```
import streamlit as st
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
import plotly.graph_objects as go

# Initialize analyzer
analyzer = SentimentIntensityAnalyzer()

# Page setup
st.set_page_config(page_title="Sentiment Analyzer", page_icon="🔍", layout="centered")
```

```
st.markdown("<h1 style='text-align: center; color: #6a11cb;'>Advanced Sentiment Analyzer</h1>", unsafe_allow_html=True)


# Text input
text_input = st.text_area("Enter your text here...", height=150)


# Helper functions
def get_sentiment_emoji(score):
    if score >= 0.05:
        return '😊'
    elif score <= -0.05:
```

```
        return '😔'
    else:
        return '😐'


def get_sentiment_color(score):
    if score >= 0.05:
        return '#96c93d'
    elif score <= -0.05:
        return '#ff4b2b'
    else:
        return '#6a11cb'


if st.button("Analyze"):
```

```
    vs = analyzer.polarity_scores(text_input)
    score = vs['compound']
    emoji = get_sentiment_emoji(score)
    color = get_sentiment_color(score)


    # Emoji and Score
    st.markdown(f"<h2 style='text-align: center; color: {color};'>{emoji}</h2>", unsafe_allow_html=True)
```

```
    st.markdown(f"### Sentiment Score: *{(score * 100):.1f}%*")


    # Bar Chart
    fig = go.Figure(data=[
        go.Bar(name='Positive', x=["Sentiment"], y=[vs['pos']], marker_color='green'),
        go.Bar(name='Neutral',  x=["Sentiment"], y=[vs['neu']], marker_color='blue'),
        go.Bar(name='Negative', x=["Sentiment"], y=[vs['neg']], marker_color='red')
    ])
```

```
    fig.update_layout(
        title="Sentiment Breakdown",
        barmode='stack',
        xaxis_title="Sentiment Type",
        yaxis_title="Proportion",
        height=400
    )
    st.plotly_chart(fig)

    # Detail view
    st.info(f"Details:\n- Positive: {vs['pos']*100:.1f}%\n- Neutral: {vs['neu']*100:.1f}%\n- Negative: {vs['neg']*100:.1f}%")
```