wiki

June 18, 2024

[68]:
```
# Abdurrahman Bulut
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from collections import Counter
from nltk.stem import WordNetLemmatizer
```

[32]:
```
df = pd.read_csv('wiki_data.csv')
```

[35]:
```
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\abdur\AppData\Roaming\nltk_data…
[nltk_data]   Unzipping tokenizers\punkt.zip.
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\abdur\AppData\Roaming\nltk_data…
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\abdur\AppData\Roaming\nltk_data…
[nltk_data]   Package wordnet is already up-to-date!
```

[35]:
```
True
```

[3]:
```
df.head()
```

[3]:

	Unnamed: 0	text
0	1	Anovo\n\nAnovo (formerly A Novo) is a computer…
1	2	Battery indicator\n\nA battery indicator (also…
2	3	Bob Pease\n\nRobert Allen Pease (August 22, 19…
3	4	CAVNET\n\nCAVNET was a secure military forum w…
4	5	CLidar\n\nThe CLidar is a scientific instrumen…

Adım 1

[36]:
```
# Metin temizleme fonksiyonu
def clean_text(text):
    # Büyük harfleri küçük harfe çevirme
    text = text.lower()
    # Noktalama işaretlerini çıkarma
    text = re.sub(r'[^\w\s]', '', text)
    # Numerik ifadeleri çıkarma
    text = re.sub(r'\d+', '', text)
    return text
```

Adım 2

[37]:
```
# Metin temizleme fonksiyonunu tüm metinlere uygulama
df['cleaned_text'] = df['text'].apply(clean_text)
```

[6]:
```
print(df[['text', 'cleaned_text']].head())
```

```
                                                text  \
0  Anovo\n\nAnovo (formerly A Novo) is a computer…
1  Battery indicator\n\nA battery indicator (also…
2  Bob Pease\n\nRobert Allen Pease (August 22, 19…
3  CAVNET\n\nCAVNET was a secure military forum w…
4  CLidar\n\nThe CLidar is a scientific instrumen…

                                        cleaned_text
0  anovo\n\nanovo formerly a novo is a computer s…
1  battery indicator\n\na battery indicator also …
2  bob pease\n\nrobert allen pease august  â â ju…
3  cavnet\n\ncavnet was a secure military forum w…
4  clidar\n\nthe clidar is a scientific instrumen…
```

Adım 3

[38]:
```
stop_words = set(stopwords.words('english'))
```

[39]:
```
def remove_stopwords(text):
    # Metni kelimelere ayırma
    words = text.split()
    # Stopwords olmayan kelimeleri seçme
    filtered_words = [word for word in words if word not in stop_words]
    # Kelimeleri tekrar birleştirme
    return ' '.join(filtered_words)
```

[40]:
```
# Stopwords kaldırma fonksiyonunu tüm metinlere uygulama
df['processed_text'] = df['cleaned_text'].apply(remove_stopwords)
```

Adım 5

[41]:
```
# Tüm metinleri birleştir
all_text = ' '.join(df['processed_text'])
```

[43]:
```
# Kelimelerin frekansını hesapla
word_freq = Counter(all_text.split())
```

[44]:
```
# Az geçen kelimeleri tespit et
threshold = 1000  # Örneğin, 1000'den az geçen kelimeleri çıkartacağız
rare_words = {word for word, freq in word_freq.items() if freq < threshold}
```

[46]:
```
# Az geçen kelimeleri metinlerden çıkartma fonksiyonu
def remove_rare_words(text):
    words = text.split()
    filtered_words = [word for word in words if word not in rare_words]
    return ' '.join(filtered_words)
```

[47]:
```
# Az geçen kelimeleri tüm metinlerden çıkartma
df['final_text'] = df['processed_text'].apply(remove_rare_words)
```

[19]:
```
df[['text', 'final_text']].head()
```

[19]:

	text	final_text
0	Anovo\n\nAnovo (formerly A Novo) is a computer…	computer services company based france founded…
1	Battery indicator\n\nA battery indicator (also…	battery battery also known battery device info…
2	Bob Pease\n\nRobert Allen Pease (August 22, 19…	august â â june analog integrated circuit desi…
3	CAVNET\n\nCAVNET was a secure military forum w…	military became april part allows access knowl…
4	CLidar\n\nThe CLidar is a scientific instrumen…	scientific instrument used lower camera turn l…

Adım 6

[48]:
```
# Tokenize etme fonksiyonu
def tokenize_text(text):
    tokens = nltk.word_tokenize(text)
    return tokens
```

[49]:
```
# Metinleri tokenize etme
df['tokens'] = df['final_text'].apply(tokenize_text)
```

[50]:
```
df[['text', 'tokens']].head()
```

[50]:
```
                                                text  \
0  Anovo\n\nAnovo (formerly A Novo) is a computer…
1  Battery indicator\n\nA battery indicator (also…
2  Bob Pease\n\nRobert Allen Pease (August 22, 19…
3  CAVNET\n\nCAVNET was a secure military forum w…
4  CLidar\n\nThe CLidar is a scientific instrumen…

                                              tokens
0  [computer, services, company, based, france, f…
1  [battery, battery, also, known, battery, devic…
2  [august, â, â, june, analog, integrated, circu…
3  [military, became, april, part, allows, access…
4  [scientific, instrument, used, lower, camera, …
```

Adım 7

[52]:
```
lemmatizer = WordNetLemmatizer()
```

[53]:
```
# Lemmatization fonksiyonu
def lemmatize_text(tokens):
    lemmatized_tokens = [lemmatizer.lemmatize(token) for token in tokens]
    return lemmatized_tokens
```

[54]:
```
# Lemmatization işlemini uygulama
df['lemmatized_tokens'] = df['tokens'].apply(lemmatize_text)
```

[55]:
```
df[['text', 'lemmatized_tokens']].head()
```

[55]:
```
                                                text  \
0  Anovo\n\nAnovo (formerly A Novo) is a computer…
1  Battery indicator\n\nA battery indicator (also…
2  Bob Pease\n\nRobert Allen Pease (August 22, 19…
3  CAVNET\n\nCAVNET was a secure military forum w…
4  CLidar\n\nThe CLidar is a scientific instrumen…

                                     lemmatized_tokens
0  [computer, service, company, based, france, fo…
1  [battery, battery, also, known, battery, devic…
2  [august, â, â, june, analog, integrated, circu…
3  [military, became, april, part, allows, access…
4  [scientific, instrument, used, lower, camera, …
```

1 Task2

Adım 1

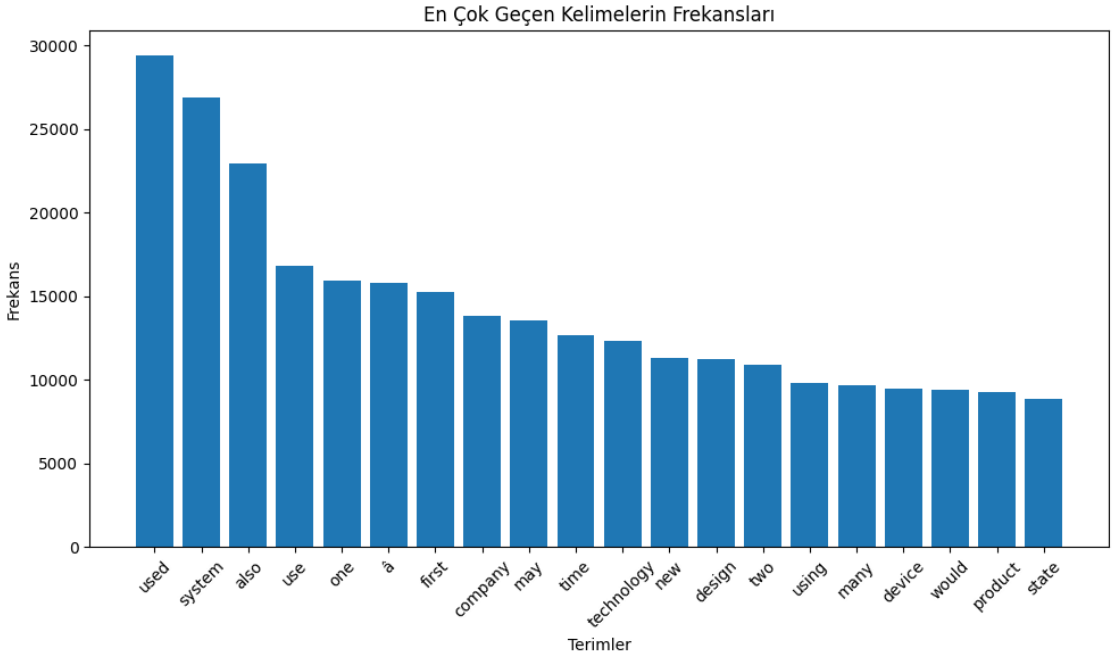
[56]:
```
# Tüm metinleri birleştir
all_text = ' '.join(df['lemmatized_tokens'].apply(lambda x: ' '.join(x)))
```

[58]:
```
# Terim frekanslarını hesapla
word_freq = Counter(all_text.split())

# Önemli olmayan kelimeleri (rare_words) çıkart
word_freq = {word: freq for word, freq in word_freq.items() if word not in␣
 ↪rare_words}
```

Adım 2

[59]:
```
import matplotlib.pyplot as plt
```

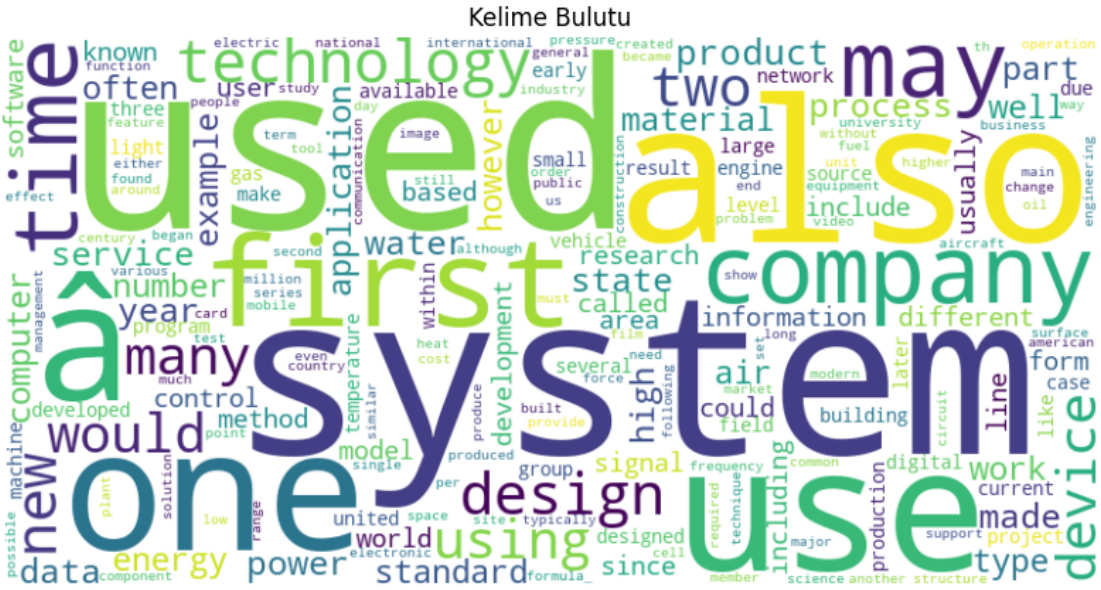
[61]:
```
# Bar plot grafiği oluşturma
plt.figure(figsize=(10, 6))
most_common_words = dict(sorted(word_freq.items(), key=lambda x: x[1],␣
 ↪reverse=True)[:20])  # En çok geçen 20 terimi seçme
plt.bar(most_common_words.keys(), most_common_words.values())
plt.xticks(rotation=45)
plt.xlabel('Terimler')
plt.ylabel('Frekans')
plt.title('En Çok Geçen Kelimelerin Frekansları')
plt.tight_layout()
plt.show()
```



Adım 3

[63]:
```
from wordcloud import WordCloud
```

[64]:
```
# WordCloud oluşturma
wordcloud = WordCloud(width=800, height=400, background_color='white').
 ↪generate_from_frequencies(word_freq)

# WordCloud'u görselleştirme
plt.figure(figsize=(10, 6))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Kelime Bulutu')
plt.show()
```



2 Task 3

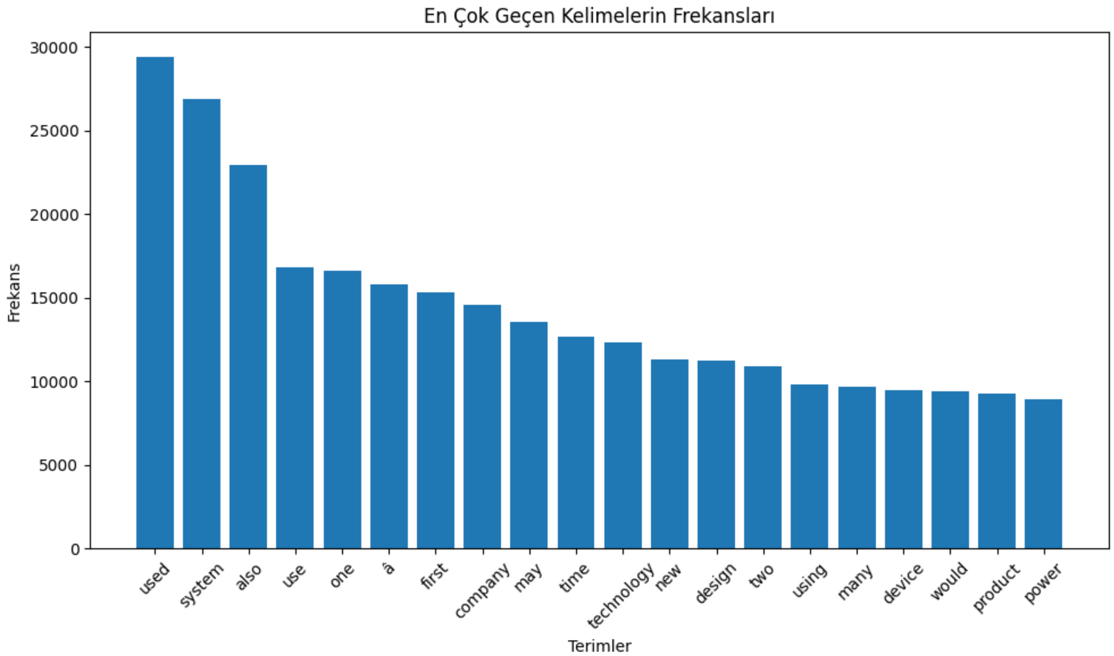
[66]:
```
def process_and_visualize_data(csv_file):
    """
    Veri setini okur, metin ön işleme işlemlerini gerçekleştirir ve␣
 ↪görselleştirme adımlarını uygular.

    Args:
    - csv_file (str): İşlenecek CSV dosyasının adı.

    Returns:
    - None: Fonksiyon çıktıyı ekrana veya dosyalara görselleştirir.
```

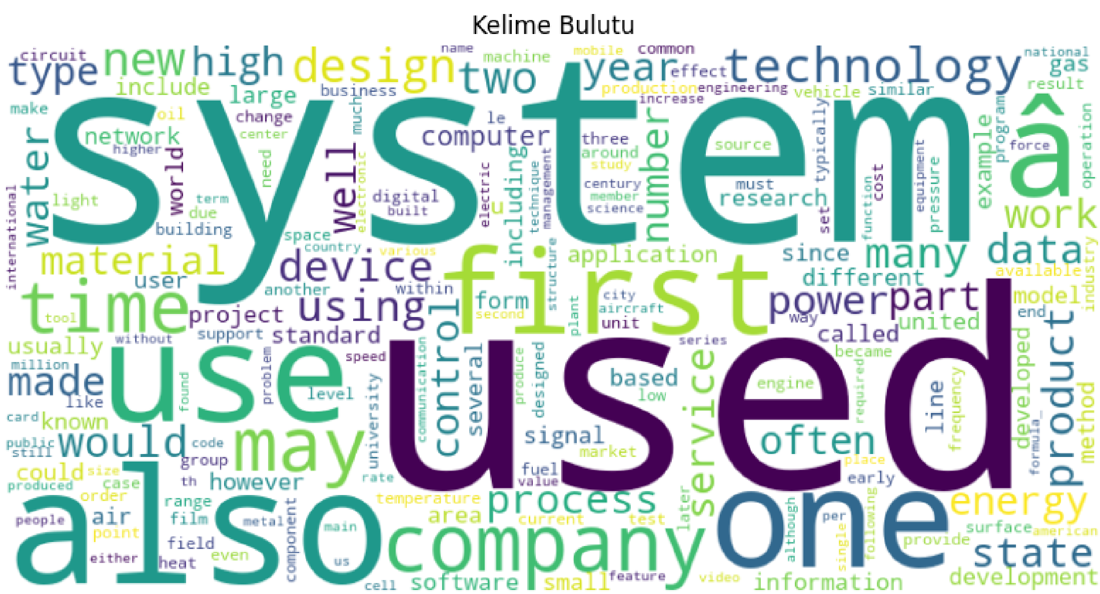
```
    """

    # Veri setini yükle
    df = pd.read_csv(csv_file)

    # Metin ön işleme fonksiyonları
    def clean_text(text):
        text = text.lower()  # Küçük harfe dönüştürme
        text = re.sub(r'[^\w\s]', '', text)  # Noktalama işaretlerini çıkarma
        text = re.sub(r'\d+', '', text)  # Numerik ifadeleri çıkarma
        return text

    def remove_stopwords(text):
        stop_words = set(stopwords.words('english'))
        words = text.split()
        filtered_words = [word for word in words if word.lower() not in␣
↪stop_words]
        return ' '.join(filtered_words)

    def tokenize_text(text):
        tokens = nltk.word_tokenize(text)
        return tokens

    def lemmatize_text(tokens):
        lemmatizer = WordNetLemmatizer()
        lemmatized_tokens = [lemmatizer.lemmatize(token) for token in tokens]
        return lemmatized_tokens

    # Metin ön işleme işlemleri
    df['cleaned_text'] = df['text'].apply(clean_text)
    df['processed_text'] = df['cleaned_text'].apply(remove_stopwords)
    df['tokens'] = df['processed_text'].apply(tokenize_text)
    df['lemmatized_tokens'] = df['tokens'].apply(lemmatize_text)

    # Terim frekanslarını hesapla
    all_text = ' '.join(df['lemmatized_tokens'].apply(lambda x: ' '.join(x)))
    word_freq = Counter(all_text.split())

    # Önemli olmayan kelimeleri (rare_words) çıkart
    threshold = 1000
    rare_words = {word for word, freq in word_freq.items() if freq < threshold}
    word_freq = {word: freq for word, freq in word_freq.items() if word not in␣
↪rare_words}

    # Frekansları görselleştirme
    print("most used 10 term:")
    print(dict(Counter(word_freq).most_common(10)))
```

```
    # Bar plot grafiği oluşturma
    plt.figure(figsize=(10, 6))
    most_common_words = dict(sorted(word_freq.items(), key=lambda x: x[1],␣
↪reverse=True)[:20])  # En çok geçen 20 terimi seçme
    plt.bar(most_common_words.keys(), most_common_words.values())
    plt.xticks(rotation=45)
    plt.xlabel('Terms')
    plt.ylabel('Frequent')
    plt.title('Most Frequent words values')
    plt.tight_layout()
    plt.show()

    # WordCloud oluşturma
    wordcloud = WordCloud(width=800, height=400, background_color='white').
↪generate_from_frequencies(word_freq)

    # WordCloud'u görselleştirme
    plt.figure(figsize=(10, 6))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis('off')
    plt.title('WordCloud')
    plt.show()
```

[67]:
```
# Fonksiyonu çağırma
process_and_visualize_data('wiki_data.csv')
```

```
En çok geçen 10 terim:
{'used': 29403, 'system': 26877, 'also': 22951, 'use': 16820, 'one': 16639, 'â':
15769, 'first': 15304, 'company': 14600, 'may': 13571, 'time': 12703}
```


En Çok Geçen Kelimelerin Frekansları
Frekans
Terimler
0
5000
10000
15000
20000
25000
30000
used
system
also
use
one
â
first
company
may
time
technology
new
design
two
using
many
device
would
product
power


Kelime Bulutu
system
used
also
use
first
one
â
company
may
time
technology
design
new
high
two
year
type
device
data
many
power
part
process
energy
state
material
water
well
number
service
control
product
often
work
would
made
using
computer
network
research
application
different
information
software
development