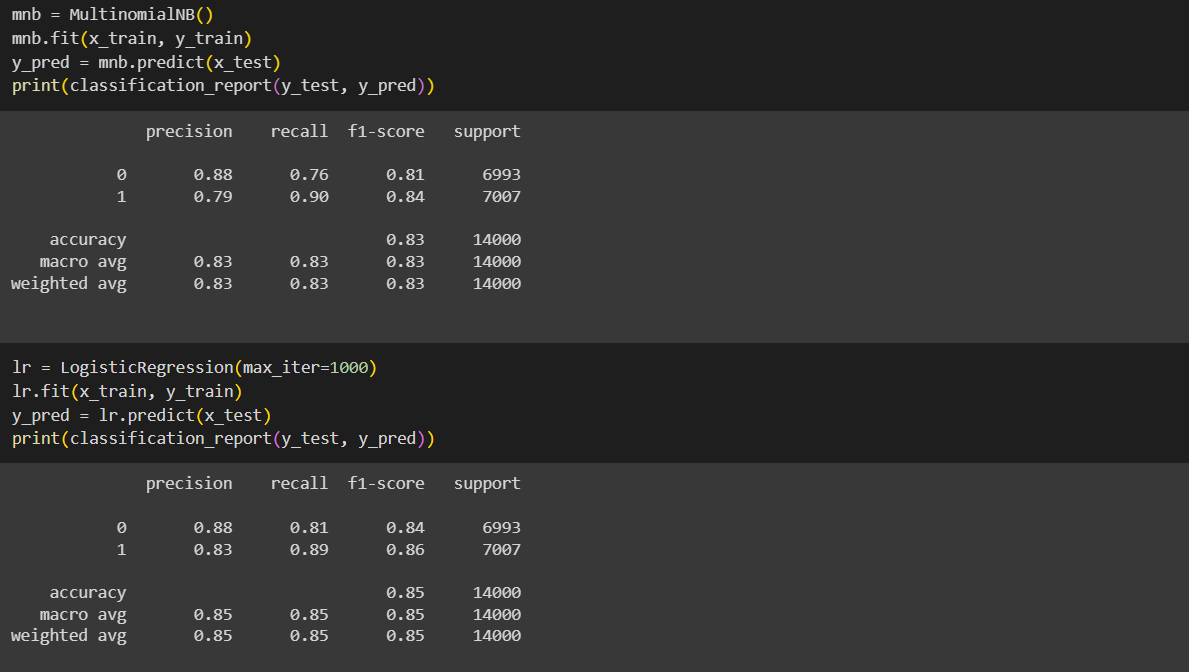
4-3-5 ساخت و ارزیابی مدل



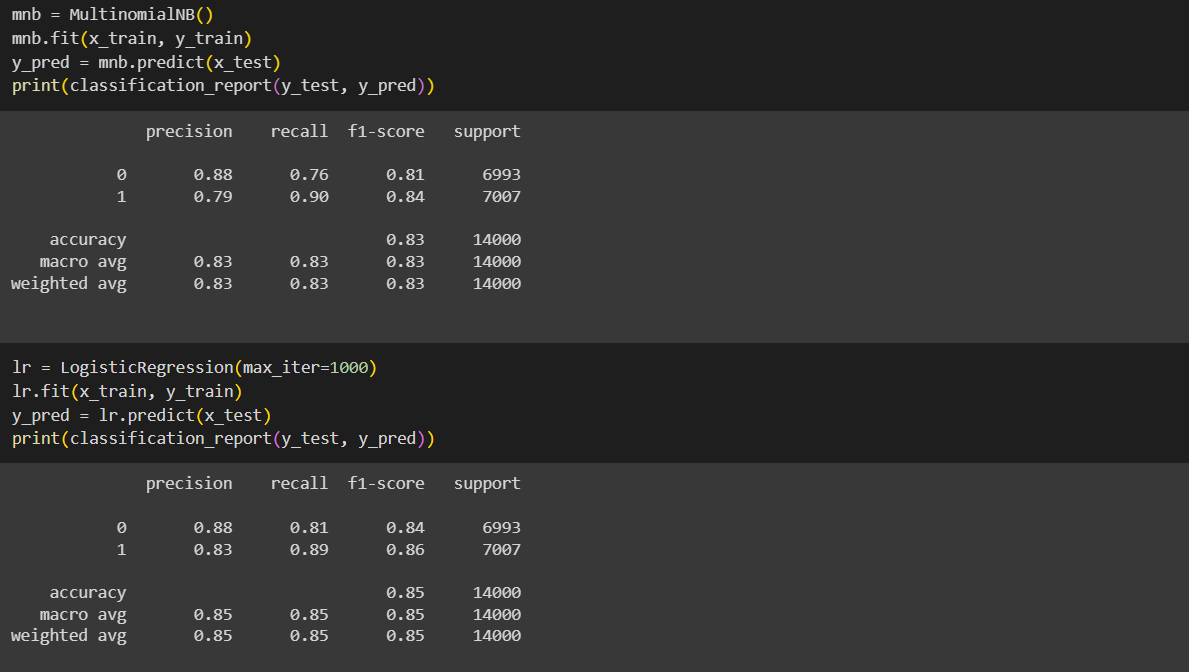
شکل 19-4 مدل های ساخته شده



شکل 20-4 مدل های ساخته شده



شکل 21-4 مدل های ساخته شده



شکل 22-4 مدل‌های ساخته شده

در این بخش از الگوریتم‌های مختلف مدل‌سازی شده و داده‌هایی که در مرحله پیش پردازش آماده شده‌ بودند، به عنوان ورودی به مدل‌ها داده شده. درنهایت گزارشی از نحوه عملکرد مدل چاپ شده که شامل اطلاعات بسیار مفیدی است و با استفاده از اطلاعات بهترین مدل انتخاب شد.

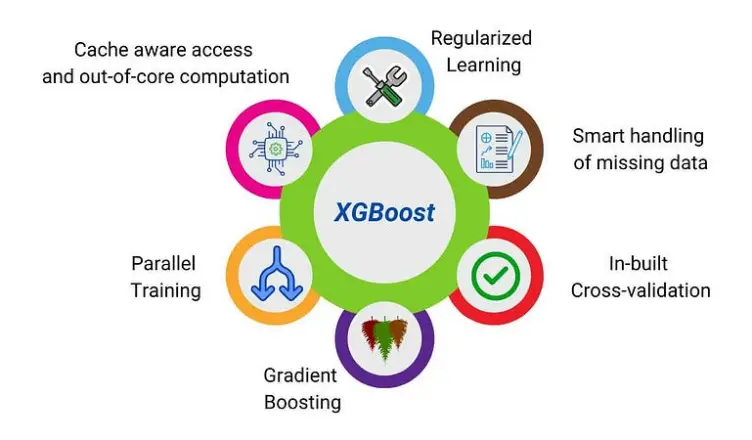
این گزارش شامل چندین متریک استاندارد است که هر کدام بخشی از عملکرد مدل را در تشخیص کلاس‌های مختلف ارزیابی می‌کنند:

1. دقت (Precision): این متریک نشان می‌دهد که از تمام پیش‌بینی‌های مثبت، چند درصد واقعاً مثبت هستند. دقت بالا نشان‌دهنده این است که تعداد کمی از مثبت‌های کاذب (false positives) وجود دارد.
2. بازیابی (Recall): بازیابی نشان می‌دهد که از تمام موارد واقعاً مثبت، چند درصد توسط مدل به درستی به عنوان مثبت شناسایی شده‌اند. بازیابی بالا نشان می‌دهد که مدل بیشتر موارد مثبت را پوشش می‌دهد و تعداد کمی از منفی‌های کاذب (false negatives) دارد.
3. نمره F1 (F1-Score): این متریک میانگین هارمونیک دقت و بازیابی است. نمره F1 در مواقعی که توازن بین دقت و بازیابی مهم است، مفید است. این متریک برای مواردی که توزیع کلاس‌ها متوازن نیست یا هزینه‌های متفاوتی برای خطاهای مختلف وجود دارد، بسیار مهم است.
4. پشتیبانی (Support): این عدد نشان می‌دهد که چه تعداد نمونه واقعی برای هر کلاس در مجموعه داده‌های تست وجود دارد. این متریک برای تعیین نمایندگی هر کلاس در داده‌های تست مفید است.
5. دقت کلی (Accuracy): این متریک نشان‌دهنده کلی‌ترین ارزیابی از عملکرد مدل است و درصد کلی نمونه‌هایی را که به درستی طبقه‌بندی شده‌اند نشان می‌دهد.
6. میانگین‌های کلی (Macro Avg, Weighted Avg): این ارزیابی‌ها میانگین‌گیری متریک‌ها را بر اساس وزن‌دهی متفاوت انجام می‌دهند. میانگین ماکرو بدون توجه به تعداد نمونه‌های هر کلاس میانگین‌گیری می‌کند، در حالی که میانگین وزن‌دار بر اساس تعداد نمونه‌ها در هر کلاس وزن‌دهی می‌کند.

این متریک‌ها با هم استفاده می‌شوند تا یک تصویر کامل و متعادل از عملکرد مدل فراهم کنند، به ویژه در مواردی که ممکن است بعضی کلاس‌ها بیش از بقیه نمایندگی شده باشند یا برخی از انواع خطاها مهم‌تر از دیگران باشند.

Random Forest یک الگوریتم یادگیری نظارتی است که به طبقه‌بندهای درخت تصمیم (Decision Trees) متکی است. این مدل با ایجاد مجموعه‌ای از درختان تصمیم‌گیری در هنگام آموزش کار می‌کند، که هر کدام روی زیرمجموعه‌ای تصادفی از داده‌ها و ویژگی‌ها آموزش می‌بینند. پیش‌بینی نهایی از طریق رأی‌گیری اکثریت در میان درختان انجام می‌شود. ویژگی‌های برجسته RFC عبارتند از کاهش بیش‌برازش (overfitting) نسبت به درختان تصمیم مجزا و قدرت بالا در مدل‌سازی مسائل پیچیده.

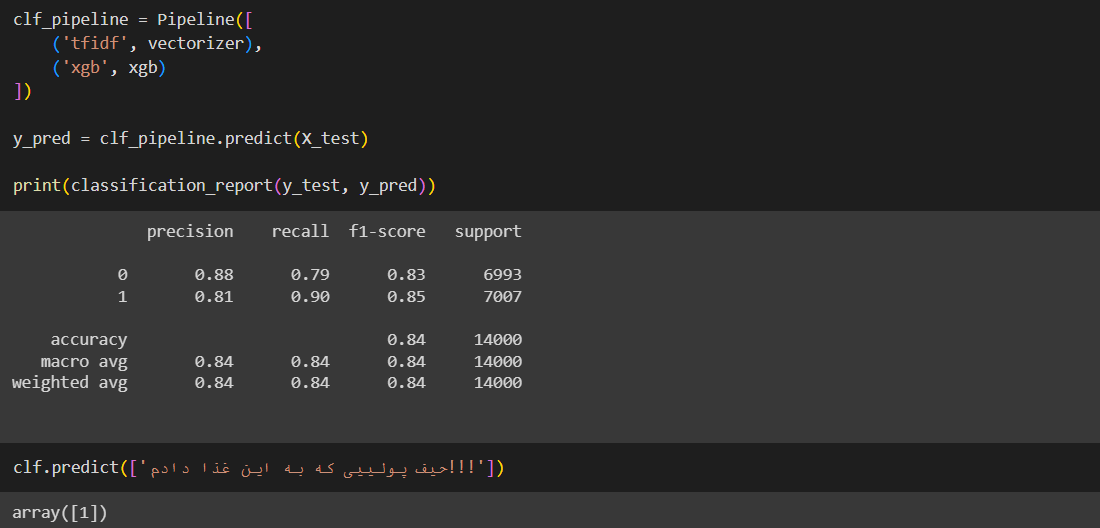
XGBoost، که مخفف eXtreme Gradient Boosting است، یک پیاده‌سازی بهینه‌شده از الگوریتم گرادیان تقویتی است. این مدل عملکرد برجسته‌ای در مسابقات داده‌کاوی به خود گرفته و به دلیل سرعت بالا، قابلیت مقیاس‌پذیری و بهبود عملکرد به شهرت رسیده است. XGB با استفاده از تکنیک‌های پیچیده مانند Pruning (هرس کردن) درختان، بهینه‌سازی موازی و روش‌های منظم‌سازی برای کاهش بیش‌برازش، قادر است مدل‌های دقیق و کارآمدی را تولید کند.



شکل 23-4 ویژگی‌های XGBoost

Multinomial Naive Bayes یک الگوریتم طبقه‌بندی احتمالاتی است که بر اساس قضیه بیز و با فرض استقلال شرطی بیز ساده بین ویژگی‌ها کار می‌کند. این مدل به ویژه در زمینه‌هایی مانند تحلیل احساسات و دسته‌بندی متن کاربرد دارد و برای داده‌هایی با توزیع‌های چندجمله‌ای مناسب است. MNB برای مسائلی که تعداد دفعات وقوع ویژگی‌ها (مانند کلمات در یک متن) مهم است، موثر عمل می‌کند.

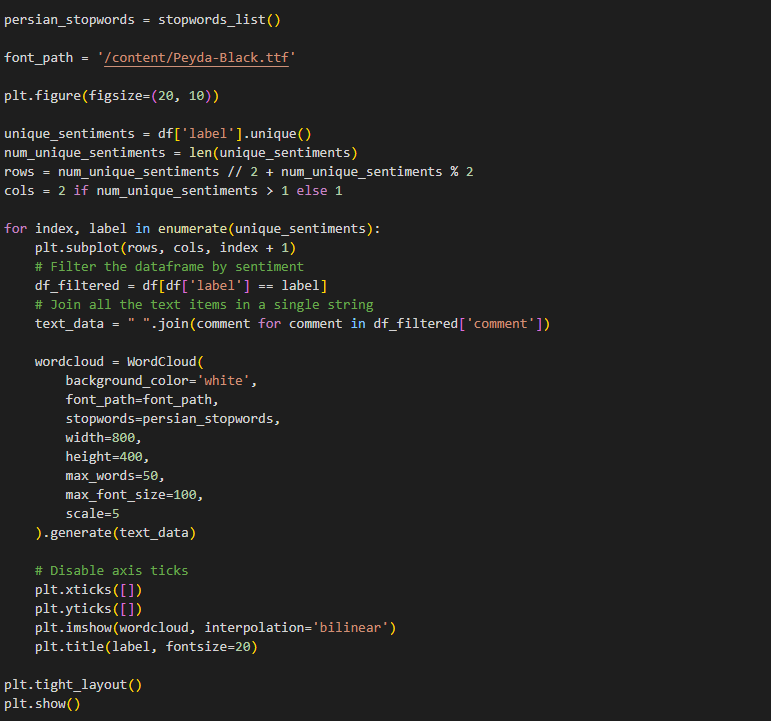
Logistic Regression یک الگوریتم طبقه‌بندی است که برای پیش‌بینی احتمال وقوع یک رویداد بر اساس یک یا چند متغیر مستقل استفاده می‌شود. این مدل یک مورد خاص از مدل‌های خطی است که از تابع لجستیک برای مدل‌سازی احتمال‌ها استفاده می‌کند و می‌تواند احتمال‌ها را به دو کلاس 0 یا 1 تقسیم کند. Logistic Regression به دلیل سادگی، شفافیت مدل و قابلیت تفسیر پذیری بالا، در بین پژوهشگران و صنعتگران محبوبیت دارد.



شکل 24-4 ساخت Pipeline

در آخرین مرحله مدلسازی با استفاده از الگوریتم XGBoost که یکی از بهترین دقت‌ها را داشت، یک Pipeline تعریف شده است. از پایپلاین‌ها به منظور ساده‌سازی کد و اطمینان از اینکه همان مراحل پیش‌پردازش و طبقه‌بندی به طور متوالی بر روی داده‌های جدید اعمال ‌شود، استفاده می‌شود. هم TfidfVectorizer و هم مدل (XGBClassifier) باید به Pipeline داده شود تا خط لوله به درستی عمل کند. در پایان، یک نمونه متن به مدل داده شده است تا کلاس آن پیش‌بینی شود و مدل هم به درستی پیش‌بینی را انجام داده است.

4-3-6 مصورسازی



شکل 25-4 کدهای مصورسازی

برای مصورسازی این پروژه از تکنیک ابر کلمات (word clouds) استفاده شده است. این تکنیک یک روش بصری برای نشان دادن کلمات بیشتر استفاده شده در یک مجموعه متن است، که کلمات بزرگ‌تر نشان‌دهنده فراوانی بیشتر آن‌ها هستند. این روش معمولاً برای درک بهتر و تحلیل بصری داده‌های متنی و بررسی تفاوت‌های احساسی بین دسته‌های مختلف استفاده می‌شود.

از کتابخانه‌های matplotlib برای نمایش و wordcloud برای ایجاد ابر کلمات استفاده شده است. برای نمایش صحیح کاراکترهای فارسی در ابر کلمات باید فونت فارسی به پروژه اضافه و آدرس آن در کد آورده شود.



شکل 26-4 نتیجه مصورسازی

4-3-7 ذخیره‌سازی مدل آموزش داده شده

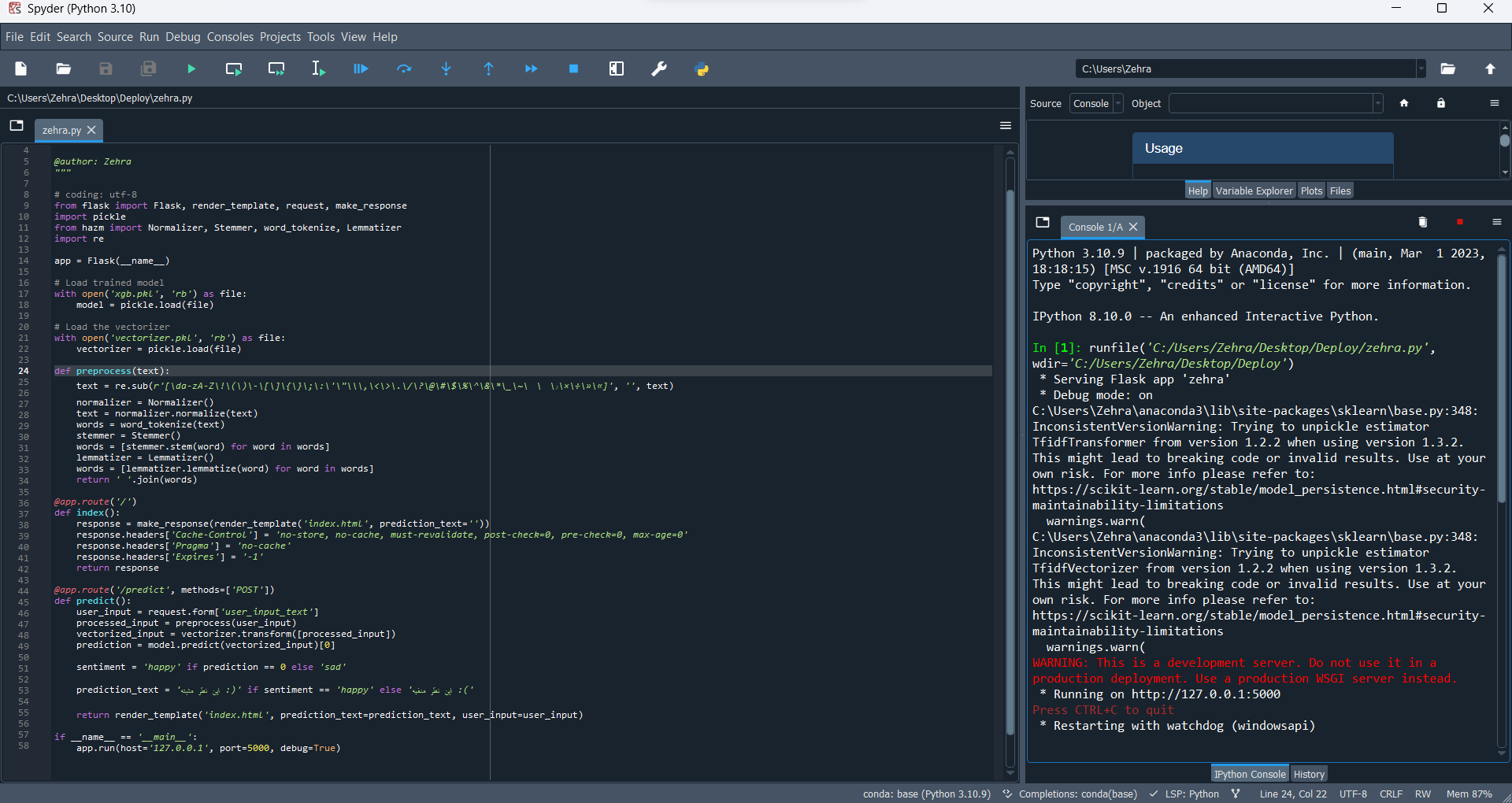


شکل 27-4 ساخت فایل pickle

درنهایت برای دخیره‌سازی مدل و Deploy کردن آن از کتابخانه pickle در پایتون استفاده شده است که اجازه می‌دهد اشیاء پایتون را به فرمت فایل باینری ذخیره کنیم. این شیوه اجازه می‌دهد که مدل‌ها و بردارسازها را بدون نیاز به آموزش مجدد در آینده، بتوان مجدداً بارگذاری و استفاده کرد. این رویکرد زمان‌بندی و منابع را صرفه‌جویی می‌کند و به سرعت بخشیدن به پروسه‌ی استقرار مدل در تولید (production) کمک می‌کند.

فایل‌های pickle به دست آمده از مدل و فرآیند بردارسازی را دانلود می‌کنیم تا در ادامه بتوانیم به کمک پکیج Flask و کدهای html – css یک وب پیج ساده که به مدل متصل (End to End Pipeline) است ایجاد کنیم.

4-3-8 استقرار مدل با پکیج Flask



شکل 28-4 کد پکیج فلسک

این کد مربوط به اپلیکیشن وب این پروژه است که با استفاده از فریم‌ورک Flask در زبان برنامه‌نویسی پایتون نوشته شده است. این اسکریپت یک مدل یادگیری ماشین آموزش‌دیده را به عنوان یک سرویس وب قابل دسترس قرار می‌دهد. در این کد، دو مدل پیکل شده که یکی مربوط به مدل XGBoost و دیگری مربوط به Vectorizer هستند، بارگذاری شده‌اند. همچنین تابعی به نام preprocess تعریف شده که وظیفه پیش‌پردازش متن ورودی را دارد. این پیش‌پردازش شامل نرمال‌سازی، توکن‌سازی، ریشه‌یابی و اشتقاق می‌شود. دو مسیر HTTP تعریف شده‌اند: یکی برای نمایش صفحه اصلی و دیگری برای پیش‌بینی احساسات متن ورودی که از طریق یک درخواست POST به سرور فرستاده می‌شود. پس از دریافت متن، این اسکریپت آن را پیش‌پردازش کرده و با استفاده از مدل یادگیری ماشین، احساس آن را پیش‌بینی می‌کند.

4-3-9 کدهای HTML, CSS, js

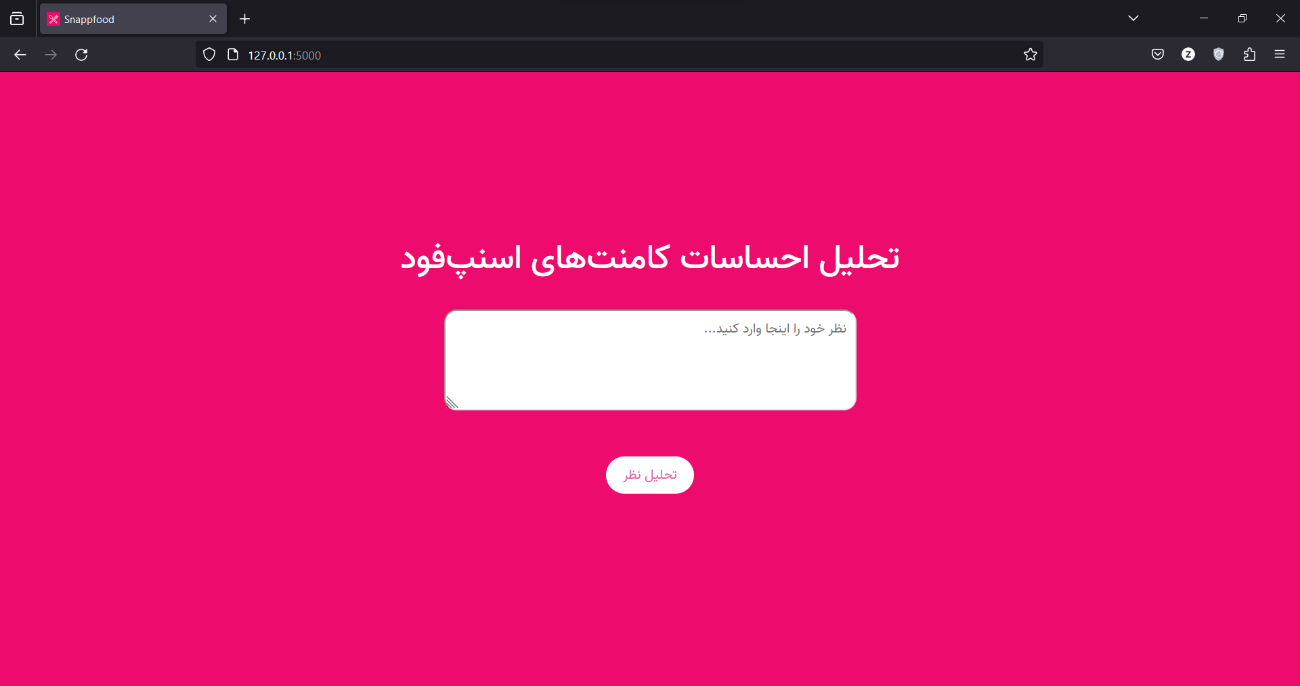


شکل 29-4 بخشی از کد رابط کاربری

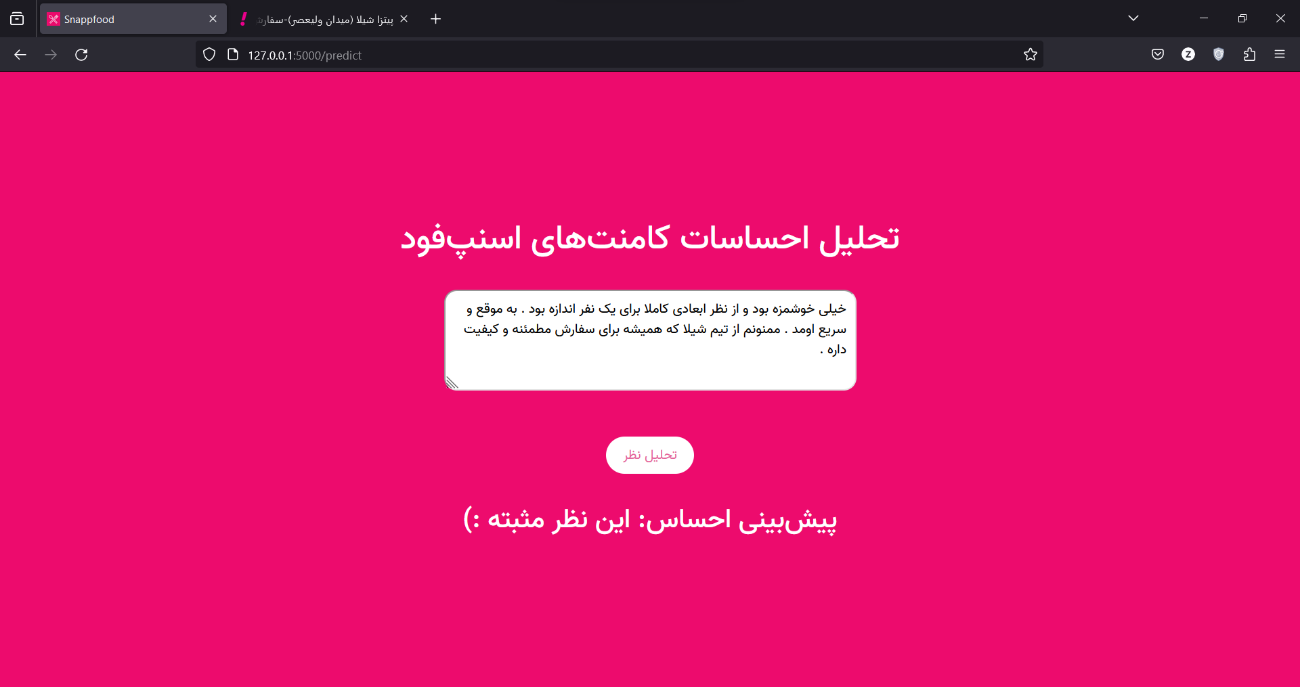
فایل index.html به عنوان رابط کاربری اصلی اپلیکیشن وب برای تحلیل احساسات متون فارسی طراحی شده است. این صفحه دارای فرمی است که کاربران می‌توانند متن خود را برای تحلیل وارد کنند. پس از ارسال متن، این درخواست به سروری که مدل یادگیری ماشین را اجرا می‌کند فرستاده می‌شود و نتایج تحلیل به صورت دینامیک روی همین صفحه نمایش داده می‌شود. طراحی صفحه با المان‌های زیبا و کاربردی، از جمله تصاویر، دکمه‌ها، فونت‌ها و رنگ‌بندی، تمرکز بر تجربه کاربری و دسترسی آسان و واکنش‌گرا را نشان می‌دهد. این فایل با استانداردهای وب و دسترسی، مطابقت با انواع دستگاه‌ها و مرورگرها را تضمین می‌کند و تجربه‌ای روان و مؤثر را برای کاربران فراهم می‌آورد.

4-4 اجرای نرم­افزار

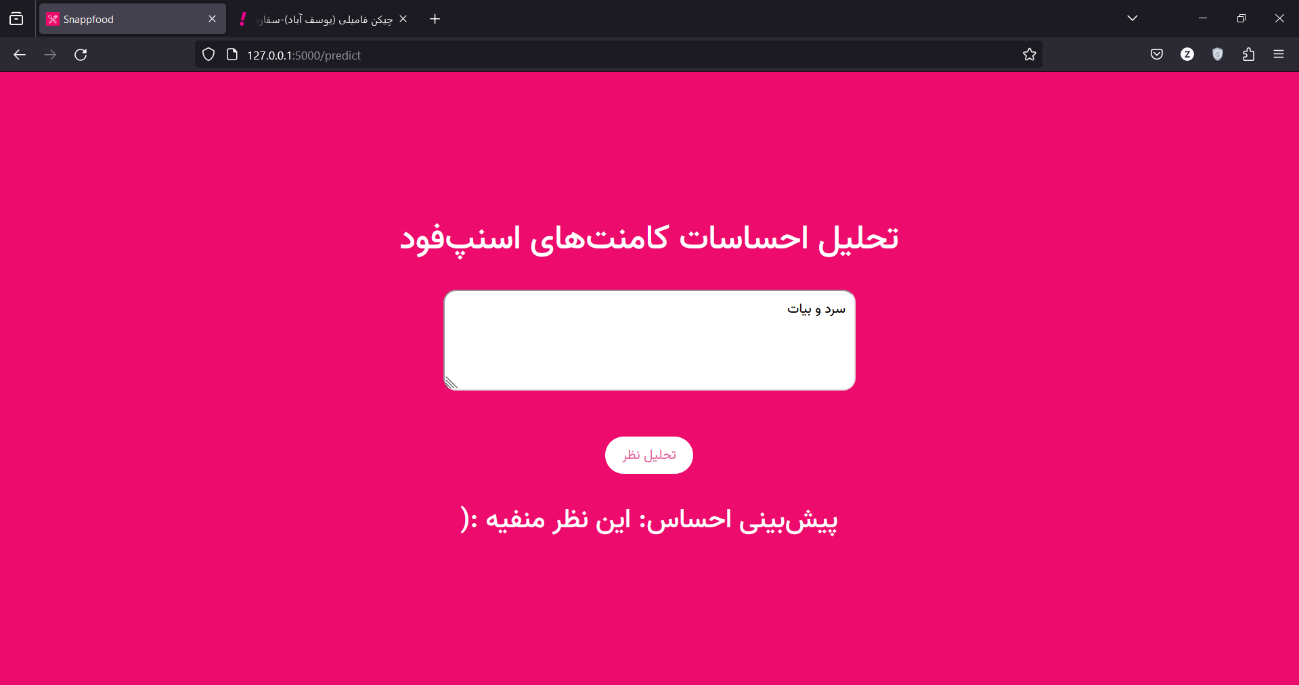
بعد از اجرای کد پایتون حاوی پکیج فلسک، بر روی localhost (127.0.0.1) و پورت 5000 وب‌سایت تحلیل نظر بالا می‌آید. در ادامه کامنت‌های واقعی از پلتفرم اسنپ‌فود کپی می‌شوند و به عنوان ورودی به مدل داده می‌شوند و بعد از پردازش مدل خروجی را مشخص می‌کند.



شکل 30-4 وب‌سایت تشخیص احساس کامنت



شکل 31-4 وارد کردن نظر مثبت و دیدن نتیجه



شکل 32-4 وارد کردن نظر منفی و دیدن نتیجه

4-5 نتیجه‌گیری

این پروژه با استفاده از ابزارهای مدرن پیاده‌سازی شده و شامل بخش‌های مختلفی است. از جمله کارهای انجام شده می‌توان به نصب کتابخانه‌ها، وارد کردن دیتاست، پیش‌پردازش و بردارسازی متن، ساخت و ارزیابی مدل‌ها، و مصورسازی داده‌ها اشاره کرد. در این پروژه، تمرکز بر روی تحلیل احساسات نظرات فارسی است و از داده‌های اسنپ‌فود استفاده شده است. برای پیاده‌سازی، ابزارهایی چون Hazm، Pandas، و Scikit-learn به کار رفته‌اند. این پروژه شامل مراحل مختلفی از جمله پیش‌پردازش داده‌ها، تقسیم داده‌ها به دو بخش آموزشی و تست، نرمال‌سازی، و استفاده از الگوریتم‌های یادگیری ماشین مانند Random Forest و XGBoost است. در نهایت، مدل‌ها با استفاده از معیارهایی مانند دقت و نمره F1 ارزیابی شده‌اند. در فصل بعد جمع‌بندی و پیشنهادات ارائه می‌شوند.

فصل پنجم  
«جمع‌بندی و پیشنهادها»

5-1 نتیجه­گیری

این پژوهش به تحلیل دقیق و عمیق احساسات در شبکه‌های اجتماعی پرداخته و با به‌کارگیری الگوریتم‌های نوین و معتبری چون XGBoost و Random Forest، توانسته است دینامیک‌های پیچیده‌ی جوامع مجازی را شناسایی کند. مطالعه بر روی داده‌های اسنپ‌فود و توییتر نشان داد که تعاملات عاطفی کاربران نقش بسزایی در شکل‌گیری و استحکام اجتماعات دارند. به‌کارگیری روش‌های پیشرفته‌ی تحلیل داده‌ها، از جمله استفاده از کتابخانه‌هایی نظیر Hazm و Scikit-learn، به تفسیر و درک بهتر رفتارهای کاربری منجر شده است. نتایج این تحقیق می‌توانند در حوزه‌هایی مانند بازاریابی دیجیتال، سیاست‌گذاری‌های اجتماعی و مطالعات رفتاری مورد استفاده قرار گیرند و زمینه‌ساز توسعه‌ی راهکارهای نوآورانه در این عرصه‌ها شوند.

5-2 پیشنهادهایی برای کارهای آتی

برای پروژه‌های آینده می‌توان بر روی توسعه و بهینه‌سازی الگوریتم‌های یادگیری ماشین در زمینه شبکه‌های عصبی عمیق کار کرد که می‌توانند در تحلیل احساسات و داده‌های شبکه‌های اجتماعی کاربردی‌تر باشند. همچنین، افزایش حجم و تنوع داده‌هایی که برای تحلیل مورد استفاده قرار می‌گیرند، می‌تواند به بهبود کیفیت و دقت مدل‌های موجود کمک کند. بررسی اثربخشی الگوریتم‌های جدید در زمینه‌های مختلفی نظیر بازاریابی دیجیتال و سیاست‌گذاری‌های اجتماعی نیز می‌تواند بینش‌های جدیدی در این حوزه‌ها ارائه دهد.

# فهرست منابع

# نمونه مقاله

1. Emotional community detection in social networks, Andreas Kanavos, Isidoros Perikos, Ioannis Hatzilygeroudis

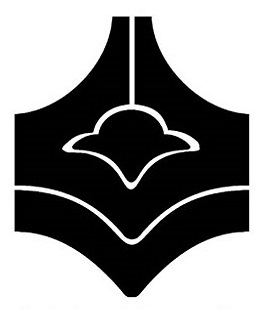
# نمونه صفحات وب

1. <https://www.kaggle.com/datasets/soheiltehranipour/snappfood-persian-sentiment-analysis>
2. <https://www.kaggle.com/code/nimanta/snappfood-sentiment-analysis>

# Abstract

This thesis aims to identify and analyze emotional communities in social networks through advanced data science and machine learning techniques. Utilizing credible datasets and innovative text processing methods, an innovative approach for analyzing user sentiments has been developed. The research findings indicate that the implemented models are capable of accurately and automatically detecting emotional states in Persian texts. The results of this study have significant implications for designing recommendation systems and analyzing user behavior on digital platforms, offering suggestions for improving existing systems and developing text analysis methods in Persian.

**Keywords:** Data Science, Machine Learning, Sentiment Detection, Community Detection



Technical and Vocational University

Dr. Shariaty- Tehran Girls Technical and Vocational College

B.S Thesis on Software Engineering

Title:

Modeling Emotional Communities and Emotion Recognition

Supervisor:

Dr. Zahra Valadkhani

By:

Zahra Rahimian

Fall 1402