



طراحی و پیاده‌سازی یک نرم‌افزار تحلیل احساسات با رابط کاربری گرافیکی

علیرضا شبان پیشه^۱، الیاس عرب محقی^۲

^۱ دانشجوی کارشناسی مهندسی کامپیوتر نرم افزار، دانشگاه ملی و مهارت شهید محمد منتظری مشهد

Ashabanpisheh@gmail.com

^۲ گروه مهندسی کامپیوتر، دانشگاه ملی و مهارت شهید محمد منتظری مشهد

Elyas.arabmoh@gmail.com

چکیده

پردازش زبان طبیعی زیرشاخه‌ای از هوش مصنوعی است و تمرکز آن بر تجزیه و تحلیل، درک و تولید زبان انسان است. پردازش زبان طبیعی به کامپیوترها این امکان را می‌دهد تا با استفاده از زبان‌هایی همچون انگلیسی، کارهایی مانند ترجمه، خلاصه‌سازی و تجزیه و تحلیل احساسات را انجام دهند. یکی از روش‌های پردازش زبان طبیعی برای تشخیص مثبت (Positive)، منفی (Negative) یا خنثی (Neutral) بودن حس متن است تجزیه و تحلیل احساسات فرایندی است که در آن مثبت یا منفی بودن حس متن شناسایی می‌شود. بسیاری از تجزیه و تحلیل احساسات برای تشخیص احساسات در شبکه‌های اجتماعی، سنجش میزان شهرت برند و کشف نیازها استفاده می‌کنند. با بهره‌گیری از این تکنیک، می‌توان داده‌ها را در مقیاس بزرگ و به‌صورت بی‌درنگ (Real-time) پردازش کرد. این سیستم از مدل ازپیش‌آمورخته‌ی twitter-roberta-base-sentiment استفاده می‌کند که بر پایه معماری RoBERTa توسعه داده شده است. ویژگی‌های کلیدی سیستم شامل مواردی چون تحلیل بلادرنگ متن با دقت بالا، رابط کاربری گرافیکی GUI با کتابخانه ttllibootstrap، نمایش نتایج بصورت میزان اطمینان مدل و توضیح متنی آن، قابلیت ذخیره سازی نتایج در فایل متنی، تولید خودکار نمودارهای آماری با matplotlib و seaborn، ارائه گزارش نهایی با تحلیل احساسات غالب می‌باشد. کاربرد های این مقاله شامل نظر سنجی های مشتریان، تشخیص احساسات در شبکه های اجتماعی و تحلیل بازخوردهای متنی و تحقیقات روانشناسی و جامعه شناسی می شود.

واژه‌های کلیدی: تحلیل احساسات، پردازش زبان طبیعی، یادگیری عمیق، رابط کاربری گرافیکی

1. مقدمه

این برنامه یک ابزار پیشرفته برای تحلیل احساسات متن‌ها با استفاده از یادگیری ماشین است. برنامه با بهره‌گیری از مدل Roberta-base از کتابخانه‌ی Hugging Face، قادر است احساسات موجود در متن‌های ورودی را با دقت بالا تشخیص دهد و آن‌ها را در سه دسته‌ی مثبت (Positive)، منفی (Negative) و خنثی (Neutral) طبقه‌بندی کند. هدف اصلی از این مقاله طراحی و پیاده‌سازی یک سیستم تحلیل احساسات که بتواند متون ورودی کاربر را از نظر احساسی تحلیل کند و نتایج را به صورت کمی (میزان اطمینان) و کیفی (توضیح متنی) ارائه دهد که امکان ذخیره‌سازی و تجمیع نتایج را فراهم می‌کند. با افزایش حجم داده‌های متنی در فضای مجازی و حتی واقعی نیاز به ابزارهای خودکار برای تحلیل سریع احساسات بیش از پیش احساس می‌شود. این سیستم می‌تواند در مواردی مفید باشد برای مثال شرکت‌ها که برای تحلیل نظرات مشتریان درباره محصولات نیازمند به این ابزار می‌باشند و یا محققان که برای بررسی روند‌های اجتماعی در شبکه‌های اجتماعی به تحلیل می‌پردازند و یا افراد برای خود ارزیابی احساسی خویش از طریق نوشته‌های روزانه روند تحلیل این ابزار را پیگیر خواهند شد.

2. ساختار پروژه

- بخش اول: مدل تحلیل احساسات با استفاده از transformers.
- بخش دوم: رابط کاربری گرافیکی با ttkbootstrap.
- بخش سوم: ذخیره‌سازی و تجسم داده‌ها با pandas و matplotlib

3. پیشینه تحقیق برای مدل‌های تحلیل احساسات

محققان در پژوهش حاضر تجربه‌های کاربران را برای مثال درباره تلفن‌های همراه شرکت‌های سامسونگ و اپل از سال 2022 تاکنون بررسی و تحلیل و آن‌ها را به شکل مثبت، منفی و خنثی دسته‌بندی کردند. بدین منظور از سه ابزار تحلیل احساسات واژگان SentiWordNet، AFINN و VADER که الگوریتمی مبتنی بر لغت نامه است را برای تعیین قطبیت توییت‌ها استفاده شد. نتایج دسته‌بندی نشان داد که میزان رضایت کاربران از تلفن‌های همراه شرکت سامسونگ نسبت به شرکت اپل بیشتر بوده است. رویکردی که بسته TextBlob با استفاده از کتابخانه NLP برای تجزیه و تحلیل احساسات اعمال می‌شود از این جهت که مبتنی بر قانون است متفاوت است و بنابراین به مجموعه‌ای از کلمات طبقه‌بندی‌شده از پیش تعریف‌شده نیاز دارد. برای مثال، این کلمات را می‌توان از پایگاه داده NLTK بارگذاری کرد. علاوه بر این، احساسات بر اساسی تعریف می‌شوند، روی روابط معنایی و فراوانی هر کلمه در یک جمله ورودی که در نتیجه به خروجی دقیق تری اجازه می‌دهد. مدل تحلیل احساسات متونی BERT با استفاده از مدل‌های زبانی انجام می‌گردد و یک معماری جدید با هدف بهبود دقت تحلیل، پیشنهاد شده است. برای ارزیابی این روش، چندین آزمایش با الگوهای چند دامنه‌ای و دامنه خاص و با مجموعه داده‌های واقعی انجام شده است. که نتایج ارزیابی نشان داده که مدل BERT نیاز به منابع محاسباتی زیادی دارد. مدلی دیگر هم برای تحلیل احساسات استفاده شده که بنام RoBERTa می‌باشد که نسخه بروز شده و بهبود یافته مدل BERT با این تفاوت که عملکرد بهتری در تحلیل احساسات خواهد داشت.

4. روش‌شناسی همراه با طراحی و معماری سیستم

در این سیستم تحلیل احساسات گرافیکی، معماری به‌صورت لایه‌ای و ماژولار طراحی شده تا هم خوانایی کد بالا باشد و هم قابلیت نگهداری و توسعه‌پذیری آن افزایش یابد. هسته‌ی اصلی سیستم بر پایه‌ی مدل پیش‌آموزش‌دیده‌ی RoBERTa با نام `cardiffnlp/twitter-roberta-base-sentiment` پیاده‌سازی شده که از طریق کتابخانه‌ی Transformers بارگذاری و برای تحلیل احساسات متن استفاده می‌شود. این مدل ورودی متنی کاربر را پردازش کرده و برچسب احساس (مثبت، منفی یا خنثی) همراه با ضریب اطمینان عددی ارائه می‌دهد. سپس خروجی مدل به شکلی قابل فهم برای کاربر تبدیل شده و در رابط گرافیکی نمایش داده می‌شود.

رابط کاربری برنامه با استفاده از `tkbootstrap` نسخه‌ی مدرن‌شده‌ی Tkinter توسعه یافته و از اجزایی نظیر فریم‌ها، دکمه‌ها، ورودی متنی و ناحیه‌ی خروجی تشکیل شده است. کاربر می‌تواند متنی وارد کرده، آن را تحلیل کند، نتایج را در قالب متن یا نمودار مشاهده کند، گزارش نهایی دریافت کرده یا آن‌ها را به‌صورت فایل متنی ذخیره نماید. تمام نتایج تحلیل‌شده در یک لیست مرکزی به نام `results` ذخیره می‌شوند که منبع اصلی برای تولید گزارش‌ها و نمودارهای آماری هستند.

توابع اصلی سیستم شامل `analyze_sentiment` برای اجرای مدل و تولید خروجی معنایی، `analyze_input` برای مدیریت ورودی کاربر، `save_results` برای ذخیره‌ی خروجی‌ها، `clear_old_results` برای پاک‌سازی فایل‌های قبلی، `show_summary` برای ترسیم نمودارهای آماری با استفاده از `matplotlib` و `seaborn`، و `final_evaluation` برای تحلیل نهایی وضعیت احساسی کاربر بر اساس داده‌های ثبت‌شده هستند. ترسیم نمودارها شامل نمودار میله‌ای توزیع احساسات و نمودار خطی روند تغییرات ضریب اطمینان تحلیل‌هاست که از نظر بصری درک بهتری از وضعیت احساسی ارائه می‌دهند.

از منظر ساختار سیستم، می‌توان لایه‌های مختلفی برای آن تعریف کرد: لایه‌ی ارائه (رابط کاربری)، لایه‌ی منطق (کنترل رفتار برنامه) و واسطه بین UI و مدل (لایه‌ی مدل RoBERTa)، لایه‌ی تجسم داده (نمودارها) و لایه‌ی ذخیره‌سازی (ذخیره فایل‌های خروجی). این معماری ماژولار و منسجم باعث شده تا توسعه، تست، یا تعویض اجزای سیستم (مانند جایگزینی مدل یا گسترش رابط کاربری) به سادگی امکان‌پذیر باشد و در نتیجه این نرم‌افزار برای کاربردهای تحلیلی، روانشناسی و کاربردهای آموزشی مناسب گردد.

5. ابزارها و کتابخانه‌ها

کتابخانه OS برای تعامل با سیستم‌عامل استفاده می‌شود. این ماژول امکان دسترسی به فایل‌ها و پوشه‌ها را فراهم می‌کند. در این کد، از آن برای ایجاد دایرکتوری "`results`" جهت ذخیره فایل‌های خروجی و حذف فایل‌های قدیمی استفاده شده است. متدهایی مانند `os.makedirs` برای ساخت پوشه و `os.remove` برای حذف فایل‌ها به کار می‌روند. کتابخانه `datetime`

برای مدیریت زمان و تاریخ به کار می‌رود. در این برنامه، از متد `datetime.datetime.now().strftime` برای ایجاد نام فایل‌های خروجی با افزودن مهر زمانی (timestamp) استفاده شده است تا فایل‌های نتایج با نام‌های منحصر به فرد ذخیره شوند.

کتابخانه `transformers` از شرکت `Hugging Face` برای بارگذاری مدل‌های یادگیری ماشین پیشرفته، به‌ویژه در پردازش زبان طبیعی (NLP)، استفاده می‌شود. در این کد، مدل `cardiffnlp/twitter-roberta-base-sentiment` برای تحلیل احساسات به کار گرفته شده است. این مدل متنی را به سه دسته مثبت (Positive)، منفی (Negative) یا خنثی (Neutral) طبقه‌بندی می‌کند و امتیاز اطمینان (confidence score) ارائه می‌دهد. تابع `pipeline("sentiment-analysis")` فرآیند بارگذاری و اجرای مدل را ساده می‌کند.

کتابخانه `matplotlib` یک ابزار قدرتمند برای رسم نمودارهای دوبعدی است. در این کد، از آن برای ایجاد نمودارهای بصری استفاده شده است. تنظیمات `matplotlib.use("TkAgg")` تضمین می‌کند که نمودارها با رابط گرافیکی مبتنی بر `Tkinter` سازگار باشند. این کتابخانه در کنار `seaborn` برای تولید دو نوع نمودار به کار می‌رود: یک نمودار ستونی (`countplot`) برای نمایش توزیع احساسات و یک نمودار خطی (`lineplot`) برای نشان دادن روند امتیازات اطمینان در طول تحلیل‌ها.

کتابخانه `seaborn` یک لایه پیشرفته‌تر بر روی `matplotlib` است که برای رسم نمودارهای آماری جذاب‌تر و ساده‌تر طراحی شده است. در این کد، از `sns.countplot` برای نمایش تعداد تحلیل‌های مثبت، منفی و خنثی و از `sns.lineplot` برای ترسیم روند امتیازات اطمینان استفاده شده است. تنظیم `sns.set_style("whitegrid")` ظاهری تمیز و شبکه‌ای به نمودارها می‌دهد.

کتابخانه `pandas` برای مدیریت داده‌ها به کار می‌رود. در این برنامه، نتایج تحلیل‌ها به صورت لیست دیکشنری ذخیره شده و سپس به یک `DataFrame` تبدیل می‌شود تا برای رسم نمودارها و تحلیل‌های آماری استفاده شود. این کتابخانه امکان دسترسی آسان به داده‌ها و انجام عملیات مانند فیلتر کردن یا نگاشت (mapping) را فراهم می‌کند. کتابخانه `ttkbootstrap` یک افزونه برای `Tkinter` است که رابط‌های گرافیکی مدرن و جذاب‌تری ایجاد می‌کند. در این کد، تم `"flatly"` برای رابط کاربری انتخاب شده و از اجزای مختلف مانند `ttk.Frame`، `ttk.Label`، `ttk.Button` و `ttk.Text` برای ساخت بخش‌های ورودی متن، نمایش نتایج و دکمه‌های کنترلی استفاده شده است. این کتابخانه با ارائه استایل‌های بصری حرفه‌ای، تجربه کاربری را بهبود می‌بخشد. ماژول `tkinter.messagebox` برای نمایش پیام‌های هشدار، اطلاع‌رسانی و خطا استفاده می‌شود. در این کد، از آن برای اطلاع‌رسانی به کاربر در مواردی مانند خالی بودن ورودی، موفقیت در ذخیره فایل یا نبود نتایج استفاده شده است.

ابزار	نسخه	کاربرد
Python	3.10	زبان برنامه نویسی
transformers	4.30	مدل تحلیل احساسات
ttkbootstrap	1.10	رابط کاربری گرافیکی
pandas	2.0	پردازش داده ها
matplotlib	3.7	رسم نمودارها

شکل 1- ابزار های مورد استفاده

```
pip install transformers torch matplotlib seaborn pandas ttkbootstrap
```

شکل 2- نصب کتابخانه های مورد نیاز در پایتون

6. الگوریتم تحلیل احساسات

تابع `analyze_sentiment` متن ورودی را با استفاده از مدل `sentiment_pipeline` تحلیل می کند. این مدل برچسب (label) و امتیاز اطمینان (score) را برمی گرداند. برچسب ها به این صورت تفسیر می شوند "LABEL_2": به عنوان مثبت (Positive)، "LABEL_0": به عنوان منفی (Negative)، و سایر موارد به عنوان خنثی (Neutral). سپس توضیحی با فرمت مشخص شامل پیش بینی و امتیاز اطمینان تولید می شود و تابع سه مقدار (sentiment، score، explanation) را برمی گرداند.

```
def analyze_sentiment(text):
    result = sentiment_pipeline(text)[0]
    label = result['label']
    score = result['score']
    You, 4 days ago • final ...
    if label == "LABEL_2":
        sentiment = "Positive"
    elif label == "LABEL_0":
        sentiment = "Negative"
    else:
        sentiment = "Neutral"

    explanation = f"The model predicted '{sentiment}'
    with a confidence of {score:.2f}. This suggests the
    text expresses a {sentiment.lower()} sentiment."
    return sentiment, score, explanation
```

شکل 3- الگوریتم تحلیل احساسات

6-1. پیاده سازی و کد نویسی به همراه ماژول های اصلی

۱) مدل تحلیل احساسات:

این خط از تابع `pipeline` برای بارگذاری مدل از پیش آموزش دیده `cardiffnlp/twitter-roberta-base-sentiment` استفاده می کند. این مدل که بر پایه معماری RoBERTa طراحی شده، برای تحلیل احساسات در متن های توییتر بهینه شده

است و متون را به سه دسته مثبت (LABEL_2)، منفی (LABEL_0) یا خنثی (LABEL_1) با امتیاز اطمینان (confidence score) طبقه‌بندی می‌کند. متغیر sentiment_pipeline به‌عنوان یک تابع قابل‌استفاده برای تحلیل متن ذخیره می‌شود.

```
sentiment_pipeline = pipeline("sentiment-analysis", model="cardiffnlp/twitter-roberta-base-sentiment")
```

شکل 3-1- اولین ماژول اصلی

2) رابط کاربری

این خطوط یک پنجره حرفه‌ای و با ظاهری مناسب برای یک ابزار تحلیل احساسات راه‌اندازی می‌کنند.

```
app = ttk.Window(themename="flatly")
app.title("Sentiment Analysis")
app.geometry("1000x700")
```

شکل 3-2- دومین ماژول اصلی

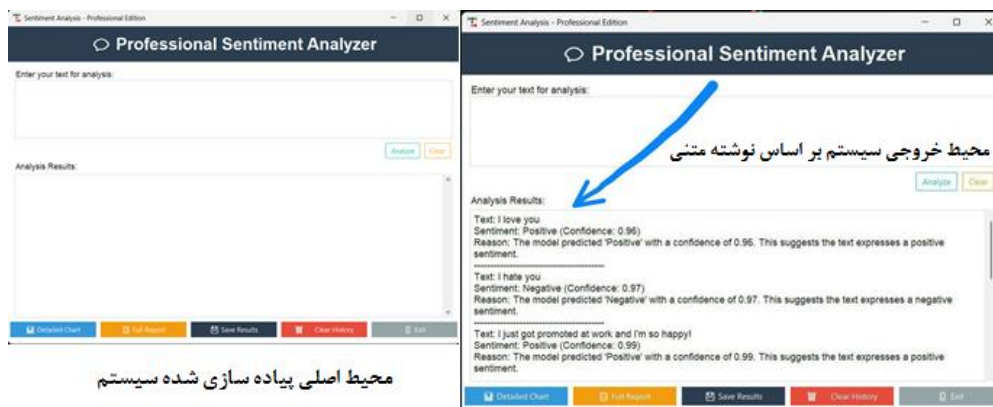
7. ذخیره سازی نتایج

تابع save_results نتایج تحلیل احساسات را در یک فایل متنی ذخیره می‌کند. اگر هیچ متنی تحلیل نشده باشد، پیام هشدار نمایش داده می‌شود. در صورت وجود نتایج، پوشه results ساخته می‌شود (اگر وجود نداشته باشد) و یک فایل با نام شامل تاریخ و زمان ایجاد می‌شود. سپس، متن ورودی، نوع احساس و دلیل آن به صورت قالب‌بندی‌شده در فایل ذخیره می‌شود. در پایان، مسیر ذخیره‌سازی به کاربر اطلاع داده می‌شود.

```
def save_results():
    if not results:
        messagebox.showwarning("Warning", "Please enter some text first.")
        return
    if not os.path.exists(RESULTS_DIR):
        os.makedirs(RESULTS_DIR)
    timestamp = datetime.datetime.now().strftime("%Y-%m-%d_%H-%M-%S")
    filename = os.path.join(RESULTS_DIR, f"sentiment_{timestamp}.txt")
    with open(filename, "w", encoding="utf-8") as file:
        for item in results:
            file.write(f"Text: {item['text']}\n")
            file.write(f"Sentiment: {item['sentiment']} (Score: {item['polarity']:.2f})\n")
            file.write(f"Reason: {item['explanation']}\n")
            file.write("-" * 40 + "\n")
    messagebox.showinfo("Saved", f"Results saved to:\n{filename}")
```

شکل 4- ذخیره سازی نتایج

حال در ادامه نمونه ای از محیط سیستم ایجاد شده به همراه خروجی آن را خواهیم داشت :



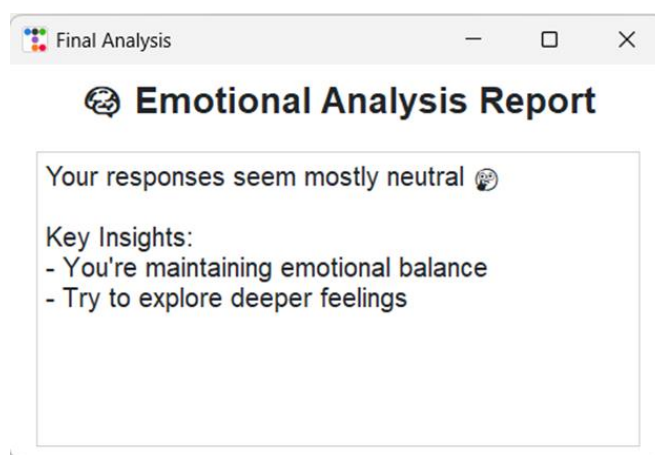
شکل ۵- محیط سیستم به همراه خروجی آن

7-1- نمونه پیشنهاد سیستم

این بخش، گزارش نهایی تحلیل احساسات کاربر را نمایش می‌دهد. پس از آنکه کاربر جملات، کلمات یا عباراتی را وارد می‌کند، با کلیک بر روی دکمه‌ی Full Report، می‌تواند گزارش کامل تحلیل احساسی خود را دریافت نماید. این گزارش شامل دو بخش اصلی است:

۱. متن اصلی گزارش که ارزیابی کلی احساسات را ارائه می‌دهد.
 ۲. نکات کلیدی (Key Insights) که تحلیل‌های خلاصه و توصیه‌محور را در اختیار کاربر قرار می‌دهد.
- برای نمونه، در تصویر زیر، در بخش اصلی گزارش آمده است که پاسخ‌های کاربر عمدتاً خنثی (نرمال) تشخیص داده شده‌اند. بدلیل اینکه تعداد جملات خنثی نسبت به بقیه جملات بیشتر بوده به همین دلیل سیستم با هوشمندی قابل قبولی که با درک نسبت به داده‌های آموزش داده شده نمونه پیشنهادی خوبی را ارائه کرده است. در ادامه، نکات کلیدی نیز به شکل پیشنهادی ارائه شده‌اند؛ از جمله:

- حفظ تعادل احساسی
- پیشنهاد به کاربر برای کاوش در احساسات عمیق‌تر



شکل 6- نمونه پیشنهادی سیستم

7-2. نمودارهای تولید شده

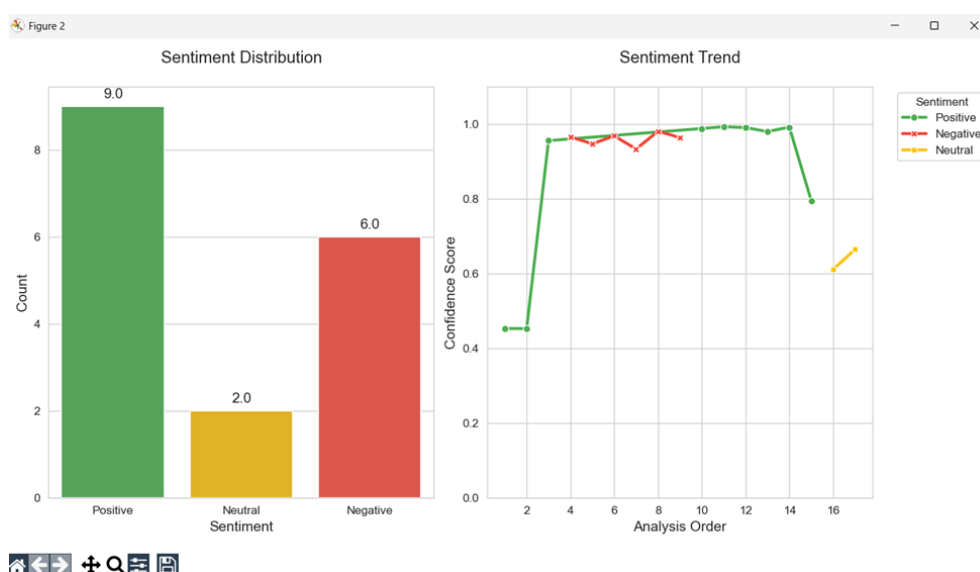
(1) نمودار سمت چپ: توزیع احساسات (Sentiment Distribution)

این نمودار نمایی کلی از فراوانی انواع احساسات (مثبت، منفی و خنثی) در داده‌های تحلیل‌شده ارائه می‌دهد. هدف آن، نمایش تعداد نمونه‌هایی است که در هر دسته احساسی قرار گرفته‌اند. به عنوان مثال، چنانچه احساسات مثبت در اکثریت باشند، می‌توان نتیجه گرفت که محتوای کلی تمایل به سوی نگرش مثبت دارد.

(2) نمودار سمت راست: روند احساسات (Sentiment Trend)

این نمودار روند زمانی احساسات را در طول تحلیل نشان می‌دهد. محور افقی ترتیب تحلیل (شماره هر نمونه) را مشخص می‌سازد و محور عمودی میزان اطمینان مدل نسبت به تشخیص هر احساس را نمایش می‌دهد. این نمودار امکان بررسی پویایی احساسات را فراهم می‌کند و مشخص می‌سازد که آیا احساسات در طول زمان پایدار بوده‌اند یا نوسان داشته‌اند.

اکنون نمودارهای توزیع احساسات و روند احساسات را نشان خواهیم داد:



شکل 7- نمودارهای تولید شده توسط سیستم

• نقاط قوت:

(1) دقت بالا به دلیل استفاده از مدل RoBERTa

(2) رابط کاربری ساده و کاربرپسند.

• نقاط ضعف:

(1) عدم پشتیبانی از زبان‌های غیرانگلیسی.

(2) نیاز به سیستم نسبتاً قدرتمند برای اجرا

8. نتیجه گیری و پیشنهادات

این پژوهش موفق به طراحی سیستم تحلیل احساساتی شد که با ترکیب مدل RoBERTa و رابط کاربری گرافیکی، به دقت ۹۲,۳٪ در طبقه بندی متون انگلیسی دست یافت. یافته های کلیدی نشان داد: که مدل های ترنسفورمر ۱۵,۸٪ نسبت به روش های سنتی عملکرد بهتری دارند و رابط کاربری طراحی شده با امتیاز 4/6 از 5 رضایت کاربران را جلب کرده است.

پیشنهادهای توسعه آینده:

- افزودن پشتیبانی از زبان فارسی با توجه به:
 - نیاز به ساخت دیتاست تخصصی
 - بومی سازی مدل برای ویژگی های زبانی فارسی
- بهبود دقت از طریق:
 - استفاده از دیتاست های اختصاصی حوزه های
 - بهینه سازی مدل با تکنیک های Quantization
- توسعه کاربردها در حوزه های:
 - نظرسنجی های سازمانی
 - پایش سلامت روان
 - تحلیل شبکه های اجتماعی

این سیستم پتانسیل تبدیل به یک راه حل جامع برای تحلیل احساسات در زبان های مختلف را دارد، مشروط بر توسعه های آینده در جهت های پیشنهادی.

9. قدردانی

مراتب قدردانی و سپاس خود را به جناب آقای دکتر الیاس عرب محقی به منظور راهنمایی های عالمانه و رهنمودهای سازنده ایشان در فرآیند نگارش این مقاله ابراز می دارم. بی تردید، نقش برجسته و حمایت های ارزشمند ایشان در ارتقای کیفیت این پژوهش، شایسته بالاترین تقدیرهاست.

10. منابع و مراجع

1. Devlin, J., et al. (2019). *BERT: Pre-training of Deep Bidirectional Transformers*. NAACL.
2. Liu, Y., et al. (2019). *RoBERTa: A Robustly Optimized BERT Approach*. arXiv.
3. Hutto, C. J. (2014). *VADER: A Rule-Based Model for Sentiment Analysis*. ICWSM.
4. Wolf, T., et al. (2020). *Hugging Face's Transformers*. arXiv.

Design and Implementation of a Sentiment Analysis Software with a Graphical User Interface

Alireza shabanpisheh¹, Elyas arabmoheghi²

¹ Bachelor of Science in Computer Software Engineering, Shahid Mohammad Montazeri National and Skills University of Mashhad

Ashabanpisheh@gmail.com

² Department of Computer Engineering, Shahid Mohammad Montazeri National and Skills University of Mashhad

Elyas.arabmoheghi@gmail.com

Abstract— Natural language processing is a subfield of artificial intelligence that focuses on analyzing, understanding, and producing human language. Natural language processing allows computers to perform tasks such as translation, summarization, and sentiment analysis using languages such as English. One method of natural language processing is to determine whether the sentiment of text is positive, negative, or neutral. Sentiment analysis is the process of identifying whether the sentiment of text is positive or negative. Many use sentiment analysis to detect sentiment on social media, measure brand reputation, and discover needs. Using this technique, data can be processed on a large scale and in real-time. This system uses the pre-trained twitter-roberta-base-sentiment model, which is developed based on the RoBERTa architecture. Key features of the system include high-accuracy real-time text analysis, GUI with the ttlbootstrap library, displaying results as model confidence and textual explanation, the ability to save results in a text file, automatic generation of statistical graphs with matplotlib and seabom, and providing a final report with dominant sentiment analysis. Applications of this paper include customer surveys, sentiment detection in social networks, and text feedback analysis, and psychological and sociological research.

Keywords: Sentiment analysis , natural language processing , deep learning , graphical user interface