

LAMPIRAN

Lampiran 1: Kode Program Pengembangan Sistem.

```

1 import streamlit as st
2 from sklearn.svm import SVC
3 from imblearn.over_sampling import SMOTE
4 from gensim.models import Word2Vec
5 import pandas as pd
6 import numpy as np
7 import warnings
8 warnings.filterwarnings('ignore')
9
10 # Visualization Libraries
11 import matplotlib.pyplot as plt
12 import seaborn as sns
13
14 pd.set_option("display.max_columns", None)
15
16 import nltk
17 nltk.download('punkt')
18 from nltk.tokenize import word_tokenize
19 nltk.download('stopwords')
20 from nltk.corpus import stopwords
21 from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
22
23 st.set_page_config(
24     page_title="Sentiment Analysis Dashboard",
25     layout="wide",
26     initial_sidebar_state="auto",
27     page_icon="🌟"
28 )
29
30 root_container = st.container()
31 root_container.markdown(
32     f"""
33     <style>
34     .reportview-container .main .block-container{{
35         max-width: 100%;
36     }}
37     </style>
38     """,
39     unsafe_allow_html=True
40 )

```

```

41
42 st.sidebar.success("Made by 🇮🇩 Ditha Lozera Devi")
43
44 st.markdown(
45     """
46     <div style="background-color:#333; padding:15px;">
47     <h1 style="text-align:center;">
48     <span style="display:block;font-size:50px;color:white">
49     </span>
50     <strong>Sentiment Analysis Access by KAI</strong>
51     </h1>
52     <span style="display:block;font-size:medium;color:white;text-align:center;">
53     "Website ini menggunakan metode Word2Vec model Skip-gram dan algoritma SVM kernel RBF"
54     </span>
55     </div>
56     """,
57     unsafe_allow_html=True
58 )
59
60 def main():
61     st.title("Download Template Data")
62     st.write("Klik tautan di bawah ini untuk mengunduh template data.")
63
64     # Tampilkan tautan Google Drive
65     show_drive_link()
66
67 def show_drive_link():
68     st.markdown(f"Link ke Google Drive](https://drive.google.com/drive/folders/1tX_IDeH8jcmIEuRgQ9EHF8YVwWNSkaT0?usp=sharing)")
69
70 if __name__ == "__main__":
71     main()
72
73 st.title("Data File Reader")
74
75 uploaded_file = st.file_uploader("Upload your file here:", type=["csv","txt","xlsx","xls"])
76
77 st.warning("Warning: upload the data for give you the result!")
78 if uploaded_file is not None:
79     df = pd.read_excel(uploaded_file)
80     df.dropna(inplace=True)

```

```

C:\Users\hp\1-Sentiment-Project-main\1_0_Homepage.py (1-Sentiment-Project-main) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES
x 1_0_Homepage.py
FOLDERS
  1-Sentiment-Project-main
    myenv
    pages
    Source
  .gitignore
  / 1_0_Homepage.py
  pip scripts.txt
  predicted_sentiments.csv
  requirements.txt
1_0_Homepage.py
81 df.reset_index(drop=True, inplace=True)
82
83 st.write("Data from Excel file:")
84 st.dataframe(df['review'], use_container_width=True)
85
86 import re
87
88 def remove_emoticons(text):
89     emoticon_pattern = re.compile(u'[
90         u'\U0001F600-\U0001F64F'
91         u'\U0001F300-\U0001F3FF'
92         u'\U0001F500-\U0001F5FF'
93         u'\U0001F700-\U0001F7FF'
94         ]+', flags=re.UNICODE)
95     return emoticon_pattern.sub('', text)
96
97 def remove_special_characters(text):
98     special_char_pattern = re.compile(r'[a-zA-Z\w]')
99     return special_char_pattern.sub('', text)
100
101 df['review'] = df['review'].apply(remove_emoticons)
102 df['review'] = df['review'].apply(remove_special_characters)
103
104 stop = stopwords.words('Indonesian')
105 factory = StemmerFactory()
106 stemmer = factory.create_stemmer()
107
108 def preprocess_text(text):
109     tokens = word_tokenize(text)
110     tokens = [word for word in tokens if word not in stop]
111     stemmed_tokens = [stemmer.stem(token) for token in tokens]
112     return stemmed_tokens
113
114 df['review'] = df['review'].apply(lambda x: ' '.join(preprocess_text(x)))
115
116 # Convert review column to list of words
117 df['review'] = df['review'].apply(lambda x: word_tokenize(x))
118
119 # Train Word2Vec model with skip-gram
120 word2vec_model = Word2Vec(sentences=df['review'], size=100, window=5, min_count=1, sg=1)

```

```

C:\Users\hp\1-Sentiment-Project-main\1_0_Homepage.py (1-Sentiment-Project-main) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES
x 1_0_Homepage.py
FOLDERS
  1-Sentiment-Project-main
    myenv
    pages
    Source
  .gitignore
  / 1_0_Homepage.py
  pip scripts.txt
  predicted_sentiments.csv
  requirements.txt
1_0_Homepage.py
121
122 # Convert words to vectors
123 def vectorize_text(text):
124     vector = np.zeros(100)
125     count = 0
126     for word in text:
127         if word in word2vec_model.wv:
128             vector += word2vec_model.wv[word]
129             count += 1
130     if count != 0:
131         vector /= count
132     return vector
133
134 df['review_vector'] = df['review'].apply(vectorize_text)
135
136 # Prepare data for model
137 X = np.array(df['review_vector'].tolist())
138 y = df['sentiment']
139
140 # Apply SMOTE to the data
141 smote = SMOTE(random_state=42)
142 X_smote, y_smote = smote.fit_resample(X, y)
143
144 # Train SVM with RBF kernel
145 svm_model = SVC(kernel='rbf', random_state=42, probability=True)
146 svm_model.fit(X_smote, y_smote)
147
148 # Predict sentiments for all reviews
149 df['predicted_sentiment'] = svm_model.predict(X)
150
151 st.title("Analysis Result")
152
153 # Display the dataframe with the new sentiment column
154 st.dataframe(df[['review', 'predicted_sentiment']], use_container_width=True)
155
156 # Save the dataframe with the new column to a CSV file
157 df.to_csv("predicted_sentiments.csv", index=False)
158 with open("predicted_sentiments.csv", "rb") as file:
159     st.download_button(
160         label="Download the result CSV",

```

```

C:\Users\hp\1-Sentiment-Project-main\1_0_Homepage.py (1-Sentiment-Project-main) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
OPEN FILES
x 1_0_Homepage.py
FOLDERS
  1-Sentiment-Project-main
    myenv
    pages
    Source
  .gitignore
  / 1_0_Homepage.py
  pip scripts.txt
  predicted_sentiments.csv
  requirements.txt
1_0_Homepage.py
161 data_file,
162 file_name="predicted_sentiments.csv",
163 mime="text/csv"
164 )
165
166 # HTML for top bar
167 top_bar = """
168 <div style="background-color:#333; padding:20px">
169 <h3 style="color:white; font-size: 40px; font-weight: bold;text-align:center;">
170 <h3> Thank you for visiting, hope you like it 🍕
171 </h3>
172 </div>
173 """
174
175 st.markdown(top_bar, unsafe_allow_html=True)
176

```