

```
df['fulltext'] = df['short_description'] + ' ' + df['description']
```

```
+ cc + location
```

```
In [1]: from __future__ import absolute_import, division, print_function, unicode_literals

import numpy as np
import pandas as pd
import tensorflow as tf

from tensorflow import feature_column
from tensorflow.keras import layers
from sklearn.model_selection import train_test_split
import math
from sklearn.ensemble import RandomForestClassifier

from sklearn import metrics
from sklearn.metrics import accuracy_score, balanced_accuracy_score, f1_score, classification_report
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import confusion_matrix
```

Import data

```
In [2]: #import data (do only once)
cases = pd.read_excel('incident V2 - Enriched.xlsx')
cases.shape
```

```
Out[2]: (34564, 45)
```

Set parameters

```
In [3]: #Variables
no_of_ags = 15 #Number of AGs to consider when choosing AGs with most frequency
#AGs to exclude from analysis. Leave blank if none is excluded. This may change depending on years chosen
remove_ags = ['Global Helpdesk - Tier 1', 'Japan Helpdesk Support', 'Global Helpdesk', 'Global ITSOC - Tier 1' ] #for 2018 and 2019
#Define test and train sets cases['opened_at'].dt.to_period('M')
test_period = ['2019-02', '2019-03', '2019-04']
train_period = ['2018-01', '2018-02', '2018-03', '2018-04', '2018-05', '2018-06', '2018-07', '2018-08', '2018-09', '2018-10', '2018-11', '2018-12', '2019-01']
#rounds = 100 #Number of times RF is run
# Whether to merge AGs or not
merge_ags = 'Y' #Set to 'N' if you dont want to merge AGs
model_params = dict(((k, eval(k)) for k in ('no_of_ags', 'remove_ags', 'test_period', 'train_period', 'merge_ags' )))
```

Data prep

```

In [4]: # Use copy so that we dont have import data for every run
df = cases.copy()
print('Full dataset shape:', df.shape)

# Use only closed cases
df = df[df['state'].isin(['Closed', 'Closed (CR Implemented)', 'Closed (Purchase Required)', 'Resolved'])].copy()
print('Only closed cases shape:', df.shape)

if merge_ag == 'Y':
    df['ag_merged'] = np.where(df['ag'].isin(['Finance Support', 'IT BSA - Finance']), 'Merged Finance Support IT BSA - Finance', df['ag'])
    #Reset column names for convenience
    df.rename(columns={'ag': 'ag_old'}, inplace=True)
    df.rename(columns={'ag_merged': 'ag'}, inplace=True)

#Create new text feature
for cols in ['u_classification', 'u_requester', 'short_description', 'description', 'cldb_ci', 'Requester Person ID',
            'Requester User Id', 'Requester Full Name', 'Requester Grade', 'Requester Supervisor', 'Requester Cost Center Descr',
            'Requester Location Desc']:
    #for cols in ['short_description', 'description', 'Requester Cost Center Descr', 'Requester Location Desc']:
        df[cols] = df[cols].astype(str)

df['fulltext'] = df['short_description'] + ' ' + df['description'] + ' ' + df['Requester Cost Center Descr'] + ' ' + df['Requester Location Desc']

#Filter cases based on period chosen
df['opened_at'] = pd.to_datetime(df['opened_at'])
df = df[df['opened_at'].dt.to_period('M').astype(str).isin(test_period + train_period)].copy()
print('Shape after selecting period', df.shape)

df_size = len(df)

#Filter cases based on AGs
keep_ag = list(df['ag'].value_counts().head(no_of_ag).index)
for i in remove_ag:
    keep_ag.remove(i)
df = df[df['ag'].isin(keep_ag)].copy()
print()
print('Shape of data subset: ', df.shape)

#Percentage of cases considered
print()
print('% of cases considered after taking subset: ', len(df)*100/df_size)

#AGs List and frequency
print()
print('AG list and frequencies')
print(df['ag'].value_counts())

#Change all object type to category

```

```
df[df.select_dtypes(['object']).columns] = df.select_dtypes(['object']).apply(lambda x: x.astype('category'))

# Display code to category mapping
print()
print('AG to codes mapping')
class_to_cat_mapping = dict(enumerate(df['ag'].cat.categories))
print(class_to_cat_mapping)

#Change AG to codes
df['ag'] = df['ag'].cat.codes
```

Full dataset shape: (34564, 45)

Only closed cases shape: (30874, 45)

Shape after selecting period (5834, 47)

Shape of data subset: (5453, 47)

% of cases considered after taking subset: 93.46931779225231

AG list and frequencies

Merged Finance Support IT BSA - Finance	3182
IT BSA - Billing C&C	1171
RevOps Support	362
Bus - Billing C&C	203
IT BSA - Singleview Ops	170
IT BSA - Vertex	149
Global DBA Support	74
Hyperion Team	54
IT BSA - BI Team	52
Singleview Admin	23
GCC Support Group	13

Name: ag, dtype: int64

AG to codes mapping

```
{0: 'Bus - Billing C&C', 1: 'GCC Support Group', 2: 'Global DBA Support',
3: 'Hyperion Team', 4: 'IT BSA - BI Team', 5: 'IT BSA - Billing C&C', 6: 'IT BSA - Singleview Ops', 7: 'IT BSA - Vertex', 8: 'Merged Finance Support IT BSA - Finance', 9: 'RevOps Support', 10: 'Singleview Admin'}
```

```
In [5]: #d = df[['u_requester', 'Requester Grade', 'Requester Supervisor', 'Requester Cost Center Descr', 'Requester Location Desc', 'ag', 'fulltext']].copy()
d = df[['Requester Cost Center Descr', 'Requester Location Desc', 'ag', 'fulltext']].copy()
```

```
In [6]: d.columns = ['cc', 'location', 'target', 'text']
```

```
In [7]: d.to_csv('df text and cols.csv')
```

Create tf idf

```
In [8]: d = pd.read_csv('df text and cols.csv')
d.drop('Unnamed: 0', axis=1, inplace=True)
```

```
In [9]: d.columns
```

```
Out[9]: Index(['cc', 'location', 'target', 'text'], dtype='object')
```

```
In [10]: for cols in ['cc', 'location', 'text']:
          d[cols] = d[cols].astype(str)
```

```
In [11]: #Remove stop words in English when creating tf idf vector and create train set
vectorizer = TfidfVectorizer(stop_words='english')
tfidf = vectorizer.fit_transform(d['text'].values)
#tfidf = vectorizer.transform(rose['text'].values)
tfidf
```

```
Out[11]: <5453x56048 sparse matrix of type '<class 'numpy.float64'>'
         with 592933 stored elements in Compressed Sparse Row format>
```

```
In [12]: #Run RF on tf idf and fit and find important features
m = RandomForestClassifier(n_estimators=1000, n_jobs=-1)
m.fit(tfidf, d['target'])

#Use the feature importance to find the most important words
feature_importance = pd.DataFrame({'Feature' : vectorizer.get_feature_names(),
                                   'Importance' : m.feature_importances_})
feature_importance.sort_values('Importance', ascending=False, inplace=True)
```

```
In [13]: # Create customer stop words
#Consider words with importance less than 0.0001 as unimportant and remove them from tf idf
words_to_remove = feature_importance[feature_importance['Importance'] < 0.0001]['Feature']

#Add words to remove to stop words and create new tf idf
from sklearn.feature_extraction import text
from sklearn.feature_extraction.text import TfidfVectorizer
my_stop_words = text.ENGLISH_STOP_WORDS.union(words_to_remove)
len(my_stop_words)
```

```
Out[13]: 54831
```

```
In [14]: #Remove custom stop words in English when creating tf idf vector and create train set
vectorizer = TfidfVectorizer(stop_words=my_stop_words)
tfidf = vectorizer.fit_transform(d['text'].values)
```

```
In [15]: tfidf.shape
```

```
Out[15]: (5453, 1535)
```

```
In [16]: tfidf_df = pd.DataFrame(tfidf.todense())
```

```
In [17]: tfidf_df.columns
```

```
Out[17]: RangeIndex(start=0, stop=1535, step=1)
```

```
In [18]: cols = []
for i in tfidf_df.columns:
    cols.append('word_' + str(i))

tfidf_df.columns = cols
```

```
In [19]: tfidf_df.head()
```

```
Out[19]:
```

	word_0	word_1	word_2	word_3	word_4	word_5	word_6	word_7	word_8	word_9	...	word_1534
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0

5 rows × 1535 columns

Add other columns and split into train and test

```
In [20]: tfidf_df['target'] = d['target']
```

```
In [21]: all_cols_df = pd.concat([tfidf_df, d[['cc', 'location']]], axis=1)
```

```
In [22]: for cols in ['cc', 'location']:
    all_cols_df[cols] = all_cols_df[cols].astype(str)
```

```
In [23]: all_cols_df.columns
```

```
Out[23]: Index(['word_0', 'word_1', 'word_2', 'word_3', 'word_4', 'word_5', 'word_6',
              'word_7', 'word_8', 'word_9',
              ...,
              'word_1528', 'word_1529', 'word_1530', 'word_1531', 'word_1532',
              'word_1533', 'word_1534', 'target', 'cc', 'location'],
              dtype='object', length=1538)
```

```
In [24]: #Split into train, test and val
train, test = train_test_split(all_cols_df, test_size=0.2)
train, val = train_test_split(train, test_size=0.2)
train.shape, test.shape, val.shape
```

```
Out[24]: ((3489, 1538), (1091, 1538), (873, 1538))
```

Create feature columns

```
In [25]: #Change all category columns to feature_columns
#requester = feature_column.categorical_column_with_vocabulary_list('requester', all_cols_df['requester'].unique())
#grade = feature_column.categorical_column_with_vocabulary_list('grade', all_cols_df['grade'].unique())
#sup = feature_column.categorical_column_with_vocabulary_list('sup', all_cols_df['sup'].unique())
cc = feature_column.categorical_column_with_vocabulary_list('cc', all_cols_df['cc'].unique())
location = feature_column.categorical_column_with_vocabulary_list('location', all_cols_df['location'].unique())
```

```
In [26]: #To create embeddings we use the categorical feature columns created above
#requester_embedding = feature_column.embedding_column(requester, dimension=50)
#grade_embedding = feature_column.embedding_column(grade, dimension=8)
#sup_embedding = feature_column.embedding_column(sup, dimension=8)
cc_embedding = feature_column.embedding_column(cc, dimension=8)
location_embedding = feature_column.embedding_column(location, dimension=8)
```

```
In [27]: #Create feature columns set
feature_columns = []
#feature_columns.append(requester_embedding)
#feature_columns.append(grade_embedding)
#feature_columns.append(sup_embedding)
feature_columns.append(cc_embedding)
feature_columns.append(location_embedding)
```

```
In [28]: #Append all numerical columns
cont_cols = list(all_cols_df.describe().columns)
cont_cols.remove('target')

for header in cont_cols:
    feature_columns.append(feature_column.numeric_column(header))
```

Create layers and run NN

```
In [29]: #Create a layer from the feature columns
feature_layer = tf.keras.layers.DenseFeatures(feature_columns)
```

```
In [30]: # A utility method to create a tf.data dataset from a Pandas Dataframe
def df_to_dataset(dataframe, shuffle=True, batch_size=32):
    dataframe = dataframe.copy()
    labels = dataframe.pop('target')
    ds = tf.data.Dataset.from_tensor_slices((dict(dataframe), labels))
    if shuffle:
        ds = ds.shuffle(buffer_size=len(dataframe))
    ds = ds.batch(batch_size)
    return ds
```

Find best iteration


```

In [ ]: df_batch_size = []
df_layers = []
df_epochs = []
df_accuracy = []
df_act = []
df_opt = []
df_drop = []

for batch_size in [16,32,64,128]:
    train_ds = df_to_dataset(train, batch_size=batch_size)
    val_ds = df_to_dataset(val, shuffle=False, batch_size=batch_size)
    test_ds = df_to_dataset(test, shuffle=False, batch_size=batch_size)
    for act in ['sigmoid', 'softmax']:
        for opt in ['adam', 'sgd']:
            for layer in [128, 256, 512, 720, 1024]:
                for drop in [0, 0.1, 0.2, 0.3, 0.4]:
                    #Create model
                    model = tf.keras.Sequential([
                        feature_layer,
                        layers.Dropout(drop),
                        layers.Dense(layer, activation='relu'),
                        layers.Dense(11, activation=act)
                    ])

                    #Compile model
                    model.compile(optimizer=opt,loss='sparse_categorical_
crossentropy', metrics=['accuracy'])

                    acc1 = 0
                    count = 1
                    while True:
                        model.fit(train_ds, validation_data=val_ds, epoch
s=1) #, steps_per_epoch=math.ceil(num_train_examples/batch_size))
                        acc2 = accuracy_score(test['target'], model.predi
ct(test_ds).argmax(axis=1))
                        if acc2 > acc1:
                            acc1 = acc2
                            count = count + 1
                            print('Activation:', act, '/', 'Optimizer:', opt
, '/', 'Batch size:', batch_size, '/', 'Layers:', layer, '/', 'Epochs:', count,
 '/', 'Accuracy:', acc2)

                            df_batch_size.append(batch_size)
                            df_drop.append(drop)
                            df_layers.append(layer)
                            df_epochs.append(count)
                            df_accuracy.append(acc2)
                            df_act.append(act)
                            df_opt.append(opt)
                            continue
                        else:
                            print('Activation:', act, '/', 'Optimizer:', opt
, '/', 'Batch size:', batch_size, '/', 'Layers:', layer, '/', 'Epochs:', count,
 '/', 'Accuracy:', acc1)

                            df_batch_size.append(batch_size)
                            df_drop.append(drop)
                            df_layers.append(layer)

```

```
df_epochs.append(count)
df_accuracy.append(acc1)
df_act.append(act)
df_opt.append(opt)
break
```

219/219 [=====] - 102s 466ms/step - loss: 1.1884 - accuracy: 0.6301 - val_loss: 0.7483 - val_accuracy: 0.7824
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 2 / Accuracy: 0.7873510540788268

219/219 [=====] - 94s 427ms/step - loss: 0.6079 - accuracy: 0.8054 - val_loss: 0.6244 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 3 / Accuracy: 0.8139321723189734

219/219 [=====] - 94s 427ms/step - loss: 0.4585 - accuracy: 0.8467 - val_loss: 0.5780 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 4 / Accuracy: 0.8240146654445463

219/219 [=====] - 98s 447ms/step - loss: 0.3617 - accuracy: 0.8773 - val_loss: 0.5633 - val_accuracy: 0.8179
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 4 / Accuracy: 0.8240146654445463

219/219 [=====] - 108s 492ms/step - loss: 1.1506 - accuracy: 0.6250 - val_loss: 0.7534 - val_accuracy: 0.7778
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 2 / Accuracy: 0.7891842346471127

219/219 [=====] - 98s 448ms/step - loss: 0.6219 - accuracy: 0.7988 - val_loss: 0.6312 - val_accuracy: 0.7915
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 3 / Accuracy: 0.8120989917506874

219/219 [=====] - 101s 461ms/step - loss: 0.4827 - accuracy: 0.8383 - val_loss: 0.5798 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 4 / Accuracy: 0.8194317140238313

219/219 [=====] - 102s 467ms/step - loss: 0.3985 - accuracy: 0.8664 - val_loss: 0.5585 - val_accuracy: 0.8167
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 5 / Accuracy: 0.8249312557286893

219/219 [=====] - 99s 452ms/step - loss: 0.3306 - accuracy: 0.8859 - val_loss: 0.5603 - val_accuracy: 0.8190
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 5 / Accuracy: 0.8249312557286893

219/219 [=====] - 168s 766ms/step - loss: 1.1708 - accuracy: 0.6110 - val_loss: 0.7520 - val_accuracy: 0.7812
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 2 / Accuracy: 0.7901008249312558

219/219 [=====] - 160s 730ms/step - loss: 0.6370 - accuracy: 0.7997 - val_loss: 0.6294 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 3 / Accuracy: 0.8139321723189734

219/219 [=====] - 160s 731ms/step - loss: 0.5013 - accuracy: 0.8332 - val_loss: 0.5822 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 3 / Accuracy: 0.8139321723189734

219/219 [=====] - 173s 789ms/step - loss: 1.1889 - accuracy: 0.6591 - val_loss: 0.7642 - val_accuracy: 0.7778
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 2 / Accuracy: 0.7873510540788268

219/219 [=====] - 157s 717ms/step - loss: 0.6696 - accuracy: 0.7893 - val_loss: 0.6504 - val_accuracy: 0.7915
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 3 / Accuracy: 0.8075160403299725

219/219 [=====] - 158s 723ms/step - loss: 0.5379 -

accuracy: 0.8237 - val_loss: 0.5944 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 4 / Accuracy: 0.8157653528872594
219/219 [=====] - 159s 726ms/step - loss: 0.4606 - accuracy: 0.8469 - val_loss: 0.5707 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 5 / Accuracy: 0.8230980751604033
219/219 [=====] - 155s 707ms/step - loss: 0.4008 - accuracy: 0.8667 - val_loss: 0.5696 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 5 / Accuracy: 0.8230980751604033
219/219 [=====] - 108s 493ms/step - loss: 1.1727 - accuracy: 0.6223 - val_loss: 0.7996 - val_accuracy: 0.7721
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 2 / Accuracy: 0.7791017415215399
219/219 [=====] - 100s 455ms/step - loss: 0.6997 - accuracy: 0.7825 - val_loss: 0.6630 - val_accuracy: 0.7915
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 3 / Accuracy: 0.8029330889092575
219/219 [=====] - 105s 479ms/step - loss: 0.5741 - accuracy: 0.8151 - val_loss: 0.6088 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 106s 486ms/step - loss: 0.4934 - accuracy: 0.8346 - val_loss: 0.5742 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 5 / Accuracy: 0.8185151237396884
219/219 [=====] - 100s 456ms/step - loss: 0.4415 - accuracy: 0.8550 - val_loss: 0.5636 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 128 / Epochs: 5 / Accuracy: 0.8185151237396884
219/219 [=====] - 114s 519ms/step - loss: 0.9666 - accuracy: 0.6953 - val_loss: 0.6866 - val_accuracy: 0.7904
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 2 / Accuracy: 0.8020164986251146
219/219 [=====] - 104s 476ms/step - loss: 0.5148 - accuracy: 0.8300 - val_loss: 0.6000 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 3 / Accuracy: 0.8157653528872594
219/219 [=====] - 107s 488ms/step - loss: 0.3841 - accuracy: 0.8702 - val_loss: 0.5723 - val_accuracy: 0.8156
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 3 / Accuracy: 0.8157653528872594
219/219 [=====] - 130s 596ms/step - loss: 0.9733 - accuracy: 0.6721 - val_loss: 0.6978 - val_accuracy: 0.7858
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 2 / Accuracy: 0.7974335472043996
219/219 [=====] - 111s 507ms/step - loss: 0.5380 - accuracy: 0.8266 - val_loss: 0.6045 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 3 / Accuracy: 0.8139321723189734
219/219 [=====] - 103s 470ms/step - loss: 0.4182 - accuracy: 0.8576 - val_loss: 0.5684 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8175985334555453
219/219 [=====] - 108s 491ms/step - loss: 0.3351 - accuracy: 0.8851 - val_loss: 0.5627 - val_accuracy: 0.8213

Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8175985334555453
219/219 [=====] - 113s 517ms/step - loss: 0.9995 - accuracy: 0.6663 - val_loss: 0.7060 - val_accuracy: 0.7858
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 2 / Accuracy: 0.8001833180568286
219/219 [=====] - 105s 479ms/step - loss: 0.5713 - accuracy: 0.8174 - val_loss: 0.6092 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 3 / Accuracy: 0.8130155820348305
219/219 [=====] - 110s 502ms/step - loss: 0.4550 - accuracy: 0.8512 - val_loss: 0.5743 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8240146654445463
219/219 [=====] - 113s 514ms/step - loss: 0.3707 - accuracy: 0.8753 - val_loss: 0.5645 - val_accuracy: 0.8156
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8240146654445463
219/219 [=====] - 112s 509ms/step - loss: 1.0344 - accuracy: 0.6555 - val_loss: 0.7338 - val_accuracy: 0.7755
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 2 / Accuracy: 0.7901008249312558
219/219 [=====] - 99s 452ms/step - loss: 0.6099 - accuracy: 0.8042 - val_loss: 0.6259 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 3 / Accuracy: 0.8038496791934006
219/219 [=====] - 102s 464ms/step - loss: 0.4967 - accuracy: 0.8401 - val_loss: 0.5795 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 101s 461ms/step - loss: 0.4107 - accuracy: 0.8576 - val_loss: 0.5631 - val_accuracy: 0.8225
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 5 / Accuracy: 0.8203483043079743
219/219 [=====] - 100s 455ms/step - loss: 0.3632 - accuracy: 0.8730 - val_loss: 0.5611 - val_accuracy: 0.8167
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 5 / Accuracy: 0.8203483043079743
219/219 [=====] - 110s 503ms/step - loss: 1.0489 - accuracy: 0.6469 - val_loss: 0.7301 - val_accuracy: 0.7835
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 2 / Accuracy: 0.7882676443629697
219/219 [=====] - 97s 441ms/step - loss: 0.6429 - accuracy: 0.7954 - val_loss: 0.6222 - val_accuracy: 0.7892
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 3 / Accuracy: 0.8084326306141155
219/219 [=====] - 98s 449ms/step - loss: 0.5214 - accuracy: 0.8289 - val_loss: 0.5822 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8185151237396884
219/219 [=====] - 100s 459ms/step - loss: 0.4478 - accuracy: 0.8475 - val_loss: 0.5674 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 256 / Epochs: 4 / Accuracy: 0.8185151237396884
219/219 [=====] - 127s 579ms/step - loss: 0.8648 - accuracy: 0.7039 - val_loss: 0.6660 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs:

```
hs: 2 / Accuracy: 0.8010999083409716
219/219 [=====] - 105s 481ms/step - loss: 0.4662 -
accuracy: 0.8441 - val_loss: 0.5885 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 3 / Accuracy: 0.8166819431714024
219/219 [=====] - 112s 512ms/step - loss: 0.3369 -
accuracy: 0.8879 - val_loss: 0.5733 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 3 / Accuracy: 0.8166819431714024
219/219 [=====] - 133s 607ms/step - loss: 0.8911 -
accuracy: 0.6792 - val_loss: 0.6681 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 2 / Accuracy: 0.8075160403299725
219/219 [=====] - 105s 481ms/step - loss: 0.4928 -
accuracy: 0.8343 - val_loss: 0.5927 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 3 / Accuracy: 0.8148487626031164
219/219 [=====] - 109s 496ms/step - loss: 0.3669 -
accuracy: 0.8762 - val_loss: 0.5682 - val_accuracy: 0.8225
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 110s 502ms/step - loss: 0.2903 -
accuracy: 0.8962 - val_loss: 0.5796 - val_accuracy: 0.8213
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 119s 543ms/step - loss: 0.8929 -
accuracy: 0.6723 - val_loss: 0.6802 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 2 / Accuracy: 0.8065994500458296
219/219 [=====] - 107s 490ms/step - loss: 0.5139 -
accuracy: 0.8277 - val_loss: 0.5923 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 3 / Accuracy: 0.8157653528872594
219/219 [=====] - 107s 487ms/step - loss: 0.3985 -
accuracy: 0.8647 - val_loss: 0.5671 - val_accuracy: 0.8293
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 111s 505ms/step - loss: 0.3266 -
accuracy: 0.8891 - val_loss: 0.5639 - val_accuracy: 0.8293
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 117s 536ms/step - loss: 0.9295 -
accuracy: 0.6912 - val_loss: 0.6879 - val_accuracy: 0.7904
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 2 / Accuracy: 0.8056828597616865
219/219 [=====] - 118s 539ms/step - loss: 0.5570 -
accuracy: 0.8234 - val_loss: 0.5988 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 3 / Accuracy: 0.8102658111824015
219/219 [=====] - 163s 743ms/step - loss: 0.4372 -
accuracy: 0.8490 - val_loss: 0.5709 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 4 / Accuracy: 0.8221814848762603
219/219 [=====] - 163s 745ms/step - loss: 0.3553 -
accuracy: 0.8725 - val_loss: 0.5654 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epoc
hs: 5 / Accuracy: 0.8240146654445463
```

219/219 [=====] - 164s 749ms/step - loss: 0.3075 - accuracy: 0.8954 - val_loss: 0.5764 - val_accuracy: 0.8167
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 5 / Accuracy: 0.8240146654445463

219/219 [=====] - 176s 806ms/step - loss: 0.9434 - accuracy: 0.6658 - val_loss: 0.6978 - val_accuracy: 0.7835
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 2 / Accuracy: 0.7956003666361137

219/219 [=====] - 142s 646ms/step - loss: 0.5959 - accuracy: 0.8094 - val_loss: 0.6099 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 3 / Accuracy: 0.8084326306141155

219/219 [=====] - 107s 488ms/step - loss: 0.4861 - accuracy: 0.8372 - val_loss: 0.5730 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 4 / Accuracy: 0.8130155820348305

219/219 [=====] - 103s 472ms/step - loss: 0.4117 - accuracy: 0.8604 - val_loss: 0.5534 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 5 / Accuracy: 0.8175985334555453

219/219 [=====] - 113s 515ms/step - loss: 0.3676 - accuracy: 0.8750 - val_loss: 0.5688 - val_accuracy: 0.8247
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 6 / Accuracy: 0.8249312557286893

219/219 [=====] - 110s 501ms/step - loss: 0.3211 - accuracy: 0.8891 - val_loss: 0.5769 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 512 / Epochs: 6 / Accuracy: 0.8249312557286893

219/219 [=====] - 117s 533ms/step - loss: 0.8229 - accuracy: 0.6905 - val_loss: 0.6672 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 2 / Accuracy: 0.8001833180568286

219/219 [=====] - 109s 499ms/step - loss: 0.4376 - accuracy: 0.8490 - val_loss: 0.5925 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8139321723189734

219/219 [=====] - 106s 485ms/step - loss: 0.3086 - accuracy: 0.8960 - val_loss: 0.5849 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8139321723189734

219/219 [=====] - 119s 542ms/step - loss: 0.8477 - accuracy: 0.6795 - val_loss: 0.6703 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 2 / Accuracy: 0.8056828597616865

219/219 [=====] - 113s 517ms/step - loss: 0.4664 - accuracy: 0.8392 - val_loss: 0.5913 - val_accuracy: 0.8099
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8203483043079743

219/219 [=====] - 112s 512ms/step - loss: 0.3432 - accuracy: 0.8828 - val_loss: 0.5823 - val_accuracy: 0.8225
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8203483043079743

219/219 [=====] - 110s 502ms/step - loss: 0.8591 - accuracy: 0.6824 - val_loss: 0.6693 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 2 / Accuracy: 0.8065994500458296

219/219 [=====] - 102s 468ms/step - loss: 0.4981 -

accuracy: 0.8303 - val_loss: 0.5875 - val_accuracy: 0.8099
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8194317140238313
219/219 [=====] - 110s 503ms/step - loss: 0.3825 - accuracy: 0.8684 - val_loss: 0.5707 - val_accuracy: 0.8190
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8194317140238313
219/219 [=====] - 123s 562ms/step - loss: 0.8782 - accuracy: 0.6746 - val_loss: 0.6781 - val_accuracy: 0.7938
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 2 / Accuracy: 0.8038496791934006
219/219 [=====] - 110s 503ms/step - loss: 0.5200 - accuracy: 0.8292 - val_loss: 0.5915 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8075160403299725
219/219 [=====] - 107s 488ms/step - loss: 0.4168 - accuracy: 0.8553 - val_loss: 0.5775 - val_accuracy: 0.8121
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 4 / Accuracy: 0.8166819431714024
219/219 [=====] - 115s 526ms/step - loss: 0.3436 - accuracy: 0.8811 - val_loss: 0.5619 - val_accuracy: 0.8179
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 5 / Accuracy: 0.8249312557286893
219/219 [=====] - 110s 504ms/step - loss: 0.2950 - accuracy: 0.8997 - val_loss: 0.5816 - val_accuracy: 0.8259
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 5 / Accuracy: 0.8249312557286893
219/219 [=====] - 112s 512ms/step - loss: 0.9083 - accuracy: 0.6788 - val_loss: 0.6942 - val_accuracy: 0.7858
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 2 / Accuracy: 0.8029330889092575
219/219 [=====] - 105s 480ms/step - loss: 0.5705 - accuracy: 0.8137 - val_loss: 0.6056 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8157653528872594
219/219 [=====] - 110s 504ms/step - loss: 0.4644 - accuracy: 0.8389 - val_loss: 0.5895 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 720 / Epochs: 3 / Accuracy: 0.8157653528872594
219/219 [=====] - 117s 532ms/step - loss: 0.7789 - accuracy: 0.7154 - val_loss: 0.6659 - val_accuracy: 0.7938
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 2 / Accuracy: 0.7965169569202566
219/219 [=====] - 108s 494ms/step - loss: 0.4144 - accuracy: 0.8570 - val_loss: 0.5992 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8102658111824015
219/219 [=====] - 108s 495ms/step - loss: 0.2859 - accuracy: 0.8988 - val_loss: 0.5998 - val_accuracy: 0.8213
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8102658111824015
219/219 [=====] - 120s 546ms/step - loss: 0.8041 - accuracy: 0.7125 - val_loss: 0.6597 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 2 / Accuracy: 0.8020164986251146
219/219 [=====] - 163s 746ms/step - loss: 0.4388 - accuracy: 0.8510 - val_loss: 0.5898 - val_accuracy: 0.8156

Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8111824014665444
219/219 [=====] - 162s 742ms/step - loss: 0.3152 - accuracy: 0.8928 - val_loss: 0.5936 - val_accuracy: 0.8190
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8111824014665444
219/219 [=====] - 179s 817ms/step - loss: 0.8225 - accuracy: 0.6793 - val_loss: 0.6692 - val_accuracy: 0.7904
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 2 / Accuracy: 0.7983501374885427
219/219 [=====] - 168s 766ms/step - loss: 0.4745 - accuracy: 0.8412 - val_loss: 0.5911 - val_accuracy: 0.8121
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8166819431714024
219/219 [=====] - 165s 755ms/step - loss: 0.3595 - accuracy: 0.8736 - val_loss: 0.5857 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 4 / Accuracy: 0.8185151237396884
219/219 [=====] - 165s 752ms/step - loss: 0.2914 - accuracy: 0.8942 - val_loss: 0.5951 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 4 / Accuracy: 0.8185151237396884
219/219 [=====] - 177s 810ms/step - loss: 0.8577 - accuracy: 0.6872 - val_loss: 0.6650 - val_accuracy: 0.7938
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 2 / Accuracy: 0.8038496791934006
219/219 [=====] - 165s 753ms/step - loss: 0.5013 - accuracy: 0.8381 - val_loss: 0.6033 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8212648945921174
219/219 [=====] - 166s 756ms/step - loss: 0.3958 - accuracy: 0.8659 - val_loss: 0.5771 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8212648945921174
219/219 [=====] - 178s 812ms/step - loss: 0.8818 - accuracy: 0.6832 - val_loss: 0.6928 - val_accuracy: 0.7881
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 2 / Accuracy: 0.7974335472043996
219/219 [=====] - 166s 757ms/step - loss: 0.5533 - accuracy: 0.8186 - val_loss: 0.5941 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8185151237396884
219/219 [=====] - 167s 761ms/step - loss: 0.4400 - accuracy: 0.8490 - val_loss: 0.5622 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 4 / Accuracy: 0.8212648945921174
219/219 [=====] - 165s 753ms/step - loss: 0.3727 - accuracy: 0.8719 - val_loss: 0.5656 - val_accuracy: 0.8213
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 5 / Accuracy: 0.8221814848762603
219/219 [=====] - 166s 758ms/step - loss: 0.3280 - accuracy: 0.8819 - val_loss: 0.5806 - val_accuracy: 0.8156
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 6 / Accuracy: 0.8240146654445463
219/219 [=====] - 165s 755ms/step - loss: 0.2985 - accuracy: 0.8919 - val_loss: 0.5777 - val_accuracy: 0.8225
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs:

chs: 7 / Accuracy: 0.8249312557286893
219/219 [=====] - 165s 752ms/step - loss: 0.2831 - accuracy: 0.8931 - val_loss: 0.5938 - val_accuracy: 0.8167
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 8 / Accuracy: 0.8258478460128322
219/219 [=====] - 164s 750ms/step - loss: 0.2492 - accuracy: 0.9048 - val_loss: 0.6153 - val_accuracy: 0.8213
Activation: sigmoid / Optimizer: adam / Batch size: 16 / Layers: 1024 / Epochs: 8 / Accuracy: 0.8258478460128322
110/110 [=====] - 99s 904ms/step - loss: 1.2485 - accuracy: 0.6611 - val_loss: 0.8059 - val_accuracy: 0.7526
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 2 / Accuracy: 0.7552703941338221
110/110 [=====] - 91s 826ms/step - loss: 0.6639 - accuracy: 0.7925 - val_loss: 0.6995 - val_accuracy: 0.7869
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 3 / Accuracy: 0.7937671860678277
110/110 [=====] - 86s 780ms/step - loss: 0.5181 - accuracy: 0.8257 - val_loss: 0.6490 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 4 / Accuracy: 0.7992667277726856
110/110 [=====] - 90s 820ms/step - loss: 0.4221 - accuracy: 0.8619 - val_loss: 0.6219 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8102658111824015
110/110 [=====] - 85s 774ms/step - loss: 0.3540 - accuracy: 0.8813 - val_loss: 0.6080 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 6 / Accuracy: 0.8111824014665444
110/110 [=====] - 90s 816ms/step - loss: 0.3022 - accuracy: 0.9028 - val_loss: 0.6034 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 7 / Accuracy: 0.8166819431714024
110/110 [=====] - 87s 787ms/step - loss: 0.2611 - accuracy: 0.9146 - val_loss: 0.6053 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 7 / Accuracy: 0.8166819431714024
110/110 [=====] - 107s 975ms/step - loss: 1.2720 - accuracy: 0.6324 - val_loss: 0.8071 - val_accuracy: 0.7342
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 2 / Accuracy: 0.7406049495875344
110/110 [=====] - 89s 806ms/step - loss: 0.6746 - accuracy: 0.7845 - val_loss: 0.7089 - val_accuracy: 0.7812
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 3 / Accuracy: 0.7873510540788268
110/110 [=====] - 81s 739ms/step - loss: 0.5378 - accuracy: 0.8234 - val_loss: 0.6540 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 4 / Accuracy: 0.8029330889092575
110/110 [=====] - 58s 528ms/step - loss: 0.4452 - accuracy: 0.8538 - val_loss: 0.6171 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8139321723189734
110/110 [=====] - 62s 561ms/step - loss: 0.3784 - accuracy: 0.8719 - val_loss: 0.6025 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8139321723189734

110/110 [=====] - 72s 655ms/step - loss: 1.3070 - accuracy: 0.5653 - val_loss: 0.8436 - val_accuracy: 0.7365
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 2 / Accuracy: 0.7286892758936755

110/110 [=====] - 66s 598ms/step - loss: 0.7140 - accuracy: 0.7767 - val_loss: 0.7140 - val_accuracy: 0.7801
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 3 / Accuracy: 0.7873510540788268

110/110 [=====] - 62s 564ms/step - loss: 0.5737 - accuracy: 0.8140 - val_loss: 0.6533 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 4 / Accuracy: 0.7992667277726856

110/110 [=====] - 66s 604ms/step - loss: 0.4814 - accuracy: 0.8381 - val_loss: 0.6131 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8120989917506874

110/110 [=====] - 58s 524ms/step - loss: 0.4149 - accuracy: 0.8590 - val_loss: 0.5968 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 6 / Accuracy: 0.8148487626031164

110/110 [=====] - 69s 631ms/step - loss: 0.3737 - accuracy: 0.8716 - val_loss: 0.5935 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 7 / Accuracy: 0.8175985334555453

110/110 [=====] - 60s 544ms/step - loss: 0.3298 - accuracy: 0.8879 - val_loss: 0.5748 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 8 / Accuracy: 0.8240146654445463

110/110 [=====] - 55s 500ms/step - loss: 0.2966 - accuracy: 0.9005 - val_loss: 0.5863 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 9 / Accuracy: 0.8249312557286893

110/110 [=====] - 65s 589ms/step - loss: 0.2767 - accuracy: 0.9063 - val_loss: 0.5809 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 9 / Accuracy: 0.8249312557286893

110/110 [=====] - 65s 591ms/step - loss: 1.3478 - accuracy: 0.5702 - val_loss: 0.8251 - val_accuracy: 0.7308
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 2 / Accuracy: 0.7314390467461045

110/110 [=====] - 59s 537ms/step - loss: 0.7285 - accuracy: 0.7782 - val_loss: 0.7099 - val_accuracy: 0.7812
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 3 / Accuracy: 0.7864344637946837

110/110 [=====] - 59s 535ms/step - loss: 0.6047 - accuracy: 0.8088 - val_loss: 0.6557 - val_accuracy: 0.7915
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 4 / Accuracy: 0.8020164986251146

110/110 [=====] - 65s 587ms/step - loss: 0.5162 - accuracy: 0.8315 - val_loss: 0.6197 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8084326306141155

110/110 [=====] - 61s 552ms/step - loss: 0.4565 - accuracy: 0.8515 - val_loss: 0.5910 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 6 / Accuracy: 0.8185151237396884

110/110 [=====] - 62s 563ms/step - loss: 0.4034 -

accuracy: 0.8627 - val_loss: 0.6027 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 6 / Accuracy: 0.8185151237396884
110/110 [=====] - 62s 567ms/step - loss: 1.3762 - accuracy: 0.5464 - val_loss: 0.8349 - val_accuracy: 0.7308
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 2 / Accuracy: 0.7296058661778185
110/110 [=====] - 54s 491ms/step - loss: 0.7729 - accuracy: 0.7664 - val_loss: 0.7297 - val_accuracy: 0.7755
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 3 / Accuracy: 0.7827681026581118
110/110 [=====] - 55s 500ms/step - loss: 0.6469 - accuracy: 0.7985 - val_loss: 0.6651 - val_accuracy: 0.7881
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 4 / Accuracy: 0.8029330889092575
110/110 [=====] - 56s 508ms/step - loss: 0.5529 - accuracy: 0.8189 - val_loss: 0.6276 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8120989917506874
110/110 [=====] - 60s 542ms/step - loss: 0.4977 - accuracy: 0.8366 - val_loss: 0.6097 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 128 / Epochs: 5 / Accuracy: 0.8120989917506874
110/110 [=====] - 61s 556ms/step - loss: 1.1227 - accuracy: 0.6504 - val_loss: 0.7583 - val_accuracy: 0.7709
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 2 / Accuracy: 0.7772685609532539
110/110 [=====] - 57s 521ms/step - loss: 0.5930 - accuracy: 0.8071 - val_loss: 0.6689 - val_accuracy: 0.7950
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 3 / Accuracy: 0.8010999083409716
110/110 [=====] - 56s 505ms/step - loss: 0.4579 - accuracy: 0.8467 - val_loss: 0.6256 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 4 / Accuracy: 0.8102658111824015
110/110 [=====] - 56s 508ms/step - loss: 0.3681 - accuracy: 0.8756 - val_loss: 0.6067 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8120989917506874
110/110 [=====] - 61s 556ms/step - loss: 0.3025 - accuracy: 0.9008 - val_loss: 0.6047 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 6 / Accuracy: 0.8166819431714024
110/110 [=====] - 55s 504ms/step - loss: 0.2518 - accuracy: 0.9235 - val_loss: 0.6129 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 6 / Accuracy: 0.8166819431714024
110/110 [=====] - 72s 657ms/step - loss: 1.1100 - accuracy: 0.6583 - val_loss: 0.7670 - val_accuracy: 0.7663
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 2 / Accuracy: 0.7745187901008249
110/110 [=====] - 59s 537ms/step - loss: 0.6119 - accuracy: 0.8011 - val_loss: 0.6757 - val_accuracy: 0.7950
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 3 / Accuracy: 0.7965169569202566
110/110 [=====] - 62s 566ms/step - loss: 0.4817 - accuracy: 0.8392 - val_loss: 0.6361 - val_accuracy: 0.7973

Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 4 / Accuracy: 0.8047662694775435
110/110 [=====] - 55s 503ms/step - loss: 0.4008 - accuracy: 0.8636 - val_loss: 0.5958 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8185151237396884
110/110 [=====] - 57s 516ms/step - loss: 0.3349 - accuracy: 0.8885 - val_loss: 0.5980 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8185151237396884
110/110 [=====] - 77s 701ms/step - loss: 1.1255 - accuracy: 0.6401 - val_loss: 0.7792 - val_accuracy: 0.7652
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 2 / Accuracy: 0.770852428964253
110/110 [=====] - 57s 519ms/step - loss: 0.6388 - accuracy: 0.7959 - val_loss: 0.6823 - val_accuracy: 0.7904
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 3 / Accuracy: 0.7946837763519706
110/110 [=====] - 65s 594ms/step - loss: 0.5112 - accuracy: 0.8303 - val_loss: 0.6337 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 4 / Accuracy: 0.8111824014665444
110/110 [=====] - 58s 525ms/step - loss: 0.4313 - accuracy: 0.8561 - val_loss: 0.5858 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8230980751604033
110/110 [=====] - 62s 560ms/step - loss: 0.3722 - accuracy: 0.8702 - val_loss: 0.5848 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8230980751604033
110/110 [=====] - 69s 630ms/step - loss: 1.1572 - accuracy: 0.6168 - val_loss: 0.7896 - val_accuracy: 0.7572
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 2 / Accuracy: 0.768102658111824
110/110 [=====] - 55s 497ms/step - loss: 0.6817 - accuracy: 0.7893 - val_loss: 0.6926 - val_accuracy: 0.7847
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 3 / Accuracy: 0.7937671860678277
110/110 [=====] - 59s 533ms/step - loss: 0.5481 - accuracy: 0.8212 - val_loss: 0.6232 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 4 / Accuracy: 0.8093492208982584
110/110 [=====] - 59s 533ms/step - loss: 0.4643 - accuracy: 0.8498 - val_loss: 0.5963 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8230980751604033
110/110 [=====] - 55s 503ms/step - loss: 0.4061 - accuracy: 0.8581 - val_loss: 0.5834 - val_accuracy: 0.8121
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 5 / Accuracy: 0.8230980751604033
110/110 [=====] - 68s 620ms/step - loss: 1.1493 - accuracy: 0.6165 - val_loss: 0.7970 - val_accuracy: 0.7549
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs: 2 / Accuracy: 0.7543538038496792
110/110 [=====] - 56s 509ms/step - loss: 0.7140 - accuracy: 0.7822 - val_loss: 0.6843 - val_accuracy: 0.7892
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs:

hs: 3 / Accuracy: 0.8020164986251146
110/110 [=====] - 59s 538ms/step - loss: 0.6010 -
accuracy: 0.8085 - val_loss: 0.6288 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs:
hs: 4 / Accuracy: 0.8148487626031164
110/110 [=====] - 55s 502ms/step - loss: 0.5181 -
accuracy: 0.8323 - val_loss: 0.5872 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs:
hs: 5 / Accuracy: 0.8240146654445463
110/110 [=====] - 56s 505ms/step - loss: 0.4462 -
accuracy: 0.8518 - val_loss: 0.6012 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 256 / Epochs:
hs: 5 / Accuracy: 0.8240146654445463
110/110 [=====] - 70s 639ms/step - loss: 0.9763 -
accuracy: 0.6511 - val_loss: 0.7382 - val_accuracy: 0.7778
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 2 / Accuracy: 0.7855178735105408
110/110 [=====] - 57s 519ms/step - loss: 0.5382 -
accuracy: 0.8177 - val_loss: 0.6467 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 3 / Accuracy: 0.8093492208982584
110/110 [=====] - 61s 556ms/step - loss: 0.4002 -
accuracy: 0.8639 - val_loss: 0.6060 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 4 / Accuracy: 0.8111824014665444
110/110 [=====] - 58s 525ms/step - loss: 0.3118 -
accuracy: 0.8977 - val_loss: 0.6002 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 5 / Accuracy: 0.8148487626031164
110/110 [=====] - 57s 523ms/step - loss: 0.2476 -
accuracy: 0.9249 - val_loss: 0.6114 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 6 / Accuracy: 0.8166819431714024
110/110 [=====] - 61s 555ms/step - loss: 0.1985 -
accuracy: 0.9401 - val_loss: 0.6360 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 6 / Accuracy: 0.8166819431714024
110/110 [=====] - 67s 613ms/step - loss: 0.9869 -
accuracy: 0.6588 - val_loss: 0.7361 - val_accuracy: 0.7778
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 2 / Accuracy: 0.7836846929422548
110/110 [=====] - 58s 529ms/step - loss: 0.5564 -
accuracy: 0.8154 - val_loss: 0.6475 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 3 / Accuracy: 0.8038496791934006
110/110 [=====] - 59s 532ms/step - loss: 0.4298 -
accuracy: 0.8550 - val_loss: 0.5890 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 4 / Accuracy: 0.8185151237396884
110/110 [=====] - 59s 533ms/step - loss: 0.3432 -
accuracy: 0.8868 - val_loss: 0.5783 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 5 / Accuracy: 0.8203483043079743
110/110 [=====] - 59s 540ms/step - loss: 0.2817 -
accuracy: 0.9057 - val_loss: 0.5913 - val_accuracy: 0.8099
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs:
hs: 5 / Accuracy: 0.8203483043079743

110/110 [=====] - 68s 617ms/step - loss: 1.0250 - accuracy: 0.6123 - val_loss: 0.7546 - val_accuracy: 0.7709
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 2 / Accuracy: 0.7827681026581118

110/110 [=====] - 58s 531ms/step - loss: 0.5876 - accuracy: 0.8091 - val_loss: 0.6568 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 3 / Accuracy: 0.8075160403299725

110/110 [=====] - 62s 567ms/step - loss: 0.4723 - accuracy: 0.8404 - val_loss: 0.5940 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 4 / Accuracy: 0.8230980751604033

110/110 [=====] - 60s 544ms/step - loss: 0.3771 - accuracy: 0.8719 - val_loss: 0.5783 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 4 / Accuracy: 0.8230980751604033

110/110 [=====] - 67s 611ms/step - loss: 1.0031 - accuracy: 0.6812 - val_loss: 0.7562 - val_accuracy: 0.7675
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 2 / Accuracy: 0.7809349220898258

110/110 [=====] - 64s 578ms/step - loss: 0.6342 - accuracy: 0.7934 - val_loss: 0.6783 - val_accuracy: 0.7950
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 3 / Accuracy: 0.8020164986251146

110/110 [=====] - 58s 525ms/step - loss: 0.5069 - accuracy: 0.8349 - val_loss: 0.6118 - val_accuracy: 0.7950
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 4 / Accuracy: 0.8157653528872594

110/110 [=====] - 59s 532ms/step - loss: 0.4221 - accuracy: 0.8544 - val_loss: 0.5848 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 5 / Accuracy: 0.8258478460128322

110/110 [=====] - 94s 854ms/step - loss: 0.3611 - accuracy: 0.8719 - val_loss: 0.5911 - val_accuracy: 0.8121
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 5 / Accuracy: 0.8258478460128322

110/110 [=====] - 101s 914ms/step - loss: 1.0748 - accuracy: 0.6076 - val_loss: 0.7615 - val_accuracy: 0.7663
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 2 / Accuracy: 0.7781851512373968

110/110 [=====] - 94s 858ms/step - loss: 0.6670 - accuracy: 0.7908 - val_loss: 0.6649 - val_accuracy: 0.7869
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 3 / Accuracy: 0.8038496791934006

110/110 [=====] - 87s 787ms/step - loss: 0.5462 - accuracy: 0.8212 - val_loss: 0.6050 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 4 / Accuracy: 0.8194317140238313

110/110 [=====] - 87s 787ms/step - loss: 0.4715 - accuracy: 0.8404 - val_loss: 0.5807 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 5 / Accuracy: 0.8230980751604033

110/110 [=====] - 87s 795ms/step - loss: 0.4120 - accuracy: 0.8621 - val_loss: 0.5673 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 6 / Accuracy: 0.8240146654445463

110/110 [=====] - 87s 790ms/step - loss: 0.3747 -

accuracy: 0.8739 - val_loss: 0.5553 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 7 / Accuracy: 0.8276810265811182
110/110 [=====] - 88s 796ms/step - loss: 0.3340 - accuracy: 0.8854 - val_loss: 0.5902 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 512 / Epochs: 7 / Accuracy: 0.8276810265811182
110/110 [=====] - 104s 948ms/step - loss: 0.9202 - accuracy: 0.6678 - val_loss: 0.7430 - val_accuracy: 0.7801
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 2 / Accuracy: 0.7946837763519706
110/110 [=====] - 83s 757ms/step - loss: 0.5050 - accuracy: 0.8263 - val_loss: 0.6384 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 3 / Accuracy: 0.8075160403299725
110/110 [=====] - 83s 757ms/step - loss: 0.3692 - accuracy: 0.8710 - val_loss: 0.6099 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8185151237396884
110/110 [=====] - 91s 830ms/step - loss: 0.2819 - accuracy: 0.9066 - val_loss: 0.6117 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8185151237396884
110/110 [=====] - 103s 932ms/step - loss: 0.9506 - accuracy: 0.6492 - val_loss: 0.7371 - val_accuracy: 0.7801
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 2 / Accuracy: 0.7846012832263978
110/110 [=====] - 87s 793ms/step - loss: 0.5421 - accuracy: 0.8151 - val_loss: 0.6374 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 3 / Accuracy: 0.8102658111824015
110/110 [=====] - 92s 833ms/step - loss: 0.4067 - accuracy: 0.8590 - val_loss: 0.6040 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8148487626031164
110/110 [=====] - 86s 781ms/step - loss: 0.3276 - accuracy: 0.8874 - val_loss: 0.5917 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8148487626031164
110/110 [=====] - 98s 891ms/step - loss: 0.9553 - accuracy: 0.6501 - val_loss: 0.7306 - val_accuracy: 0.7812
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 2 / Accuracy: 0.7864344637946837
110/110 [=====] - 63s 575ms/step - loss: 0.5622 - accuracy: 0.8111 - val_loss: 0.6303 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 3 / Accuracy: 0.8075160403299725
110/110 [=====] - 57s 523ms/step - loss: 0.4434 - accuracy: 0.8504 - val_loss: 0.5842 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8249312557286893
110/110 [=====] - 59s 535ms/step - loss: 0.3592 - accuracy: 0.8770 - val_loss: 0.5745 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8249312557286893
110/110 [=====] - 79s 718ms/step - loss: 0.9827 - accuracy: 0.6494 - val_loss: 0.7531 - val_accuracy: 0.7721

Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 2 / Accuracy: 0.7827681026581118
110/110 [=====] - 56s 507ms/step - loss: 0.6152 - accuracy: 0.8017 - val_loss: 0.6560 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 3 / Accuracy: 0.8065994500458296
110/110 [=====] - 57s 517ms/step - loss: 0.4947 - accuracy: 0.8283 - val_loss: 0.6105 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8166819431714024
110/110 [=====] - 63s 571ms/step - loss: 0.4024 - accuracy: 0.8584 - val_loss: 0.5883 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 5 / Accuracy: 0.8175985334555453
110/110 [=====] - 57s 519ms/step - loss: 0.3455 - accuracy: 0.8819 - val_loss: 0.5692 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 5 / Accuracy: 0.8175985334555453
110/110 [=====] - 67s 608ms/step - loss: 1.0151 - accuracy: 0.6317 - val_loss: 0.7562 - val_accuracy: 0.7698
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 2 / Accuracy: 0.7791017415215399
110/110 [=====] - 57s 514ms/step - loss: 0.6415 - accuracy: 0.7945 - val_loss: 0.6665 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 3 / Accuracy: 0.8010999083409716
110/110 [=====] - 59s 533ms/step - loss: 0.5364 - accuracy: 0.8255 - val_loss: 0.5913 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 4 / Accuracy: 0.8194317140238313
110/110 [=====] - 58s 527ms/step - loss: 0.4543 - accuracy: 0.8424 - val_loss: 0.5673 - val_accuracy: 0.8099
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 5 / Accuracy: 0.8304307974335472
110/110 [=====] - 58s 524ms/step - loss: 0.4011 - accuracy: 0.8584 - val_loss: 0.5626 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 720 / Epochs: 5 / Accuracy: 0.8304307974335472
110/110 [=====] - 72s 651ms/step - loss: 0.8753 - accuracy: 0.6840 - val_loss: 0.7396 - val_accuracy: 0.7801
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 2 / Accuracy: 0.7956003666361137
110/110 [=====] - 76s 693ms/step - loss: 0.4852 - accuracy: 0.8320 - val_loss: 0.6293 - val_accuracy: 0.7938
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 3 / Accuracy: 0.8102658111824015
110/110 [=====] - 87s 795ms/step - loss: 0.3468 - accuracy: 0.8802 - val_loss: 0.6123 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 4 / Accuracy: 0.8148487626031164
110/110 [=====] - 93s 849ms/step - loss: 0.2587 - accuracy: 0.9183 - val_loss: 0.6197 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 4 / Accuracy: 0.8148487626031164
110/110 [=====] - 105s 958ms/step - loss: 0.9056 - accuracy: 0.6597 - val_loss: 0.7460 - val_accuracy: 0.7812
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 4 / Accuracy: 0.8148487626031164

```
chs: 2 / Accuracy: 0.7873510540788268
110/110 [=====] - 92s 840ms/step - loss: 0.5165 -
accuracy: 0.8234 - val_loss: 0.6235 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 3 / Accuracy: 0.8203483043079743
110/110 [=====] - 102s 925ms/step - loss: 0.3856 -
accuracy: 0.8676 - val_loss: 0.5978 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8258478460128322
110/110 [=====] - 90s 821ms/step - loss: 0.2980 -
accuracy: 0.9017 - val_loss: 0.6014 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8258478460128322
110/110 [=====] - 106s 960ms/step - loss: 0.9330 -
accuracy: 0.6498 - val_loss: 0.7372 - val_accuracy: 0.7812
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 2 / Accuracy: 0.7882676443629697
110/110 [=====] - 97s 879ms/step - loss: 0.5519 -
accuracy: 0.8157 - val_loss: 0.6358 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 3 / Accuracy: 0.8148487626031164
110/110 [=====] - 91s 826ms/step - loss: 0.4225 -
accuracy: 0.8619 - val_loss: 0.5917 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8212648945921174
110/110 [=====] - 90s 821ms/step - loss: 0.3457 -
accuracy: 0.8788 - val_loss: 0.6000 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8212648945921174
110/110 [=====] - 108s 982ms/step - loss: 0.9467 -
accuracy: 0.6552 - val_loss: 0.7316 - val_accuracy: 0.7778
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 2 / Accuracy: 0.7919340054995417
110/110 [=====] - 99s 899ms/step - loss: 0.5843 -
accuracy: 0.8031 - val_loss: 0.6326 - val_accuracy: 0.7995
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 3 / Accuracy: 0.8185151237396884
110/110 [=====] - 91s 825ms/step - loss: 0.4615 -
accuracy: 0.8381 - val_loss: 0.5974 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8203483043079743
110/110 [=====] - 91s 825ms/step - loss: 0.3785 -
accuracy: 0.8676 - val_loss: 0.6104 - val_accuracy: 0.8018
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8203483043079743
110/110 [=====] - 118s 1s/step - loss: 0.9730 - ac
curacy: 0.6588 - val_loss: 0.7559 - val_accuracy: 0.7755
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 2 / Accuracy: 0.7846012832263978
110/110 [=====] - 91s 830ms/step - loss: 0.6249 -
accuracy: 0.7982 - val_loss: 0.6500 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 3 / Accuracy: 0.8130155820348305
110/110 [=====] - 91s 824ms/step - loss: 0.5000 -
accuracy: 0.8309 - val_loss: 0.6153 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epo
chs: 4 / Accuracy: 0.8166819431714024
```

110/110 [=====] - 92s 838ms/step - loss: 0.4271 - accuracy: 0.8561 - val_loss: 0.5778 - val_accuracy: 0.8156
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 5 / Accuracy: 0.8221814848762603

110/110 [=====] - 98s 893ms/step - loss: 0.3794 - accuracy: 0.8664 - val_loss: 0.5549 - val_accuracy: 0.8305
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 6 / Accuracy: 0.8267644362969753

110/110 [=====] - 92s 833ms/step - loss: 0.3288 - accuracy: 0.8828 - val_loss: 0.5858 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 32 / Layers: 1024 / Epochs: 6 / Accuracy: 0.8267644362969753

55/55 [=====] - 65s 1s/step - loss: 1.5330 - accuracy: 0.6128 - val_loss: 0.9607 - val_accuracy: 0.7079
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 2 / Accuracy: 0.7140238313473877

55/55 [=====] - 46s 839ms/step - loss: 0.7627 - accuracy: 0.7501 - val_loss: 0.7653 - val_accuracy: 0.7652
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 3 / Accuracy: 0.7809349220898258

55/55 [=====] - 47s 848ms/step - loss: 0.5977 - accuracy: 0.7994 - val_loss: 0.7030 - val_accuracy: 0.7938
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 4 / Accuracy: 0.7901008249312558

55/55 [=====] - 58s 1s/step - loss: 0.5031 - accuracy: 0.8286 - val_loss: 0.6575 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 5 / Accuracy: 0.8065994500458296

55/55 [=====] - 46s 837ms/step - loss: 0.4322 - accuracy: 0.8535 - val_loss: 0.6260 - val_accuracy: 0.8076
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 6 / Accuracy: 0.8130155820348305

55/55 [=====] - 46s 838ms/step - loss: 0.3770 - accuracy: 0.8736 - val_loss: 0.6087 - val_accuracy: 0.8099
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 7 / Accuracy: 0.8203483043079743

55/55 [=====] - 49s 898ms/step - loss: 0.3328 - accuracy: 0.8914 - val_loss: 0.5987 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 8 / Accuracy: 0.8212648945921174

55/55 [=====] - 46s 831ms/step - loss: 0.2959 - accuracy: 0.9037 - val_loss: 0.5933 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 8 / Accuracy: 0.8212648945921174

55/55 [=====] - 66s 1s/step - loss: 1.5714 - accuracy: 0.5375 - val_loss: 0.9799 - val_accuracy: 0.7079
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 2 / Accuracy: 0.7066911090742438

55/55 [=====] - 48s 874ms/step - loss: 0.7880 - accuracy: 0.7418 - val_loss: 0.7695 - val_accuracy: 0.7652
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 3 / Accuracy: 0.76993583868011

55/55 [=====] - 55s 992ms/step - loss: 0.6163 - accuracy: 0.7936 - val_loss: 0.7044 - val_accuracy: 0.7892
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 4 / Accuracy: 0.7873510540788268

55/55 [=====] - 48s 864ms/step - loss: 0.5329 - ac

curacy: 0.8252 - val_loss: 0.6588 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 5 / Accuracy: 0.8038496791934006
55/55 [=====] - 47s 858ms/step - loss: 0.4637 - accuracy: 0.8415 - val_loss: 0.6278 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 6 / Accuracy: 0.8130155820348305
55/55 [=====] - 49s 890ms/step - loss: 0.4105 - accuracy: 0.8650 - val_loss: 0.6146 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 7 / Accuracy: 0.8185151237396884
55/55 [=====] - 48s 867ms/step - loss: 0.3654 - accuracy: 0.8822 - val_loss: 0.6017 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 8 / Accuracy: 0.8212648945921174
55/55 [=====] - 47s 856ms/step - loss: 0.3308 - accuracy: 0.8891 - val_loss: 0.5909 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 9 / Accuracy: 0.8230980751604033
55/55 [=====] - 47s 850ms/step - loss: 0.3012 - accuracy: 0.9028 - val_loss: 0.5892 - val_accuracy: 0.8179
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 9 / Accuracy: 0.8230980751604033
55/55 [=====] - 76s 1s/step - loss: 1.5795 - accuracy: 0.5222 - val_loss: 0.9885 - val_accuracy: 0.6415
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 2 / Accuracy: 0.6544454628780935
55/55 [=====] - 48s 864ms/step - loss: 0.8396 - accuracy: 0.7137 - val_loss: 0.7952 - val_accuracy: 0.7503
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 3 / Accuracy: 0.7598533455545371
55/55 [=====] - 49s 886ms/step - loss: 0.6702 - accuracy: 0.7810 - val_loss: 0.7247 - val_accuracy: 0.7847
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 4 / Accuracy: 0.7873510540788268
55/55 [=====] - 48s 864ms/step - loss: 0.5801 - accuracy: 0.8077 - val_loss: 0.6762 - val_accuracy: 0.7938
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 5 / Accuracy: 0.7992667277726856
55/55 [=====] - 47s 862ms/step - loss: 0.5097 - accuracy: 0.8255 - val_loss: 0.6386 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 6 / Accuracy: 0.8075160403299725
55/55 [=====] - 47s 857ms/step - loss: 0.4550 - accuracy: 0.8521 - val_loss: 0.6078 - val_accuracy: 0.8053
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 7 / Accuracy: 0.8157653528872594
55/55 [=====] - 52s 945ms/step - loss: 0.4030 - accuracy: 0.8624 - val_loss: 0.5950 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 8 / Accuracy: 0.8221814848762603
55/55 [=====] - 47s 858ms/step - loss: 0.3709 - accuracy: 0.8779 - val_loss: 0.5899 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 9 / Accuracy: 0.8267644362969753
55/55 [=====] - 55s 998ms/step - loss: 0.3405 - accuracy: 0.8874 - val_loss: 0.5848 - val_accuracy: 0.8156

Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 9 / Accuracy: 0.8267644362969753
55/55 [=====] - 73s 1s/step - loss: 1.5338 - accuracy: 0.5002 - val_loss: 0.9716 - val_accuracy: 0.6403
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 2 / Accuracy: 0.6471127406049496
55/55 [=====] - 48s 875ms/step - loss: 0.8369 - accuracy: 0.7226 - val_loss: 0.7903 - val_accuracy: 0.7549
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 3 / Accuracy: 0.7589367552703942
55/55 [=====] - 50s 901ms/step - loss: 0.6854 - accuracy: 0.7810 - val_loss: 0.7257 - val_accuracy: 0.7904
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 4 / Accuracy: 0.7873510540788268
55/55 [=====] - 47s 862ms/step - loss: 0.6038 - accuracy: 0.8045 - val_loss: 0.6730 - val_accuracy: 0.7973
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 5 / Accuracy: 0.7965169569202566
55/55 [=====] - 47s 862ms/step - loss: 0.5408 - accuracy: 0.8183 - val_loss: 0.6377 - val_accuracy: 0.8007
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 6 / Accuracy: 0.8084326306141155
55/55 [=====] - 49s 890ms/step - loss: 0.4839 - accuracy: 0.8449 - val_loss: 0.6135 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 7 / Accuracy: 0.8166819431714024
55/55 [=====] - 57s 1s/step - loss: 0.4427 - accuracy: 0.8518 - val_loss: 0.5989 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 8 / Accuracy: 0.8203483043079743
55/55 [=====] - 48s 877ms/step - loss: 0.3987 - accuracy: 0.8653 - val_loss: 0.5916 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 9 / Accuracy: 0.8230980751604033
55/55 [=====] - 48s 871ms/step - loss: 0.3771 - accuracy: 0.8753 - val_loss: 0.5880 - val_accuracy: 0.8110
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 10 / Accuracy: 0.8249312557286893
55/55 [=====] - 47s 857ms/step - loss: 0.3618 - accuracy: 0.8793 - val_loss: 0.5729 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 11 / Accuracy: 0.8258478460128322
55/55 [=====] - 47s 856ms/step - loss: 0.3251 - accuracy: 0.8908 - val_loss: 0.5814 - val_accuracy: 0.8167
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 11 / Accuracy: 0.8258478460128322
55/55 [=====] - 70s 1s/step - loss: 1.5623 - accuracy: 0.5073 - val_loss: 1.0034 - val_accuracy: 0.5808
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 2 / Accuracy: 0.5802016498625114
55/55 [=====] - 49s 900ms/step - loss: 0.9003 - accuracy: 0.6758 - val_loss: 0.8258 - val_accuracy: 0.7274
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 3 / Accuracy: 0.7296058661778185
55/55 [=====] - 58s 1s/step - loss: 0.7486 - accuracy: 0.7607 - val_loss: 0.7452 - val_accuracy: 0.7686
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs:

hs: 4 / Accuracy: 0.7800183318056828
55/55 [=====] - 47s 862ms/step - loss: 0.6503 - accuracy: 0.7902 - val_loss: 0.6888 - val_accuracy: 0.7927
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 5 / Accuracy: 0.7956003666361137
55/55 [=====] - 47s 861ms/step - loss: 0.5746 - accuracy: 0.8111 - val_loss: 0.6461 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 6 / Accuracy: 0.8056828597616865
55/55 [=====] - 48s 869ms/step - loss: 0.5356 - accuracy: 0.8214 - val_loss: 0.6168 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 7 / Accuracy: 0.8166819431714024
55/55 [=====] - 47s 857ms/step - loss: 0.4995 - accuracy: 0.8395 - val_loss: 0.5992 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 8 / Accuracy: 0.8203483043079743
55/55 [=====] - 47s 863ms/step - loss: 0.4530 - accuracy: 0.8490 - val_loss: 0.5853 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 9 / Accuracy: 0.8249312557286893
55/55 [=====] - 47s 862ms/step - loss: 0.4243 - accuracy: 0.8512 - val_loss: 0.5794 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 10 / Accuracy: 0.8276810265811182
55/55 [=====] - 47s 858ms/step - loss: 0.3881 - accuracy: 0.8710 - val_loss: 0.5712 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 11 / Accuracy: 0.8340971585701191
55/55 [=====] - 54s 990ms/step - loss: 0.3780 - accuracy: 0.8702 - val_loss: 0.5645 - val_accuracy: 0.8202
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 128 / Epochs: 11 / Accuracy: 0.8340971585701191
55/55 [=====] - 70s 1s/step - loss: 1.2890 - accuracy: 0.6620 - val_loss: 0.8461 - val_accuracy: 0.7274
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 2 / Accuracy: 0.7296058661778185
55/55 [=====] - 48s 864ms/step - loss: 0.6707 - accuracy: 0.7727 - val_loss: 0.7228 - val_accuracy: 0.7881
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 3 / Accuracy: 0.7855178735105408
55/55 [=====] - 47s 862ms/step - loss: 0.5292 - accuracy: 0.8197 - val_loss: 0.6628 - val_accuracy: 0.7984
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 4 / Accuracy: 0.8029330889092575
55/55 [=====] - 47s 859ms/step - loss: 0.4379 - accuracy: 0.8504 - val_loss: 0.6303 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 5 / Accuracy: 0.8111824014665444
55/55 [=====] - 48s 876ms/step - loss: 0.3704 - accuracy: 0.8750 - val_loss: 0.6106 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 6 / Accuracy: 0.8185151237396884
55/55 [=====] - 48s 866ms/step - loss: 0.3175 - accuracy: 0.8957 - val_loss: 0.6010 - val_accuracy: 0.8133
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 7 / Accuracy: 0.8194317140238313

55/55 [=====] - 47s 854ms/step - loss: 0.2747 - accuracy: 0.9126 - val_loss: 0.5970 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 7 / Accuracy: 0.8194317140238313

55/55 [=====] - 135s 2s/step - loss: 1.3277 - accuracy: 0.5307 - val_loss: 0.9061 - val_accuracy: 0.7056
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 2 / Accuracy: 0.7140238313473877

55/55 [=====] - 50s 901ms/step - loss: 0.7337 - accuracy: 0.7458 - val_loss: 0.7500 - val_accuracy: 0.7583
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 3 / Accuracy: 0.770852428964253

55/55 [=====] - 49s 890ms/step - loss: 0.5811 - accuracy: 0.7968 - val_loss: 0.6804 - val_accuracy: 0.7961
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 4 / Accuracy: 0.7956003666361137

55/55 [=====] - 49s 891ms/step - loss: 0.4835 - accuracy: 0.8381 - val_loss: 0.6355 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 5 / Accuracy: 0.8148487626031164

55/55 [=====] - 57s 1s/step - loss: 0.4097 - accuracy: 0.8619 - val_loss: 0.6062 - val_accuracy: 0.8099
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 6 / Accuracy: 0.8240146654445463

55/55 [=====] - 49s 886ms/step - loss: 0.3602 - accuracy: 0.8770 - val_loss: 0.5949 - val_accuracy: 0.8144
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 7 / Accuracy: 0.8249312557286893

55/55 [=====] - 48s 877ms/step - loss: 0.3165 - accuracy: 0.8948 - val_loss: 0.6044 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 7 / Accuracy: 0.8249312557286893

55/55 [=====] - 76s 1s/step - loss: 1.3144 - accuracy: 0.5397 - val_loss: 0.8801 - val_accuracy: 0.7216
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 2 / Accuracy: 0.7241063244729606

55/55 [=====] - 49s 893ms/step - loss: 0.7210 - accuracy: 0.7658 - val_loss: 0.7429 - val_accuracy: 0.7743
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 3 / Accuracy: 0.7873510540788268

55/55 [=====] - 49s 887ms/step - loss: 0.5911 - accuracy: 0.8065 - val_loss: 0.6782 - val_accuracy: 0.8030
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 4 / Accuracy: 0.8001833180568286

55/55 [=====] - 49s 893ms/step - loss: 0.5019 - accuracy: 0.8358 - val_loss: 0.6423 - val_accuracy: 0.8041
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 5 / Accuracy: 0.8065994500458296

55/55 [=====] - 67s 1s/step - loss: 0.4284 - accuracy: 0.8553 - val_loss: 0.6157 - val_accuracy: 0.8087
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 6 / Accuracy: 0.8185151237396884

55/55 [=====] - 48s 879ms/step - loss: 0.3992 - accuracy: 0.8670 - val_loss: 0.5992 - val_accuracy: 0.8064
Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 7 / Accuracy: 0.8258478460128322

55/55 [=====] - 47s 863ms/step - loss: 0.3539 - ac

curacy: 0.8831 - val_loss: 0.5881 - val_accuracy: 0.8156

Activation: sigmoid / Optimizer: adam / Batch size: 64 / Layers: 256 / Epochs: 7 / Accuracy: 0.8258478460128322

```
In [ ]: df_results = pd.DataFrame({'activation':df_act, 'optimizer':df_opt,'batch':  
df_batch_size, 'dropout':df_drop, 'layers':df_layers, 'epochs':df_epochs,  
'accuracy': df_accuracy})  
#df_results = pd.DataFrame({'batch': df_batch_size, 'layers':df_layers, 'epochs':df_epochs, 'accuracy': df_accuracy})  
df_results = df_results.sort_values('accuracy', ascending=False)  
print('Best iteration')  
df_results.head(1)
```

```
In [53]: df_results.to_csv('results df - descriptions cc and location.csv')
```

```
In [ ]:
```