mlmodel-test-for-1panel-2

May 18, 2025

[ ]:
```
import pandas as pd
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Load CSV Data
df = pd.read_csv("/content/Sample data.csv")

# Preprocess Text Data
def preprocess_text(text):
    text = str(text).lower()  # Convert to lowercase
    text = re.sub(r'[^\w\s]', '', text)  # Remove punctuation
    return text

# Apply preprocessing to all columns
for col in df.columns:
    df[col] = df[col].apply(preprocess_text)

# Concatenate all text columns into one
# Concatenate all text columns into one
df['all_text'] = df.astype(str).apply(' '.join, axis=1)

# TF-IDF Vectorization
tfidf_vectorizer = TfidfVectorizer()
tfidf_matrix = tfidf_vectorizer.fit_transform(df['all_text'])

# Define the search similar words function
def search_similar_words(input_word, top_n=5):
    input_word_vector = tfidf_vectorizer.transform([input_word])
    cosine_similarities = cosine_similarity(input_word_vector, tfidf_matrix).
 ↪flatten()
    related_indices = cosine_similarities.argsort()[::-1][:top_n]
    return df.iloc[related_indices]

# Example usage
similar_words_df = search_similar_words("Urbanic")
print(similar_words_df)
```

```
   ï»¿Respondent ID Collector ID       Start Date         End Date  \
91      13257735481    413955084  20220113 191020  20220113 192457
56      13257291158    413905455  20220113 142919  20220113 154227
7       13257059830    413905455  20220113 105722  20220113 111905
50      13257398803    413905455  20220113 154500  20220113 162429
72      13257228982    413905455  20220113 134051  20220113 134734

      IP Address Email Address First Name Last Name Custom Data 1  \
91     493687176           nan        nan       nan           nan
56   11799169203           nan        nan       nan           nan
7    11799169203           nan        nan       nan           nan
50   11799169203           nan        nan       nan           nan
72  122166119123           nan        nan       nan           nan

   When was the last time you ordered apparel online?  …  \
91                              in the last 6 months  …
56                              in the last 6 months  …
7                               in the last 6 months  …
50                              in the last 6 months  …
72                              in the last 6 months  …

   What is the gender of the respondent? What is your education level?  \
91                                female                  postgraduate
56                                female                  postgraduate
7                                 female                  postgraduate
50                                female    other professional courses
72                                female                secondary10th

     What is your employment status? Specify employment status if Other  \
91                           student                                 nan
56                 private employee                                  nan
7                  private employee                                  nan
50  professionals doctor  ca  lawyer                                 nan
72                           student                                 nan

   What is your monthly income (in INR)? City of the respondent:  \
91                    less than inr 1000          new delhi  ncr
56                      inr 20001  50000                  mumbai
7                     inr 50001 â 100000          new delhi  ncr
50                      inr 20001  50000                  mumbai
72                                   nan               bangalore

   Name of the respondent: Contact number of the respondent:  \
91              nayanshree                         8527203163
56                   avni                          8319757611
7                  sonali                          8287153386
50                manisha                          8789508751
72            navya saboo                          8088920019
```

```
   Name of the Surveyor:                                              all_text
91         mahima mimani  13257735481 413955084 20220113 191020 20220113…
56              awantika  13257291158 413905455 20220113 142919 20220113…
7               awantika  13257059830 413905455 20220113 105722 20220113…
50              awantika  13257398803 413905455 20220113 154500 20220113…
72                 tusar  13257228982 413905455 20220113 134051 20220113…

[5 rows x 108 columns]
```

```
<ipython-input-14-df524cafdfed>:21: PerformanceWarning: DataFrame is highly
fragmented.  This is usually the result of calling `frame.insert` many times,
which has poor performance.  Consider joining all columns at once using
pd.concat(axis=1) instead. To get a de-fragmented frame, use `newframe =
frame.copy()`
  df['all_text'] = df.astype(str).apply(' '.join, axis=1)
```

important code

[ ]:
```python
import pandas as pd
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Load CSV Data
df = pd.read_csv("/content/Sample data.csv")

# Preprocess Text Data
def preprocess_text(text):
    text = str(text).lower()  # Convert to lowercase
    text = re.sub(r'[^\w\s]', '', text)  # Remove punctuation
    return text

# Apply preprocessing to all columns
for col in df.columns:
    df[col] = df[col].apply(preprocess_text)

# Concatenate all text columns into one
df['all_text'] = pd.concat([df[col].astype(str) for col in df.columns], axis=1).
 ↪apply(lambda row: ' '.join(row), axis=1)

# TF-IDF Vectorization
tfidf_vectorizer = TfidfVectorizer()
tfidf_matrix = tfidf_vectorizer.fit_transform(df['all_text'])

# Define the search similar words function
def search_similar_words(input_word):
    input_word_vector = tfidf_vectorizer.transform([input_word])
```

```
    cosine_similarities = cosine_similarity(input_word_vector, tfidf_matrix).
 ↪flatten()
    related_indices = cosine_similarities.argsort()[::-1]
    similar_rows = df.iloc[related_indices]
    return similar_rows[cosine_similarities > 0]  # Return all matches

# Example usage
similar_words_df = search_similar_words("male")
print(similar_words_df)
```

```
   ï»¿Respondent ID Collector ID       Start Date         End Date  \
8       13257054510    413905455  20220113 105145  20220113 111428
65      13257254534    413905455  20220113 135918  20220113 142033
71      13257228073    413905455  20220113 134005  20220113 135004
73      13257218552    413905455  20220113 133213  20220113 134110
98      13259764922    413956607  20220114 140826  20220114 141941
28      13254635013    413905455  20220112 150210  20220112 150839
67      13257241476    413905455  20220113 135129  20220113 140940
35      13254475727    413905455  20220112 130805  20220112 132737
69      13257233604    413905455  20220113 134448  20220113 140034
15      13254821748    413905455  20220112 170000  20220112 170656
51      13257443454    413905455  20220113 161546  20220113 162106
79      13257185311    413905455  20220113 130238  20220113 130851
68      13257245385    413905455  20220113 135439  20220113 140116
37      13254428638    413905455  20220112 122819  20220112 123642
45      13257554819    413905455  20220113 173148  20220113 173852
34      13254494061    413905455  20220112 132331  20220112 133004
11      13254869551    413905455  20220112 172536  20220112 172938
2       13257116601    413905455  20220113 115757  20220113 120429
32      13254567738    413905455  20220112 141829  20220112 143433
31      13254595331    413905455  20220112 143527  20220112 144415
13      13254825727    413905455  20220112 170209  20220112 171940
14      13254833186    413905455  20220112 170546  20220112 171247
29      13254624282    413905455  20220112 145443  20220112 150610
16      13254810699    413905455  20220112 165357  20220112 170301
26      13254645605    413905455  20220112 150906  20220112 152408
38      13254416448    413905455  20220112 121710  20220112 122631
99      13259759044    413905455  20220114 140318  20220114 141125
82      13258143221    413956607  20220113 220112  20220113 223207
85      13257912142    413956607  20220113 202924  20220113 203651
86      13257871247    413956607  20220113 201136  20220113 202419
87      13257807368    413955084  20220113 194353  20220113 195126
90      13257746570    413956607  20220113 191528  20220113 192937
91      13257735481    413955084  20220113 191020  20220113 192457
92      13257743122    413956607  20220113 191345  20220113 192334
95      13259844197    413905455  20220114 150707  20220114 152041
96      13259831295    413905455  20220114 125439  20220114 151004
```

```
97      13259806111    413955084  20220114 144150  20220114 145513
72      13257228982    413905455  20220113 134051  20220113 134734
70      13257238447    413905455  20220113 134846  20220113 140008

      IP Address Email Address First Name Last Name Custom Data 1  \
8     4936185174           nan        nan       nan           nan
65    4936185174           nan        nan       nan           nan
71   15024217236           nan        nan       nan           nan
73     275990248           nan        nan       nan           nan
98    1214611595           nan        nan       nan           nan
28    1037742172           nan        nan       nan           nan
67   15024217236           nan        nan       nan           nan
35  106223158134           nan        nan       nan           nan
69    1221618115           nan        nan       nan           nan
15    1221618115           nan        nan       nan           nan
51     103774359           nan        nan       nan           nan
79    1037742172           nan        nan       nan           nan
68    1037742172           nan        nan       nan           nan
37    1573548176           nan        nan       nan           nan
45       4937333           nan        nan       nan           nan
34    1573715995           nan        nan       nan           nan
11    1221618115           nan        nan       nan           nan
2     1037742172           nan        nan       nan           nan
32    1573715995           nan        nan       nan           nan
31    1573715995           nan        nan       nan           nan
13   22319093200           nan        nan       nan           nan
14    1573715995           nan        nan       nan           nan
29    1573715995           nan        nan       nan           nan
16    1221618115           nan        nan       nan           nan
26    1573715995           nan        nan       nan           nan
38    1037742172           nan        nan       nan           nan
99    1038314566           nan        nan       nan           nan
82  122170191109           nan        nan       nan           nan
85   12216164123           nan        nan       nan           nan
86   12216164123           nan        nan       nan           nan
87    4731240101           nan        nan       nan           nan
90  106213125240           nan        nan       nan           nan
91     493687176           nan        nan       nan           nan
92    1413924568           nan        nan       nan           nan
95    4936185174           nan        nan       nan           nan
96    4936185174           nan        nan       nan           nan
97   15734100233           nan        nan       nan           nan
72  122166119123           nan        nan       nan           nan
70    4936185174           nan        nan       nan           nan

   When was the last time you ordered apparel online?  …  \
8                              in the last 6 months  …
65                             in the last 6 months  …
```

```
71                                in the last 6 months  …
73                                in the last 6 months  …
98                                in the last 6 months  …
28                                in the last 6 months  …
67                                in the last 6 months  …
35                                in the last 6 months  …
69                                in the last 6 months  …
15                                in the last 6 months  …
51                                in the last 6 months  …
79                                in the last 6 months  …
68                                in the last 6 months  …
37                                in the last 6 months  …
45                                in the last 6 months  …
34                                in the last 6 months  …
11                                in the last 6 months  …
2                                 in the last 6 months  …
32                                in the last 6 months  …
31                                in the last 6 months  …
13                                in the last 6 months  …
14                                in the last 6 months  …
29                                in the last 6 months  …
16                                in the last 6 months  …
26                                in the last 6 months  …
38                                in the last 6 months  …
99                                in the last 6 months  …
82                                in the last 6 months  …
85                                in the last 6 months  …
86                                in the last 6 months  …
87                                in the last 6 months  …
90                                in the last 6 months  …
91                                in the last 6 months  …
92                                in the last 6 months  …
95                                in the last 6 months  …
96                                in the last 6 months  …
97                                in the last 6 months  …
72                                in the last 6 months  …
70                                in the last 6 months  …

   What is the gender of the respondent? What is your education level?  \
8                                    male                      graduate
65                                   male                  postgraduate
71                                   male                      graduate
73                                   male                      graduate
98                                   male                      graduate
28                                   male                      graduate
67                                   male                      graduate
35                                   male                  postgraduate
69                                   male                  postgraduate
```

15		male	graduate
51		male	graduate
79		male	graduate
68		male	graduate
37		male	graduate
45		male	graduate
34		male	graduate
11		male	graduate
2		female	higher secondary12th
32		female	graduate
31		female	graduate
13		female	graduate
14		female	postgraduate
29		female	graduate
16		female	graduate
26		female	graduate
38		female	graduate
99		female	graduate
82		female	postgraduate
85		female	graduate
86		female	graduate
87		female	postgraduate
90		female	graduate
91		female	postgraduate
92		female	postgraduate
95		female	graduate
96		female	graduate
97		female	postgraduate
72		female	secondary10th
70		female	graduate

	What is your employment status?	Specify employment status if Other
8	private employee	nan
65	student	nan
71	private employee	nan
73	student	nan
98	private employee	nan
28	private employee	nan
67	private employee	nan
35	private employee	nan
69	private employee	nan
15	private employee	nan
51	private employee	nan
79	private employee	nan
68	private employee	nan
37	private employee	nan
45	private employee	nan
34	private employee	nan

```
11                 private employee                                  nan
2                            student                                 nan
32                 private employee                                  nan
31              housewife  homemaker                                 nan
13                 private employee                                  nan
14                           student                                 nan
29                 private employee                                  nan
16              housewife  homemaker                                 nan
26                 private employee                                  nan
38                 private employee                                  nan
99                 private employee                                  nan
82              housewife  homemaker                                 nan
85                 private employee                                  nan
86                 private employee                                  nan
87                           student                                 nan
90                 private employee                                  nan
91                           student                                 nan
92                           student                                 nan
95              housewife  homemaker                                 nan
96              housewife  homemaker                                 nan
97                 private employee                                  nan
72                           student                                 nan
70                 private employee                                  nan

   What is your monthly income (in INR)? City of the respondent:  \
8                        inr 20001  50000          new delhi  ncr
65                    inr 50001 â 100000          new delhi  ncr
71                       inr 20001  50000                  mumbai
73                        inr 5001  10000               bangalore
98                       inr 20001  50000                  mumbai
28                    inr 50001 â 100000          new delhi  ncr
67                       inr 20001  50000          new delhi  ncr
35                       inr 20001  50000          new delhi  ncr
69                       inr 20001  50000          new delhi  ncr
15                       inr 10001  20000          new delhi  ncr
51                       inr 20001  50000          new delhi  ncr
79                       inr 20001  50000          new delhi  ncr
68                       inr 20001  50000          new delhi  ncr
37                       inr 10001  20000          new delhi  ncr
45                    inr 50001 â 100000                  mumbai
34                       inr 20001  50000          new delhi  ncr
11                       inr 10001  20000          new delhi  ncr
2                     inr 50001 â 100000          new delhi  ncr
32                       inr 10001  20000                  mumbai
31                    inr 50001 â 100000                  mumbai
13                       inr 10001  20000          new delhi  ncr
14                       inr 20001  50000          new delhi  ncr
29                       inr 10001  20000          new delhi  ncr
```

16	inr 5001 10000	new delhi ncr
26	inr 20001 50000	new delhi ncr
38	inr 10001 20000	new delhi ncr
99	inr 50001 â 100000	mumbai
82	inr 50001 â 100000	mumbai
85	inr 20001 50000	new delhi ncr
86	inr 20001 50000	new delhi ncr
87	inr 5001 10000	new delhi ncr
90	inr 50001 â 100000	new delhi ncr
91	less than inr 1000	new delhi ncr
92	less than inr 1000	new delhi ncr
95	inr 50001 â 100000	mumbai
96	inr 50001 â 100000	mumbai
97	inr 10001 20000	new delhi ncr
72	nan	bangalore
70	inr 20001 50000	new delhi ncr

	Name of the respondent:	Contact number of the respondent: \
8	alok	9555976730
65	yash	8588909580
71	divyansh shukla	8081727612
73	srivatsa	7259915288
98	chandan agrawal	7597308846
28	deepak singh	9599752790
67	parijat pandey	9599133018
35	vivan singh rajput	7451995505
69	dev	8303259480
15	vishal srivastava	8318031794
51	mohit singh	7376161085
79	manish	7235911981
68	abhinash awasthi	7235911986
37	avinash singh	7808521627
45	vansh athwani	9830077489
34	vishu	8726124976
11	mohsin	8840123851
2	babli singh	9341821261
32	sana	9625314874
31	vaishali	8090505679
13	shivani	7753038101
14	shweta gupta	9151315808
29	astha	8354051775
16	neeru	8960189424
26	aradhana	8318112774
38	sanjana	6392998339
99	mithlesh	9354947733
82	smita	9874543210
85	ishmeet kapoor	9179284728
86	tanmeet kapoor	7089710900

87	vidisha agarwal	7017496730
90	shefaali	9818474722
91	nayanshree	8527203163
92	nikita	8800029599
95	heena	9920603317
96	rupali	9768861515
97	sakshi sadana	7580982277
72	navya saboo	8088920019
70	simran	9354137203

	Name of the Surveyor:					all_text
8	nandini	13257054510	413905455	20220113	105145	20220113…
65	nandini	13257254534	413905455	20220113	135918	20220113…
71	navneet rai	13257228073	413905455	20220113	134005	20220113…
73	tusar	13257218552	413905455	20220113	133213	20220113…
98	tusar	13259764922	413956607	20220114	140826	20220114…
28	janwi	13254635013	413905455	20220112	150210	20220112…
67	navneet rai	13257241476	413905455	20220113	135129	20220113…
35	janwi	13254475727	413905455	20220112	130805	20220112…
69	tusar	13257233604	413905455	20220113	134448	20220113…
15	aryan	13254821748	413905455	20220112	170000	20220112…
51	janwi	13257443454	413905455	20220113	161546	20220113…
79	janwi	13257185311	413905455	20220113	130238	20220113…
68	janwi	13257245385	413905455	20220113	135439	20220113…
37	janwi	13254428638	413905455	20220112	122819	20220112…
45	tusar	13257554819	413905455	20220113	173148	20220113…
34	sanjana	13254494061	413905455	20220112	132331	20220112…
11	aryan	13254869551	413905455	20220112	172536	20220112…
2	janwi	13257116601	413905455	20220113	115757	20220113…
32	sanjana	13254567738	413905455	20220112	141829	20220112…
31	sanjana	13254595331	413905455	20220112	143527	20220112…
13	aryan srivastava	13254825727	413905455	20220112	170209	20220112…
14	sanjana	13254833186	413905455	20220112	170546	20220112…
29	sanjana	13254624282	413905455	20220112	145443	20220112…
16	aryan	13254810699	413905455	20220112	165357	20220112…
26	sanjana	13254645605	413905455	20220112	150906	20220112…
38	janwi	13254416448	413905455	20220112	121710	20220112…
99	janwi	13259759044	413905455	20220114	140318	20220114…
82	ms	13258143221	413956607	20220113	220112	20220113…
85	charu shree khandelwal	13257912142	413956607	20220113	202924	20220113…
86	tusar	13257871247	413956607	20220113	201136	20220113…
87	mahima mimani	13257807368	413955084	20220113	194353	20220113…
90	aastha singh	13257746570	413956607	20220113	191528	20220113…
91	mahima mimani	13257735481	413955084	20220113	191020	20220113…
92	aastha	13257743122	413956607	20220113	191345	20220113…
95	nandini	13259844197	413905455	20220114	150707	20220114…
96	nandini	13259831295	413905455	20220114	125439	20220114…
97	mahima mimani	13259806111	413955084	20220114	144150	20220114…

```
72                  tusar  13257228982 413905455 20220113 134051 20220113…
70                nandini  13257238447 413905455 20220113 134846 20220113…

[39 rows x 108 columns]
```

```
<ipython-input-2-00ed8eb89c7b>:20: PerformanceWarning: DataFrame is highly
fragmented.  This is usually the result of calling `frame.insert` many times,
which has poor performance.  Consider joining all columns at once using
pd.concat(axis=1) instead. To get a de-fragmented frame, use `newframe =
frame.copy()`
  df['all_text'] = pd.concat([df[col].astype(str) for col in df.columns],
axis=1).apply(lambda row: ' '.join(row), axis=1)
```

[ ]:
```python
from sklearn.decomposition import TruncatedSVD
from sklearn.neighbors import NearestNeighbors

# TF-IDF Vectorization with optimized parameters
tfidf_vectorizer = TfidfVectorizer(min_df=5, max_df=0.9, ngram_range=(1, 2))
tfidf_matrix = tfidf_vectorizer.fit_transform(df['all_text'])

# Dimensionality reduction with Truncated SVD
svd = TruncatedSVD(n_components=100)
tfidf_matrix_svd = svd.fit_transform(tfidf_matrix)

# Approximate Nearest Neighbors search
ann_model = NearestNeighbors(n_neighbors=100, algorithm='auto')
ann_model.fit(tfidf_matrix_svd)

def search_similar_words(input_word):
    input_word_vector = tfidf_vectorizer.transform([input_word])
    input_word_vector_svd = svd.transform(input_word_vector)
    distances, indices = ann_model.kneighbors(input_word_vector_svd)
    all_similar_rows = df.iloc[indices.flatten()]
    # Filter rows with non-zero cosine similarity
    similar_rows = all_similar_rows[distances.flatten() > 0]
    return similar_rows


# Example usage
similar_words_df = search_similar_words("female")
print(similar_words_df)
```

```
   ï»¿Respondent ID Collector ID      Start Date        End Date  \
84      13258103821    413956607  20220113 214549  20220113 214756
63      13257277703    413905455  20220113 141930  20220113 142353
38      13254416448    413905455  20220112 121710  20220112 122631
2       13257116601    413905455  20220113 115757  20220113 120429
99      13259759044    413905455  20220114 140318  20220114 141125
```

```
..              …          …              …              …
18      13254797080    413905455  20220112 164628  20220112 165705
73      13257218552    413905455  20220113 133213  20220113 134110
11      13254869551    413905455  20220112 172536  20220112 172938
89      13257688706    413955084  20220113 184738  20220113 193729
83      13258113170    413956607  20220113 214927  20220113 215154

     IP Address Email Address First Name Last Name Custom Data 1  \
84  12217724739           nan        nan       nan           nan
63    493612757           nan        nan       nan           nan
38   1037742172           nan        nan       nan           nan
2    1037742172           nan        nan       nan           nan
99   1038314566           nan        nan       nan           nan
..          …            …          …         …             …
18   1221618115           nan        nan       nan           nan
73    275990248           nan        nan       nan           nan
11   1221618115           nan        nan       nan           nan
89  15732248203           nan        nan       nan           nan
83  12217724739           nan        nan       nan           nan

   When was the last time you ordered apparel online?  …  \
84                                in the last 6 months  …
63                                in the last 6 months  …
38                                in the last 6 months  …
2                                 in the last 6 months  …
99                                in the last 6 months  …
..                                                 …  …
18                                in the last 6 months  …
73                                in the last 6 months  …
11                                in the last 6 months  …
89                                in the last 6 months  …
83                                in the last 6 months  …

   What is the gender of the respondent? What is your education level?  \
84                                 female          higher secondary12th
63                                 female                  postgraduate
38                                 female                      graduate
2                                  female          higher secondary12th
99                                 female                      graduate
..                                    …                            …
18                                   male                      graduate
73                                   male                      graduate
11                                   male                      graduate
89             others  prefer not to say                  postgraduate
83                                   male           not finished school

     What is your employment status? Specify employment status if Other  \
84       other working professionals                                 nan
```

```
63                  private employee                                  nan
38                  private employee                                  nan
2                            student                                  nan
99                  private employee                                  nan
..                              …                                   …
18                  private employee                                  nan
73                           student                                  nan
11                  private employee                                  nan
89                           retired                                  nan
83  professionals doctor  ca  lawyer                                  nan

   What is your monthly income (in INR)? City of the respondent:  \
84                        inr 5001  10000                  mumbai
63                   more than inr 100000                  mumbai
38                       inr 10001  20000          new delhi  ncr
2                      inr 50001 â 100000          new delhi  ncr
99                     inr 50001 â 100000                  mumbai
..                                  …                     …
18                       inr 10001  20000          new delhi  ncr
73                        inr 5001  10000               bangalore
11                       inr 10001  20000          new delhi  ncr
89                       inr 20001  50000          new delhi  ncr
83                     less than inr 1000                  mumbai

   Name of the respondent: Contact number of the respondent:  \
84                   vghbjnk                        9999999999
63                    varsha                        9930346314
38                   sanjana                        6392998339
2                 babli singh                       9341821261
99                   mithlesh                       9354947733
..                      …                             …
18          tusar srivastava                        8081068813
73                  srivatsa                        7259915288
11                    mohsin                        8840123851
89                        kt                        9510688425
83                test link                         9831131477

   Name of the Surveyor:                                          all_text
84                  mahija  13258103821 413956607 20220113 214549 20220113…
63                   tusar  13257277703 413905455 20220113 141930 20220113…
38                   janwi  13254416448 413905455 20220112 121710 20220112…
2                    janwi  13257116601 413905455 20220113 115757 20220113…
99                   janwi  13259759044 413905455 20220114 140318 20220114…
..                     …                                              …
18                   aryan  13254797080 413905455 20220112 164628 20220112…
73                   tusar  13257218552 413905455 20220113 133213 20220113…
11                   aryan  13254869551 413905455 20220112 172536 20220112…
89          mahima mimani   13257688706 413955084 20220113 184738 20220113…
```

```
83              test link  13258113170 413956607 20220113 214927 20220113…

[100 rows x 108 columns]
```

[ ]:
```python
import pandas as pd
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import TruncatedSVD
from sklearn.neighbors import NearestNeighbors

# Load CSV Data
df = pd.read_csv("/content/Sample data.csv")

# Preprocess Text Data
def preprocess_text(text):
    text = str(text).lower()  # Convert to lowercase
    text = re.sub(r'[^\w\s]', '', text)  # Remove punctuation
    return text

# Apply preprocessing to all columns
for col in df.columns:
    df[col] = df[col].apply(preprocess_text)

# Concatenate all text columns into one
df['all_text'] = pd.concat([df[col].astype(str) for col in df.columns], axis=1).
 ↪apply(lambda row: ' '.join(row), axis=1)

# TF-IDF Vectorization with optimized parameters
tfidf_vectorizer = TfidfVectorizer(min_df=5, max_df=0.9, ngram_range=(1, 2))
tfidf_matrix = tfidf_vectorizer.fit_transform(df['all_text'])

# Dimensionality reduction with Truncated SVD
svd = TruncatedSVD(n_components=100)
tfidf_matrix_svd = svd.fit_transform(tfidf_matrix)

# Approximate Nearest Neighbors search
ann_model = NearestNeighbors(n_neighbors=len(df), algorithm='auto')
ann_model.fit(tfidf_matrix_svd)

def search_similar_words(input_word):
    input_word_vector = tfidf_vectorizer.transform([input_word])
    input_word_vector_svd = svd.transform(input_word_vector)
    distances, indices = ann_model.kneighbors(input_word_vector_svd)
    all_similar_rows = df.iloc[indices.flatten()]
    # Filter rows with non-zero cosine similarity
    similar_rows = all_similar_rows[distances.flatten() > 0]
    return similar_rows
```

```python
# Example usage
similar_words_df = search_similar_words("13257387743")
print(similar_words_df)
```

```
<ipython-input-51-db75771387ac>:21: PerformanceWarning: DataFrame is highly
fragmented.  This is usually the result of calling `frame.insert` many times,
which has poor performance.  Consider joining all columns at once using
pd.concat(axis=1) instead. To get a de-fragmented frame, use `newframe =
frame.copy()`
  df['all_text'] = pd.concat([df[col].astype(str) for col in df.columns],
axis=1).apply(lambda row: ' '.join(row), axis=1)
```

	ï»¿Respondent ID	Collector ID	Start Date	End Date	\
27	13254646588	413905455	20220112 150935	20220112 152255	
22	13254699723	413905455	20220112 154513	20220112 155400	
52	13257412998	413905455	20220113 155434	20220113 161142	
41	13254298362	413905455	20220112 103851	20220112 105148	
55	13257387743	413905455	20220113 153740	20220113 154641	
..	…	…	…	…	
88	13257771019	413956607	20220113 192724	20220113 194728	
33	13254501520	413905455	20220112 132941	20220112 133649	
4	13257080485	413905455	20220113 111949	20220113 114121	
58	13257347719	413905455	20220113 150857	20220113 151726	
83	13258113170	413956607	20220113 214927	20220113 215154	

	IP Address	Email Address	First Name	Last Name	Custom Data 1	\
27	122161252193	nan	nan	nan	nan	
22	122161252193	nan	nan	nan	nan	
52	15024217236	nan	nan	nan	nan	
41	122161252193	nan	nan	nan	nan	
55	103157220248	nan	nan	nan	nan	
..	…	…	…	…	…	
88	493623962	nan	nan	nan	nan	
33	106223158134	nan	nan	nan	nan	
4	11799169203	nan	nan	nan	nan	
58	1221618115	nan	nan	nan	nan	
83	12217724739	nan	nan	nan	nan	

	When was the last time you ordered apparel online?	…	\
27	in the last 6 months	…	
22	in the last 6 months	…	
52	in the last 6 months	…	
41	in the last 6 months	…	
55	in the last 6 months	…	
..	…	…	
88	in the last 6 months	…	
33	in the last 6 months	…	

```
4                              in the last 6 months  …
58                             in the last 6 months  …
83                             in the last 6 months  …

   What is the gender of the respondent? What is your education level?  \
27                                   male                      graduate
22                                 female                  postgraduate
52                                   male                  postgraduate
41                                 female                      graduate
55                                 female                      graduate
..                                    …                         …
88                                 female                      graduate
33                                   male                  postgraduate
4                                  female                      graduate
58                                 female                      graduate
83                                   male           not finished school

     What is your employment status? Specify employment status if Other  \
27                  private employee                                  nan
22                  private employee                                  nan
52                  private employee                                  nan
41                  private employee                                  nan
55                  private employee                                  nan
..                              …                                   …
88                         student                                    nan
33      entrepreneur  self employed                                  nan
4                   private employee                                  nan
58              housewife  homemaker                                  nan
83  professionals doctor  ca  lawyer                                  nan

   What is your monthly income (in INR)? City of the respondent:  \
27                      inr 20001  50000          new delhi  ncr
22                      inr 20001  50000          new delhi  ncr
52                      inr 20001  50000                  mumbai
41                      inr 20001  50000          new delhi  ncr
55                      inr 20001  50000               bangalore
..                                 …                      …
88                       inr 5001  10000               bangalore
33                  more than inr 100000                  mumbai
4                       inr 20001  50000          new delhi  ncr
58                                 nan          new delhi  ncr
83                    less than inr 1000                  mumbai

   Name of the respondent: Contact number of the respondent:  \
27                    abhay                         9650484530
22                  ipshita                         9647428307
52                    rishi                         8948321048
41                  nandini                         9304724814
```

```
55                    charu                          9887576048
..                      …                                   …
88              purva bucha                          8239122111
33               vivek kumar                         7417475505
4                    sasmita                         8587914784
58                   kalpana                         8303488799
83                 test link                         9831131477

   Name of the Surveyor:                                                 all_text
27                nandini  13254646588 413905455 20220112 150935 20220112…
22                nandini  13254699723 413905455 20220112 154513 20220112…
52           navneet rai   13257412998 413905455 20220113 155434 20220113…
41                 falak   13254298362 413905455 20220112 103851 20220112…
55                 tusar   13257387743 413905455 20220113 153740 20220113…
..                   …                                                   …
88                tusar    13257771019 413956607 20220113 192724 20220113…
33                 janwi   13254501520 413905455 20220112 132941 20220112…
4                awantika   13257080485 413905455 20220113 111949 20220113…
58                 tusar   13257347719 413905455 20220113 150857 20220113…
83               test link 13258113170 413956607 20220113 214927 20220113…

[100 rows x 108 columns]
```

[ ]:
```python
import pandas as pd
import re
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.neighbors import NearestNeighbors

# Load CSV Data
df = pd.read_csv("/content/Sample data.csv")

# Preprocess Text Data
def preprocess_text(text):
    text = str(text).lower()  # Convert to lowercase
    text = re.sub(r'[^\w\s]', '', text)  # Remove punctuation
    return text

# Apply preprocessing to all columns
for col in df.columns:
    df[col] = df[col].apply(preprocess_text)

# Concatenate all text columns into one
df['all_text'] = pd.concat([df[col].astype(str) for col in df.columns], axis=1).
 ↪apply(lambda row: ' '.join(row), axis=1)

# TF-IDF Vectorization
tfidf_vectorizer = TfidfVectorizer(min_df=5, max_df=0.9, ngram_range=(1, 2))
```

```
tfidf_matrix = tfidf_vectorizer.fit_transform(df['all_text'])

# Approximate Nearest Neighbors search with a different algorithm
ann_model = NearestNeighbors(n_neighbors=len(df), algorithm='auto')
ann_model.fit(tfidf_matrix)

def search_similar_words(input_word):
    input_word_vector = tfidf_vectorizer.transform([input_word])
    distances, indices = ann_model.kneighbors(input_word_vector)
    all_similar_rows = df.iloc[indices.flatten()]
    # Filter rows with non-zero cosine similarity
    similar_rows = all_similar_rows[distances.flatten() > 0]
    return similar_rows

# Example usage
similar_words_df = search_similar_words("zara")
print(similar_words_df)
```

```
   ï»¿Respondent ID Collector ID       Start Date         End Date  \
66      13257250421    413905455  20220113 135828  20220113 141022
1       13257124027    413905455  20220113 120549  20220113 121649
76      13257181321    413905455  20220113 125904  20220113 133652
81      13257149831    413905455  20220113 123043  20220113 124219
0       13257101478    413905455  20220113 114214  20220113 122227
..              …           …              …              …
59      13257291711    413905455  20220113 142911  20220113 144622
58      13257347719    413905455  20220113 150857  20220113 151726
33      13254501520    413905455  20220112 132941  20220112 133649
88      13257771019    413956607  20220113 192724  20220113 194728
30      13254577828    413905455  20220112 142505  20220112 144629

      IP Address Email Address First Name Last Name Custom Data 1  \
66    1573718052           nan        nan       nan           nan
1     1573718052           nan        nan       nan           nan
76   11799169203           nan        nan       nan           nan
81    1573718052           nan        nan       nan           nan
0    11799169203           nan        nan       nan           nan
..           …           …        …       …           …
59    4936185174           nan        nan       nan           nan
58    1221618115           nan        nan       nan           nan
33  106223158134           nan        nan       nan           nan
88     493623962           nan        nan       nan           nan
30    1221618115           nan        nan       nan           nan

   When was the last time you ordered apparel online?  …  \
66                              in the last 6 months  …
1                               in the last 6 months  …
```

```
76                              in the last 6 months  …
81                              in the last 6 months  …
0                               in the last 6 months  …
..                                                 …  …
59                              in the last 6 months  …
58                              in the last 6 months  …
33                              in the last 6 months  …
88                              in the last 6 months  …
30                              in the last 6 months  …

   What is the gender of the respondent? What is your education level?  \
66                                female                      graduate
1                                 female                      graduate
76                                female                  postgraduate
81                                female                      graduate
0                                 female                  postgraduate
..                                   …                             …
59                                  male          higher secondary12th
58                                female                      graduate
33                                  male                  postgraduate
88                                female                      graduate
30                                  male                      graduate

   What is your employment status? Specify employment status if Other  \
66                          student                                  nan
1                           student                                  nan
76                private employee                                   nan
81                          student                                  nan
0                 private employee                                   nan
..                              …                                    …
59                          student                                  nan
58              housewife  homemaker                                 nan
33     entrepreneur  self employed                                   nan
88                          student                                  nan
30                private employee                                   nan

   What is your monthly income (in INR)? City of the respondent:  \
66                                   nan                  mumbai
1                    less than inr 1000          new delhi  ncr
76                     inr 20001  50000          new delhi  ncr
81                                   nan          new delhi  ncr
0                      inr 20001  50000          new delhi  ncr
..                                   …                     …
59                   inr 50001 â 100000          new delhi  ncr
58                                   nan          new delhi  ncr
33                 more than inr 100000                  mumbai
88                      inr 5001  10000               bangalore
30                     inr 10001  20000          new delhi  ncr
```

```
   Name of the respondent: Contact number of the respondent:  \
66                  shreya                          9305790113
1                    kiran                          9670162212
76                  ishika                          9899130089
81                karishma                          8081730021
0                    durga                          8791798414
..                     …                                   …
59                  mayank                          7004541165
58                 kalpana                          8303488799
33             vivek kumar                          7417475505
88            purva bucha                           8239122111
30                abhyudai                          7007027337

   Name of the Surveyor:                                        all_text
66               sanjana  13257250421 413905455 20220113 135828 20220113…
1                sanjana  13257124027 413905455 20220113 120549 20220113…
76              awantika  13257181321 413905455 20220113 125904 20220113…
81               sanjana  13257149831 413905455 20220113 123043 20220113…
0               awantika  13257101478 413905455 20220113 114214 20220113…
..                   …                                                 …
59               nandini  13257291711 413905455 20220113 142911 20220113…
58                 tusar  13257347719 413905455 20220113 150857 20220113…
33                 janwi  13254501520 413905455 20220112 132941 20220112…
88                tusar   13257771019 413956607 20220113 192724 20220113…
30                 aryan  13254577828 413905455 20220112 142505 20220112…

[100 rows x 108 columns]
```

```
<ipython-input-57-e1bfe4bb29c7>:20: PerformanceWarning: DataFrame is highly
fragmented.  This is usually the result of calling `frame.insert` many times,
which has poor performance.  Consider joining all columns at once using
pd.concat(axis=1) instead. To get a de-fragmented frame, use `newframe =
frame.copy()`
  df['all_text'] = pd.concat([df[col].astype(str) for col in df.columns],
axis=1).apply(lambda row: ' '.join(row), axis=1)
```

[ ]:
```
!pip install pyspellchecker
```

```
Collecting pyspellchecker
  Downloading pyspellchecker-0.8.1-py3-none-any.whl (6.8 MB)
                           6.8/6.8 MB
20.2 MB/s eta 0:00:00
Installing collected packages: pyspellchecker
Successfully installed pyspellchecker-0.8.1
```

[ ]:
```
import pandas as pd
```

```python
# Load your CSV file
df = pd.read_csv('/content/Sample data.csv')

# Define your keyword
keyword = 'zara'

# Use the apply function to check each cell for the keyword
mask = df.applymap(lambda x: keyword.lower() in str(x).lower())

# Get the rows where the keyword is found
result = df[mask.any(axis=1)]

# Print the result
print(result)
```

```
    ï»¿Respondent ID  Collector ID           Start Date             End Date  \
0         13257101478     413905455  2022-01-13 11:42:14  2022-01-13 12:22:27
1         13257124027     413905455  2022-01-13 12:05:49  2022-01-13 12:16:49
4         13257080485     413905455  2022-01-13 11:19:49  2022-01-13 11:41:21
7         13257059830     413905455  2022-01-13 10:57:22  2022-01-13 11:19:05
8         13257054510     413905455  2022-01-13 10:51:45  2022-01-13 11:14:28
25        13254668858     413905455  2022-01-12 15:24:48  2022-01-12 15:31:47
27        13254646588     413905455  2022-01-12 15:09:35  2022-01-12 15:22:55
40        13254307047     413905455  2022-01-12 10:46:07  2022-01-12 10:57:47
50        13257398803     413905455  2022-01-13 15:45:00  2022-01-13 16:24:29
56        13257291158     413905455  2022-01-13 14:29:19  2022-01-13 15:42:27
66        13257250421     413905455  2022-01-13 13:58:28  2022-01-13 14:10:22
75        13257215194     413905455  2022-01-13 13:29:20  2022-01-13 13:38:20
76        13257181321     413905455  2022-01-13 12:59:04  2022-01-13 13:36:52
81        13257149831     413905455  2022-01-13 12:30:43  2022-01-13 12:42:19
95        13259844197     413905455  2022-01-14 15:07:07  2022-01-14 15:20:41
96        13259831295     413905455  2022-01-14 12:54:39  2022-01-14 15:10:04
97        13259806111     413955084  2022-01-14 14:41:50  2022-01-14 14:55:13

          IP Address  Email Address  First Name  Last Name  Custom Data 1  \
0     117.99.169.203            NaN         NaN        NaN            NaN
1      157.37.180.52            NaN         NaN        NaN            NaN
4     117.99.169.203            NaN         NaN        NaN            NaN
7     117.99.169.203            NaN         NaN        NaN            NaN
8      49.36.185.174            NaN         NaN        NaN            NaN
25     157.37.159.95            NaN         NaN        NaN            NaN
27   122.161.252.193            NaN         NaN        NaN            NaN
40   110.235.233.220            NaN         NaN        NaN            NaN
50    117.99.169.203            NaN         NaN        NaN            NaN
56    117.99.169.203            NaN         NaN        NaN            NaN
66     157.37.180.52            NaN         NaN        NaN            NaN
75   122.166.119.123            NaN         NaN        NaN            NaN
```

```
76   117.99.169.203            NaN         NaN        NaN            NaN
81    157.37.180.52            NaN         NaN        NaN            NaN
95    49.36.185.174            NaN         NaN        NaN            NaN
96    49.36.185.174            NaN         NaN        NaN            NaN
97   157.34.100.233            NaN         NaN        NaN            NaN

   When was the last time you ordered apparel online?  …  \
0                                In the last 6 months  …
1                                In the last 6 months  …
4                                In the last 6 months  …
7                                In the last 6 months  …
8                                In the last 6 months  …
25                               In the last 6 months  …
27                               In the last 6 months  …
40                               In the last 6 months  …
50                               In the last 6 months  …
56                               In the last 6 months  …
66                               In the last 6 months  …
75                               In the last 6 months  …
76                               In the last 6 months  …
81                               In the last 6 months  …
95                               In the last 6 months  …
96                               In the last 6 months  …
97                               In the last 6 months  …

   What is your current age? What is the gender of the respondent?  \
0              18 - 24 years                                 Female
1              18 - 24 years                                 Female
4              18 - 24 years                                 Female
7              18 - 24 years                                 Female
8              18 - 24 years                                   Male
25             25 - 34 years                                 Female
27             18 - 24 years                                   Male
40             18 - 24 years                                 Female
50             25 - 34 years                                 Female
56             25 - 34 years                                 Female
66             18 - 24 years                                 Female
75        Less than 18 years                                 Female
76             25 - 34 years                                 Female
81             18 - 24 years                                 Female
95             25 - 34 years                                 Female
96             35 - 44 years                                 Female
97             18 - 24 years                                 Female

   What is your education level?       What is your employment status?  \
0                 Post-graduate                      Private employee
1                      Graduate                               Student
4                      Graduate                      Private employee
```

7	Post-graduate	Private employee
8	Graduate	Private employee
25	Post-graduate	Housewife / Homemaker
27	Graduate	Private employee
40	Graduate	Private employee
50	Other professional courses	Professionals (Doctor / CA / Lawyer)
56	Post-graduate	Private employee
66	Graduate	Student
75	Graduate	Student
76	Post-graduate	Private employee
81	Graduate	Student
95	Graduate	Housewife / Homemaker
96	Graduate	Housewife / Homemaker
97	Post-graduate	Private employee

	Specify employment status if Other	What is your monthly income (in INR)?
0	NaN	INR 20,001 - 50,000
1	NaN	Less than INR 1,000
4	NaN	INR 20,001 - 50,000
7	NaN	INR 50,001 â€" 1,00,000
8	NaN	INR 20,001 - 50,000
25	NaN	INR 20,001 - 50,000
27	NaN	INR 20,001 - 50,000
40	NaN	INR 20,001 - 50,000
50	NaN	INR 20,001 - 50,000
56	NaN	INR 20,001 - 50,000
66	NaN	NaN
75	NaN	Less than INR 1,000
76	NaN	INR 20,001 - 50,000
81	NaN	NaN
95	NaN	INR 50,001 â€" 1,00,000
96	NaN	INR 50,001 â€" 1,00,000
97	NaN	INR 10,001 - 20,000

	City of the respondent:	Name of the respondent:
0	New Delhi / NCR	Durga
1	New Delhi / NCR	kiran
4	New Delhi / NCR	sasmita
7	New Delhi / NCR	Sonali
8	New Delhi / NCR	Alok
25	New Delhi / NCR	Ritika
27	New Delhi / NCR	Abhay
40	New Delhi / NCR	Falak
50	Mumbai	Manisha
56	Mumbai	Avni
66	Mumbai	Shreya
75	Bangalore	Ishita Saboo
76	New Delhi / NCR	ishika

```
81          New Delhi / NCR                karishma
95                   Mumbai                   Heena
96                   Mumbai                  Rupali
97          New Delhi / NCR           Sakshi Sadana

   Contact number of the respondent: Name of the Surveyor:
0                          8791798414              Awantika
1                          9670162212               sanjana
4                          8587914784              Awantika
7                          8287153386              Awantika
8                          9555976730               Nandini
25                         7754992657               sanjana
27                         9650484530               Nandini
40                         8429225110               Nandini
50                         8789508751              Awantika
56                         8319757611              Awantika
66                         9305790113               sanjana
75                         7975848446                 Tusar
76                         9899130089              Awantika
81                         8081730021               sanjana
95                         9920603317               Nandini
96                         9768861515               Nandini
97                         7580982277         Mahima Mimani

[17 rows x 107 columns]
```

[ ]:
```
/content/Sample data.csv
```

[ ]:
```
!pip install transformers
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-
packages (4.40.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-
packages (from transformers) (3.14.0)
Requirement already satisfied: huggingface-hub<1.0,>=0.19.3 in
/usr/local/lib/python3.10/dist-packages (from transformers) (0.20.3)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-
packages (from transformers) (1.25.2)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.10/dist-packages (from transformers) (24.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-
packages (from transformers) (6.0.1)
Requirement already satisfied: regex!=2019.12.17 in
/usr/local/lib/python3.10/dist-packages (from transformers) (2023.12.25)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-
packages (from transformers) (2.31.0)
Requirement already satisfied: tokenizers<0.20,>=0.19 in
/usr/local/lib/python3.10/dist-packages (from transformers) (0.19.1)
```

```
Requirement already satisfied: safetensors>=0.4.1 in
/usr/local/lib/python3.10/dist-packages (from transformers) (0.4.3)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-
packages (from transformers) (4.66.4)
Requirement already satisfied: fsspec>=2023.5.0 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub<1.0,>=0.19.3->transformers) (2023.6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.10/dist-packages (from huggingface-
hub<1.0,>=0.19.3->transformers) (4.11.0)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests->transformers) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests->transformers) (2024.2.2)
```

[ ]:
```python
import pandas as pd
from transformers import BertForMaskedLM, BertTokenizer

# Load your CSV file
df = pd.read_csv('/content/Sample data.csv')

# Define your keyword
keyword = 'frui'

# Initialize the BERT tokenizer and model
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
model = BertForMaskedLM.from_pretrained('bert-base-uncased')

# Prepare the inputs for the model
inputs = tokenizer.encode(f'{keyword} {tokenizer.mask_token}',␣
 ↪return_tensors='pt')

# Get the prediction from the model
prediction = model(inputs)[0]

# Get the index of the masked token
masked_index = tokenizer.convert_tokens_to_ids(tokenizer.mask_token)

# Get the top 5 predictions
top_5_predictions = prediction[0, masked_index].topk(5).indices.tolist()

# Get the corrected keyword
corrected_keyword = tokenizer.decode(top_5_predictions[0])
```

```python
# Use the apply function to check each cell for the keyword
mask = df.applymap(lambda x: corrected_keyword.lower() in str(x).lower())

# Get the rows where the keyword is found
result = df[mask.any(axis=1)]

# Print the result
print(result)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:88:
UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab
(https://huggingface.co/settings/tokens), set it as secret in your Google Colab
and restart your session.
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access
public models or datasets.
  warnings.warn(

tokenizer_config.json:   0%|          | 0.00/48.0 [00:00<?, ?B/s]

vocab.txt:   0%|          | 0.00/232k [00:00<?, ?B/s]

tokenizer.json:   0%|          | 0.00/466k [00:00<?, ?B/s]

config.json:   0%|          | 0.00/570 [00:00<?, ?B/s]

model.safetensors:   0%|          | 0.00/440M [00:00<?, ?B/s]

Some weights of the model checkpoint at bert-base-uncased were not used when
initializing BertForMaskedLM: ['bert.pooler.dense.bias',
'bert.pooler.dense.weight', 'cls.seq_relationship.bias',
'cls.seq_relationship.weight']
- This IS expected if you are initializing BertForMaskedLM from the checkpoint
of a model trained on another task or with another architecture (e.g.
initializing a BertForSequenceClassification model from a BertForPreTraining
model).
- This IS NOT expected if you are initializing BertForMaskedLM from the
checkpoint of a model that you expect to be exactly identical (initializing a
BertForSequenceClassification model from a BertForSequenceClassification model).
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-78-1c3b85439482> in <cell line: 24>()
     22
     23 # Get the top 5 predictions
---> 24 top_5_predictions = prediction[0, masked_index].topk(5).indices.tolist(
     25
```

```
    26 # Get the corrected keyword

IndexError: index 103 is out of bounds for dimension 1 with size 5
```