

NAME: V.VAMSHI  
HALL TICKET NUMBER: 2403A52016

```
import gensim.downloader as api
import numpy as np
import matplotlib.pyplot as plt
from sklearn.manifold import TSNE
```

```
# Load pre-trained Word2Vec model (100-dimensional)
model = api.load("glove-wiki-gigaword-100")

# Print vocabulary size
print("Vocabulary Size:", len(model))

# Display example vector
example_word = "computer"
print("\nExample Word:", example_word)
print("Vector Shape:", model[example_word].shape)
print("First 10 values of vector:\n", model[example_word][:10])
```

Vocabulary Size: 400000

```
Example Word: computer
Vector Shape: (100,)
First 10 values of vector:
[-0.16298  0.30141  0.57978  0.066548  0.45835 -0.15329  0.43258
 -0.89215  0.57747  0.36375 ]
```

```
word_list = [
    # Animals
    "dog", "cat", "lion", "tiger", "elephant", "horse",

    # Countries
    "india", "china", "usa", "france", "germany", "japan",

    # Technology
    "computer", "laptop", "keyboard", "mouse", "internet", "software",

    # Fruits
    "apple", "banana", "mango", "orange", "grape", "pineapple",

    # Vehicles
    "car", "bus", "truck", "bike", "train", "airplane"
]

# Extract vectors
vectors = np.array([model[word] for word in word_list])
```

```
# Initialize t-SNE
tsne = TSNE(n_components=2, random_state=42, perplexity=5)

# Reduce dimensions
reduced_vectors = tsne.fit_transform(vectors)
```

```
print("Original shape:", vectors.shape)
print("Reduced shape:", reduced_vectors.shape)
```

Original shape: (30, 100)  
Reduced shape: (30, 2)

```
plt.figure(figsize=(12, 8))

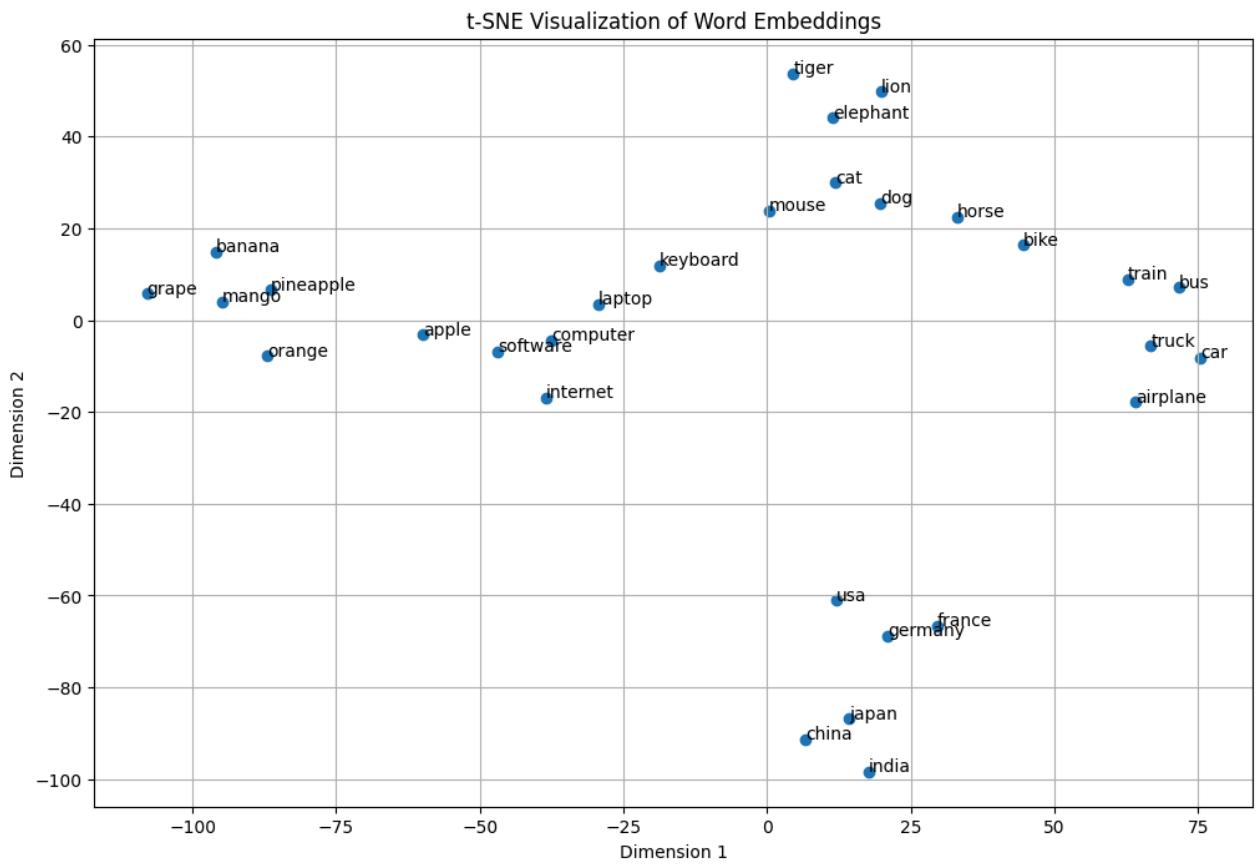
# Scatter plot
x = reduced_vectors[:, 0]
y = reduced_vectors[:, 1]

plt.scatter(x, y)

# Annotate words
for i, word in enumerate(word_list):
    plt.annotate(word, (x[i], y[i]))

plt.title("t-SNE Visualization of Word Embeddings")
plt.xlabel("Dimension 1")
plt.ylabel("Dimension 2")
plt.grid(True)

plt.show()
```



```
colors = (  
    ["red"]*6 +      # Animals  
    ["blue"]*6 +     # Countries
```

```
[ "green"]*6 +      # Technology
[ "orange"]*6 +     # Fruits
[ "purple"]*6        # Vehicles
)

plt.figure(figsize=(12, 8))
plt.scatter(x, y, c=colors)

for i, word in enumerate(word_list):
    plt.annotate(word, (x[i], y[i]))

plt.title("Colored t-SNE Visualization")
plt.show()
```

