Predicting_Heart_Problem_with_BERT_in_Tensorflow

July 28, 2020

##Mounting Google Drive

```
[ ]: from google.colab import drive
     drive.mount("/GD")
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id
=947318989803-6bn6qk8qdgf4n4g3pfee6491hc0brc4i.apps.googleusercontent.com&redire
ct_uri=urn%3aietf%3awg%3aoauth%3a2.0%3aoob&response_type=code&scope=email%20http
s%3a%2f%2fwww.googleapis.com%2fauth%2fdocs.test%20https%3a%2f%2fwww.googleapis.c
om%2fauth%2fdrive%20https%3a%2f%2fwww.googleapis.com%2fauth%2fdrive.photos.reado
nly%20https%3a%2f%2fwww.googleapis.com%2fauth%2fpeopleapi.readonly

Enter your authorization code:
··········
Mounted at /GD
```

0.1 Importing Necessary Libraries

```
[ ]: !pip install tensorflow==1.15.0
     import tensorflow as tf
     print(tf.__version__)
```

```
Collecting tensorflow==1.15.0
  Downloading https://files.pythonhosted.org/packages/3f/98/5a99af92fb911d
7a88a0005ad55005f35b4c1ba8d75fba02df726cd936e6/tensorflow-1.15.0-cp36-cp36m-many
linux2010_x86_64.whl (412.3MB)
     |                  | 412.3MB 42kB/s
Requirement already satisfied: protobuf>=3.6.1 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (3.12.2)
Requirement already satisfied: grpcio>=1.8.6 in /usr/local/lib/python3.6/dist-
packages (from tensorflow==1.15.0) (1.30.0)
Collecting gast==0.2.2
  Downloading https://files.pythonhosted.org/packages/4e/35/11749bf99b2d4e3cceb4
d55ca22590b0d7c2c62b9de38ac4a4a7f4687421/gast-0.2.2.tar.gz
Requirement already satisfied: wrapt>=1.11.1 in /usr/local/lib/python3.6/dist-
packages (from tensorflow==1.15.0) (1.12.1)
Requirement already satisfied: keras-preprocessing>=1.0.5 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.1.2)
```

```
Requirement already satisfied: google-pasta>=0.1.6 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (0.2.0)
Requirement already satisfied: numpy<2.0,>=1.16.0 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.18.5)
Requirement already satisfied: wheel>=0.26 in /usr/local/lib/python3.6/dist-
packages (from tensorflow==1.15.0) (0.34.2)
Requirement already satisfied: six>=1.10.0 in /usr/local/lib/python3.6/dist-
packages (from tensorflow==1.15.0) (1.15.0)
Requirement already satisfied: absl-py>=0.7.0 in /usr/local/lib/python3.6/dist-
packages (from tensorflow==1.15.0) (0.9.0)
Requirement already satisfied: termcolor>=1.1.0 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.1.0)
Requirement already satisfied: keras-applications>=1.0.8 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (1.0.8)
Collecting tensorflow-estimator==1.15.1
  Downloading https://files.pythonhosted.org/packages/de/62/2ee9cd74c9fa2f
a450877847ba560b260f5d0fb70ee0595203082dafcc9d/tensorflow_estimator-1.15.1-py2.p
y3-none-any.whl (503kB)
     |                 | 512kB 46.5MB/s
Requirement already satisfied: opt-einsum>=2.3.2 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (3.3.0)
Collecting tensorboard<1.16.0,>=1.15.0
  Downloading https://files.pythonhosted.org/packages/1e/e9/d3d747a97f7188
f48aa5eda486907f3b345cd409f0a0850468ba867db246/tensorboard-1.15.0-py3-none-
any.whl (3.8MB)
     |                 | 3.8MB 50.5MB/s
Requirement already satisfied: astor>=0.6.0 in
/usr/local/lib/python3.6/dist-packages (from tensorflow==1.15.0) (0.8.1)
Requirement already satisfied: setuptools in /usr/local/lib/python3.6/dist-
packages (from protobuf>=3.6.1->tensorflow==1.15.0) (49.1.0)
Requirement already satisfied: h5py in /usr/local/lib/python3.6/dist-packages
(from keras-applications>=1.0.8->tensorflow==1.15.0) (2.10.0)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.6/dist-
packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15.0) (3.2.2)
Requirement already satisfied: werkzeug>=0.11.15 in
/usr/local/lib/python3.6/dist-packages (from
tensorboard<1.16.0,>=1.15.0->tensorflow==1.15.0) (1.0.1)
Requirement already satisfied: importlib-metadata; python_version < "3.8" in
/usr/local/lib/python3.6/dist-packages (from
markdown>=2.6.8->tensorboard<1.16.0,>=1.15.0->tensorflow==1.15.0) (1.7.0)
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.6/dist-
packages (from importlib-metadata; python_version <
"3.8"->markdown>=2.6.8->tensorboard<1.16.0,>=1.15.0->tensorflow==1.15.0) (3.1.0)
Building wheels for collected packages: gast
  Building wheel for gast (setup.py) … done
  Created wheel for gast: filename=gast-0.2.2-cp36-none-any.whl size=7540
sha256=c73a6fb4c2441e60c9cc077231a14754500143267eb700eaded1530444f2f404
  Stored in directory: /root/.cache/pip/wheels/5c/2e/7e/a1d4d4fcebe6c381f378ce77
```

```
43a3ced3699feb89bcfbdadadd
Successfully built gast
ERROR: tensorflow-probability 0.10.0 has requirement gast>=0.3.2, but

you'll have gast 0.2.2 which is incompatible.
Installing collected packages: gast, tensorflow-estimator, tensorboard,
tensorflow
  Found existing installation: gast 0.3.3
    Uninstalling gast-0.3.3:
      Successfully uninstalled gast-0.3.3
  Found existing installation: tensorflow-estimator 2.2.0
    Uninstalling tensorflow-estimator-2.2.0:
      Successfully uninstalled tensorflow-estimator-2.2.0
  Found existing installation: tensorboard 2.2.2
    Uninstalling tensorboard-2.2.2:
      Successfully uninstalled tensorboard-2.2.2
  Found existing installation: tensorflow 2.2.0
    Uninstalling tensorflow-2.2.0:
      Successfully uninstalled tensorflow-2.2.0
Successfully installed gast-0.2.2 tensorboard-1.15.0 tensorflow-1.15.0
tensorflow-estimator-1.15.1
1.15.0
```

```
[ ]: import pandas as pd
     import tensorflow as tf
     import tensorflow_hub as hub
     from datetime import datetime
     from sklearn.model_selection import train_test_split
     import os

     print("tensorflow version : ", tf.__version__)
     print("tensorflow_hub version : ", hub.__version__)
```

```
tensorflow version :  1.15.0
tensorflow_hub version :  0.8.0
```

```
[ ]: #Installing BERT module
     !pip install bert-tensorflow
```

```
Collecting bert-tensorflow
  Downloading https://files.pythonhosted.org/packages/a6/66/7eb4e8b6ea35b7
cc54c322c816f976167a43019750279a8473d355800a93/bert_tensorflow-1.0.1-py2.py3-non
e-any.whl (67kB)
     |                  | 71kB 3.1MB/s
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-
packages (from bert-tensorflow) (1.15.0)
Installing collected packages: bert-tensorflow
Successfully installed bert-tensorflow-1.0.1
```

```
[ ]: #Importing BERT modules
import bert
from bert import run_classifier
from bert import optimization
from bert import tokenization
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-
packages/bert/optimization.py:87: The name tf.train.Optimizer is deprecated.
Please use tf.compat.v1.train.Optimizer instead.
```

0.2 ##Setting The Output Directory

While fine-tuning the model, we will save the training checkpoints and the model in an output directory so that we can use the trained model for our predictions later.

The following code block sets an output directory :

```
[ ]: # Set the output directory for saving model file
OUTPUT_DIR = '/GD/My Drive/Colab Notebooks/LifeHackHeart/'

#@markdown Whether or not to clear/delete the directory and create a new one
DO_DELETE = False #@param {type:"boolean"}

if DO_DELETE:
  try:
    tf.gfile.DeleteRecursively(OUTPUT_DIR)
  except:
    pass

tf.gfile.MakeDirs(OUTPUT_DIR)
print('***** Model output directory: {} *****'.format(OUTPUT_DIR))
```

```
***** Model output directory: /GD/My Drive/Colab Notebooks/LifeHackHeart/ *****
```

0.3 ##Loading The Data

We will now load the data from a Google Drive directory and will also split the training set in to training and validation sets.

```
[ ]: train = pd.read_csv("/GD/My Drive/Colab Notebooks/LifeHackHeart/smoga_ngaruh2.
 ↪csv", encoding = "ISO-8859-1")
train['question'] = train['question'].apply(str)
'''
stroke = train[train['category'] =='stroke']
nostroke = train[train['category'] !='stroke']
stroke = stroke.sample(275, random_state=None)

train = pd.concat([stroke, nostroke])
```

```
train.to_csv('final_heart_dataset.csv')
'''
from sklearn.model_selection import train_test_split

train, val =  train_test_split(train, test_size = 0.2, random_state = 100)
```

```
[ ]: train.to_csv('train.csv')
     val.to_csv('test.csv')
```

```
[ ]: #Training set sample
     train.head(15)
```

```
[ ]:       Unnamed: 0  …                                         question
     1130        1130  …  ,suami saya mengalami pembengkakan jantung ,9 …
     993          993  …  . nama saya farhan, mahasiswa umur 20 tahun. j…
     1637        1637  …  , ,   . saya tadi malam merasakan nyeri di dad…
     285          285  …  ,. , saya memiliki seorang teman yang memiliki…
     1134        1134  …  . , apakah penyakit gagal jantung bisa mempert…
     1504        1504  …  5hari yang lalu suami saya mengalami sesak naf…
     349          349  …  ,saya nita punya anak usianya 10 bulan beberap…
     1335        1335  …  , detak jantung sayang sering berdetak kencang…
     142          142  …  ,\r\n baru saja terkena stroke ringan, salah s…
     767          767  …  …\nsaya kadang merasakan rasa aneh nyeri da…
     1586        1586  …   ..mau konsultasi …dada bagian kanan saya se…
     1012        1012  …   ..sebenarnya mengukur detak jantung itu pada …
     318          318  …  , mengapaÃ¢Â elainan kromosom seperti sindrom …
     1697        1697  …   ingin  mengenai rasa nyeri pada dada sebelah …
     1557        1557  …  , saya perempuan 16 th. saya sudah 3 hari meng…

     [15 rows x 6 columns]
```

```
[ ]: print("Training Set Shape :", train.shape)
     print("Validation Set Shape :", val.shape)
     #print("Test Set Shape :", test.shape)
```

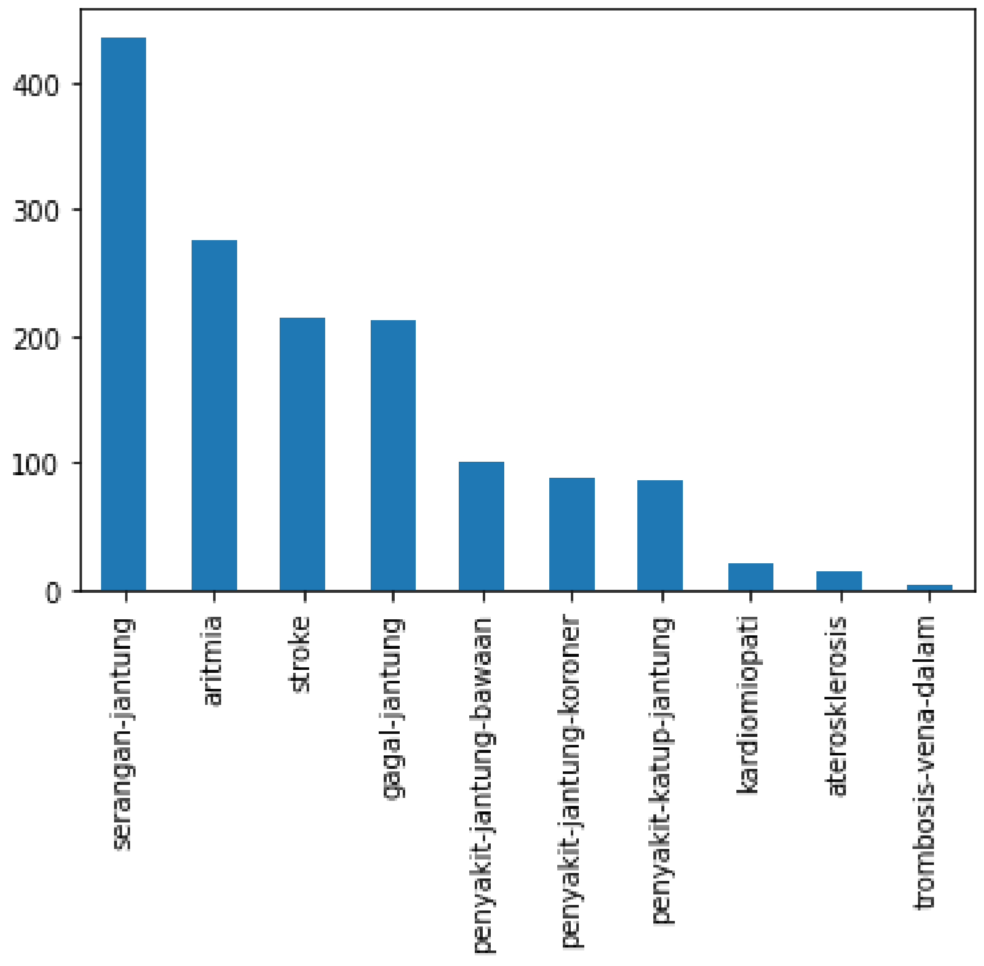
```
Training Set Shape : (1450, 6)
Validation Set Shape : (363, 6)
```

```
[ ]: #unique classes
     train['category'].unique()
```

```
[ ]: array(['gagal-jantung', 'aritmia', 'serangan-jantung',
            'penyakit-jantung-bawaan', 'stroke', 'penyakit-jantung-koroner',
            'penyakit-katup-jantung', 'trombosis-vena-dalam', 'kardiomiopati',
            'aterosklerosis'], dtype=object)
```

```
[ ]: #Distribution of classes
     train['category'].value_counts().plot(kind = 'bar')
```

```
[ ]: <matplotlib.axes._subplots.AxesSubplot at 0x7faa68f75400>
```



```
[ ]: DATA_COLUMN = 'question'
     LABEL_COLUMN = 'category'
     # The list containing all the classes (train['SECTION'].unique())
     label_list = list(train['category'].unique())
```

0.4 Data Preprocessing

BERT model accept only a specific type of input and the datasets are usually structured to have have the following four features:

- guid : A unique id that represents an observation.
- text_a : The text we need to classify into given categories

- text_b: It is used when we're training a model to understand the relationship between sentences and it does not apply for classification problems.
- label: It consists of the labels or classes or categories that a given text belongs to.

In our dataset we have text_a and label. The following code block will create objects for each of the above mentioned features for all the records in our dataset using the InputExample class provided in the BERT library.

```
[ ]: train_InputExamples = train.apply(lambda x: bert.run_classifier.
      ↪InputExample(guid=None,
                                                                     text_a =␣
      ↪x[DATA_COLUMN],
                                                                     text_b =␣
      ↪None,
                                                                     label =␣
      ↪x[LABEL_COLUMN]), axis = 1)

     val_InputExamples = val.apply(lambda x: bert.run_classifier.
      ↪InputExample(guid=None,
                                                                     text_a =␣
      ↪x[DATA_COLUMN],
                                                                     text_b =␣
      ↪None,
                                                                     label =␣
      ↪x[LABEL_COLUMN]), axis = 1)
```

```
[ ]: train_InputExamples
```

```
[ ]: 1130    <bert.run_classifier.InputExample object at 0x…
     993     <bert.run_classifier.InputExample object at 0x…
     1637    <bert.run_classifier.InputExample object at 0x…
     285     <bert.run_classifier.InputExample object at 0x…
     1134    <bert.run_classifier.InputExample object at 0x…
                                  …
     53      <bert.run_classifier.InputExample object at 0x…
     350     <bert.run_classifier.InputExample object at 0x…
     79      <bert.run_classifier.InputExample object at 0x…
     792     <bert.run_classifier.InputExample object at 0x…
     1544    <bert.run_classifier.InputExample object at 0x…
     Length: 1450, dtype: object
```

```
[ ]: print("Row 0 - guid of training set : ", train_InputExamples.iloc[0].guid)
     print("\n__________\nRow 0 - text_a of training set : ", train_InputExamples.
      ↪iloc[0].text_a)
     print("\n__________\nRow 0 - text_b of training set : ", train_InputExamples.
      ↪iloc[0].text_b)
```

```
print("\n__________\nRow 0 - label of training set : ", train_InputExamples.
↪iloc[0].label)
```

```
Row 0 - guid of training set :  None


__________
Row 0 - text_a of training set :  ,suami saya mengalami pembengkakan jantung ,9
bln terakhir ini dan kita sdhmenjalani ekg,treadmill dll,dah hasilnya ef jantung
hnya 13%,yang ingin saya kan apakah suami saya bisa sembuh kembali dan bgmn cara
n penapenanganannya,kaki jg membengkak ,mohon  petunjuk nya pengobatan dan
perawatannter,


__________
Row 0 - text_b of training set :  None


__________
Row 0 - label of training set :  gagal-jantung
```

We will now get down to business with the pretrained BERT. In this example we will use the `bert_uncased_L-12_H-768_A-12/1` model. To check all available versions click here.

We will be using the vocab.txt file in the model to map the words in the dataset to indexes. Also the loaded BERT model is trained on uncased/lowercase data and hence the data we feed to train the model should also be of lowercase.

The following code block loads the pre-trained BERT model and initializers a tokenizer object for tokenizing the texts.

```
[ ]: # This is a path to an uncased (all lowercase) version of BERT
BERT_MODEL_HUB = "https://tfhub.dev/google/bert_multi_cased_L-12_H-768_A-12/1"

def create_tokenizer_from_hub_module():
  """Get the vocab file and casing info from the Hub module."""
  with tf.Graph().as_default():
    bert_module = hub.Module(BERT_MODEL_HUB)
    tokenization_info = bert_module(signature="tokenization_info", as_dict=True)
    with tf.Session() as sess:
      vocab_file, do_lower_case = sess.run([tokenization_info["vocab_file"],
                                            tokenization_info["do_lower_case"]])

  return bert.tokenization.FullTokenizer(
      vocab_file=vocab_file, do_lower_case=do_lower_case)

tokenizer = create_tokenizer_from_hub_module()
```

```
INFO:tensorflow:Saver not created because there are no variables in the graph to
restore
```

```
INFO:tensorflow:Saver not created because there are no variables in the graph to
restore
```

[ ]:
```
#Here is what the tokenised sample of the first training set observation looks␣
↪like
print(tokenizer.tokenize(train_InputExamples.iloc[9].text_a))
```

```
['.', '.', '.', '.', 'saya', 'kadang', 'merasa', '##kan', 'rasa', 'ane', '##h',
'nye', '##ri', 'dada', '##Ã', '##£', '##Â', '##Ã', '##¢', '##Â', 'seperti',
'rasa', 'ga', '##k', 'ena', '##k', 'men', '##jala', '##r', 'dari', 'dada',
'kiri', 'naik', 'ke', 'ra', '##hang', ',', 'mu', '##ka', 'dan', 'nye', '##ri',
'tem', '##bus', 'dada', '.', 'ter', '?', 'Ã', '##£', '##Â', '##Ã', '##¢', '##Â']
```

We will now format out text in to input features which the BERT model expects. We will also set a sequence length which will be the length of the input features.

[ ]:
```
max_len = max([len(tokenizer.tokenize(train_InputExamples.iloc[IDX].text_a))␣
↪for IDX in range(1450)])
print('Max length: ', max_len)
```

```
Max length:  631
```

[ ]:
```
# We'll set sequences to be at most 128 tokens long.
MAX_SEQ_LENGTH = 256

# Convert our train and validation features to InputFeatures that BERT␣
↪understands.
train_features = bert.run_classifier.
↪convert_examples_to_features(train_InputExamples, label_list,␣
↪MAX_SEQ_LENGTH, tokenizer)

val_features = bert.run_classifier.
↪convert_examples_to_features(val_InputExamples, label_list, MAX_SEQ_LENGTH,␣
↪tokenizer)
```

```
INFO:tensorflow:Writing example 0 of 1450

INFO:tensorflow:Writing example 0 of 1450

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] , sua ##mi saya mengalami pe ##mbe ##ng ##kak ##an
jan ##tung , 9 bl ##n terakhir ini dan kita s ##dh ##men ##jalan ##i ek ##g ,
tre ##ad ##mill dl ##l , da ##h hasil ##nya ef jan ##tung h ##nya 13 % , yang
ingin saya kan apa ##kah sua ##mi saya bisa sem ##bu ##h kembali dan bg ##mn
```

```
cara n pena ##pena ##ngan ##annya , kaki j ##g me ##mbe ##ng ##kak , moh ##on
pet ##un ##juk nya pen ##go ##batan dan per ##awa ##tan ##nter , [SEP]

INFO:tensorflow:tokens: [CLS] , sua ##mi saya mengalami pe ##mbe ##ng ##kak ##an
jan ##tung , 9 bl ##n terakhir ini dan kita s ##dh ##men ##jalan ##i ek ##g ,
tre ##ad ##mill dl ##l , da ##h hasil ##nya ef jan ##tung h ##nya 13 % , yang
ingin saya kan apa ##kah sua ##mi saya bisa sem ##bu ##h kembali dan bg ##mn
cara n pena ##pena ##ngan ##annya , kaki j ##g me ##mbe ##ng ##kak , moh ##on
pet ##un ##juk nya pen ##go ##batan dan per ##awa ##tan ##nter , [SEP]

INFO:tensorflow:input_ids: 101 117 10603 10500 64981 42060 11161 35216 10376
71442 10206 63923 23091 117 130 21484 10115 36357 10592 10215 40091 187 20193
11418 95947 10116 16334 10240 117 11617 11488 100496 63940 10161 117 10143 10237
31102 10676 56331 63923 23091 176 10676 10249 110 117 10265 54419 64981 10905
32500 28977 10603 10500 64981 17103 11531 12177 10237 20879 10215 91542 47929
15903 182 39465 53303 15728 44328 117 45340 178 10240 10911 35216 10376 71442
117 49234 10263 32784 11107 23150 24091 66558 10797 47693 10215 10178 27593
12059 25446 117 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_ids: 101 117 10603 10500 64981 42060 11161 35216 10376
71442 10206 63923 23091 117 130 21484 10115 36357 10592 10215 40091 187 20193
11418 95947 10116 16334 10240 117 11617 11488 100496 63940 10161 117 10143 10237
31102 10676 56331 63923 23091 176 10676 10249 110 117 10265 54419 64981 10905
32500 28977 10603 10500 64981 17103 11531 12177 10237 20879 10215 91542 47929
15903 182 39465 53303 15728 44328 117 45340 178 10240 10911 35216 10376 71442
117 49234 10263 32784 11107 23150 24091 66558 10797 47693 10215 10178 27593
12059 25446 117 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: gagal-jantung (id = 0)

INFO:tensorflow:label: gagal-jantung (id = 0)

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] . nama saya far ##han , mahasiswa umur 20 tahun .
jadi sem ##ala ##m saya tid ##ur , pertengahan saya merasa ##kan sekali jan
##tung be ##rhenti be ##rde ##gu ##p 4 ##x be ##rtu ##rut - turut interval waktu
2 det ##ik . yang saya rasa ##kan itu jan ##tung ke ##jut ##an me ##mom ##pa
tiba - tiba setelah be ##rhenti de ##gu ##p itu jadi saya tid ##ur merasa ter
##ka ##get - ka ##get yang ane ##h . saya belum pernah ri ##way ##at jan ##tung
namun serangan jan ##tung kecil ( seperti ter ##jat ##uh saat tid ##ur ) setiap
bulan ada tapi tidak begitu sering , saya ra ##jin ola ##hraga be ##ban dan kar
##dio 4 ##x semi ##nggu . jadi gi ##mana ma ##ksud ##nya sakit saya itu ? saya
bang ##un tid ##ur hingga sekarang masih merasa ##kan nye ##ri di dada kiri ,
sem ##ala ##m itu saya tid ##ur + - 7 ##jam . . [SEP]

INFO:tensorflow:tokens: [CLS] . nama saya far ##han , mahasiswa umur 20 tahun .
jadi sem ##ala ##m saya tid ##ur , pertengahan saya merasa ##kan sekali jan
##tung be ##rhenti be ##rde ##gu ##p 4 ##x be ##rtu ##rut - turut interval waktu
2 det ##ik . yang saya rasa ##kan itu jan ##tung ke ##jut ##an me ##mom ##pa
tiba - tiba setelah be ##rhenti de ##gu ##p itu jadi saya tid ##ur merasa ter
##ka ##get - ka ##get yang ane ##h . saya belum pernah ri ##way ##at jan ##tung
namun serangan jan ##tung kecil ( seperti ter ##jat ##uh saat tid ##ur ) setiap
bulan ada tapi tidak begitu sering , saya ra ##jin ola ##hraga be ##ban dan kar
##dio 4 ##x semi ##nggu . jadi gi ##mana ma ##ksud ##nya sakit saya itu ? saya
bang ##un tid ##ur hingga sekarang masih merasa ##kan nye ##ri di dada kiri ,
sem ##ala ##m itu saya tid ##ur + - 7 ##jam . . [SEP]
```

```
INFO:tensorflow:input_ids: 101 119 15359 64981 13301 11781 117 96931 25453 10197
10989 119 17760 11531 13322 10147 64981 15201 10546 117 75128 64981 93843 10706
46233 63923 23091 10347 107329 10347 17229 12589 10410 125 10686 10347 39096
30233 118 39932 72331 22952 123 10349 10896 119 10265 64981 45772 10706 11910
63923 23091 11163 58851 10206 10911 86366 11359 52986 118 52986 18044 10347
107329 10104 12589 10410 11910 17760 64981 15201 10546 93843 12718 10371 14908
118 10730 14908 10265 46226 10237 119 64981 48118 21407 29956 14132 10526 63923
23091 22736 43407 63923 23091 23337 113 13908 12718 21757 18593 16214 15201
10546 114 24590 12385 15290 64747 11868 59130 28586 117 64981 11859 21331 76893
79767 10347 10927 10215 25085 16994 125 10686 15900 49443 119 17760 38356 20338
10824 69408 10676 57236 64981 11910 136 64981 17937 11107 15201 10546 18295
28344 20535 93843 10706 17731 10401 10120 42020 86279 117 11531 13322 10147
11910 64981 15201 10546 116 118 128 36887 119 119 102 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_ids: 101 119 15359 64981 13301 11781 117 96931 25453 10197
10989 119 17760 11531 13322 10147 64981 15201 10546 117 75128 64981 93843 10706
46233 63923 23091 10347 107329 10347 17229 12589 10410 125 10686 10347 39096
30233 118 39932 72331 22952 123 10349 10896 119 10265 64981 45772 10706 11910
63923 23091 11163 58851 10206 10911 86366 11359 52986 118 52986 18044 10347
107329 10104 12589 10410 11910 17760 64981 15201 10546 93843 12718 10371 14908
118 10730 14908 10265 46226 10237 119 64981 48118 21407 29956 14132 10526 63923
23091 22736 43407 63923 23091 23337 113 13908 12718 21757 18593 16214 15201
10546 114 24590 12385 15290 64747 11868 59130 28586 117 64981 11859 21331 76893
79767 10347 10927 10215 25085 16994 125 10686 15900 49443 119 17760 38356 20338
10824 69408 10676 57236 64981 11910 136 64981 17937 11107 15201 10546 18295
28344 20535 93843 10706 17731 10401 10120 42020 86279 117 11531 13322 10147
11910 64981 15201 10546 116 118 128 36887 119 119 102 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: aritmia (id = 1)

INFO:tensorflow:label: aritmia (id = 1)

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] , , . saya tad ##i malam merasa ##kan nye ##ri di
dada sebelah kiri sampai pagi ini tetapi tidak mengalami gang ##guan lain .
biasanya saya su ##ka mengalami gang ##guan ini tetapi tidak lama langsung sem
##bu ##b sendiri . kala ##u sekarang agak lama saya merasa ##kan nya . . gang
##guan y ##g saya alam ##i apa ##kah be ##rba ##haya ? sakit ##nya itu seperti
di teka ##n / ditu ##su ##k . . ter ##ima ##kasi ##h don [SEP]

INFO:tensorflow:tokens: [CLS] , , . saya tad ##i malam merasa ##kan nye ##ri di
dada sebelah kiri sampai pagi ini tetapi tidak mengalami gang ##guan lain .
biasanya saya su ##ka mengalami gang ##guan ini tetapi tidak lama langsung sem
##bu ##b sendiri . kala ##u sekarang agak lama saya merasa ##kan nya . . gang
##guan y ##g saya alam ##i apa ##kah be ##rba ##haya ? sakit ##nya itu seperti
di teka ##n / ditu ##su ##k . . ter ##ima ##kasi ##h don [SEP]

INFO:tensorflow:input_ids: 101 117 117 119 64981 50586 10116 51697 93843 10706
17731 10401 10120 42020 40988 86279 20853 104716 10592 17401 11868 42060 16330
72769 13514 119 17295 64981 10198 10371 42060 16330 72769 10592 17401 11868
26994 35091 11531 12177 10457 21684 119 84844 10138 28344 100501 26994 64981
93843 10706 24091 119 119 16330 72769 193 10240 64981 40796 10116 32500 28977
10347 61494 59562 136 57236 10676 11910 13908 10120 107414 10115 120 15507 12892
10174 119 119 12718 12443 37997 10237 16938 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:input_ids: 101 117 117 119 64981 50586 10116 51697 93843 10706
17731 10401 10120 42020 40988 86279 20853 104716 10592 17401 11868 42060 16330
72769 13514 119 17295 64981 10198 10371 42060 16330 72769 10592 17401 11868
26994 35091 11531 12177 10457 21684 119 84844 10138 28344 100501 26994 64981
93843 10706 24091 119 119 16330 72769 193 10240 64981 40796 10116 32500 28977
10347 61494 59562 136 57236 10676 11910 13908 10120 107414 10115 120 15507 12892
10174 119 119 12718 12443 37997 10237 16938 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:label: serangan-jantung (id = 2)
```

```
INFO:tensorflow:label: serangan-jantung (id = 2)
```

```
INFO:tensorflow:*** Example ***
```

```
INFO:tensorflow:*** Example ***
```

```
INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] , . , saya memiliki seorang teman yang memiliki
kela ##inan jan ##tung ba ##waan . dia pernah mengalami mun ##tah darah 3 ##x dl
##m 2 hari terakhir . apa ##kah sudah di ##wa ##jib ##kian ut ##k operasi , ?
[SEP]

INFO:tensorflow:tokens: [CLS] , . , saya memiliki seorang teman yang memiliki
kela ##inan jan ##tung ba ##waan . dia pernah mengalami mun ##tah darah 3 ##x dl
##m 2 hari terakhir . apa ##kah sudah di ##wa ##jib ##kian ut ##k operasi , ?
[SEP]

INFO:tensorflow:input_ids: 101 117 119 117 64981 13363 13974 71476 10265 13363
39691 39646 63923 23091 15688 109261 119 10671 21407 42060 101833 53538 43947
124 10686 63940 10147 123 18370 36357 119 32500 28977 25147 10120 11037 102994
79189 11735 10174 56516 117 136 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_ids: 101 117 119 117 64981 13363 13974 71476 10265 13363
39691 39646 63923 23091 15688 109261 119 10671 21407 42060 101833 53538 43947
124 10686 63940 10147 123 18370 36357 119 32500 28977 25147 10120 11037 102994
79189 11735 10174 56516 117 136 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: penyakit-jantung-bawaan (id = 3)

INFO:tensorflow:label: penyakit-jantung-bawaan (id = 3)

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] . , apa ##kah penyakit gagal jan ##tung bisa
mempertahankan stadium ##nya ? ( mis ##al dia sekarang sedang di stadium 2 , dia
tetap berada di stadium tersebut ) lalu bagi orang y ##g di ##pasang ring , bera
##pa lama kemungkinan dia bisa bertahan hidup dengan ring y ##g di ##pasang ?
moh ##on pen ##jela ##san ##nya . [SEP]

INFO:tensorflow:tokens: [CLS] . , apa ##kah penyakit gagal jan ##tung bisa
mempertahankan stadium ##nya ? ( mis ##al dia sekarang sedang di stadium 2 , dia
tetap berada di stadium tersebut ) lalu bagi orang y ##g di ##pasang ring , bera
##pa lama kemungkinan dia bisa bertahan hidup dengan ring y ##g di ##pasang ?
moh ##on pen ##jela ##san ##nya . [SEP]

INFO:tensorflow:input_ids: 101 119 117 32500 28977 64951 70591 63923 23091 17103
102567 27915 10676 136 113 12606 10415 10671 28344 37585 10120 27915 123 117
10671 35580 21167 10120 27915 12848 114 31288 15941 12430 193 10240 10120 106028
21550 117 85199 11359 26994 83918 10671 17103 90873 19378 10659 21550 193 10240
10120 106028 136 49234 10263 66558 37142 14434 10676 119 102 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_ids: 101 119 117 32500 28977 64951 70591 63923 23091 17103
102567 27915 10676 136 113 12606 10415 10671 28344 37585 10120 27915 123 117
10671 35580 21167 10120 27915 12848 114 31288 15941 12430 193 10240 10120 106028
21550 117 85199 11359 26994 83918 10671 17103 90873 19378 10659 21550 193 10240
```

```
10120 106028 136 49234 10263 66558 37142 14434 10676 119 102 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: gagal-jantung (id = 0)

INFO:tensorflow:label: gagal-jantung (id = 0)

INFO:tensorflow:Writing example 0 of 363

INFO:tensorflow:Writing example 0 of 363

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None
```

```
INFO:tensorflow:tokens: [CLS] , umur saya 18th ##n . sudah 4 hari ini saya
merasa ##kan sakit di bagian dada bahkan ter ##asa sampai ke belakang saat bern
##afa ##s , awalnya per ##ut saya sakit bagian atas lalu sk ##rg sakit ##nya
menjadi di dada , itu kn ##apa ? [SEP]

INFO:tensorflow:tokens: [CLS] , umur saya 18th ##n . sudah 4 hari ini saya
merasa ##kan sakit di bagian dada bahkan ter ##asa sampai ke belakang saat bern
##afa ##s , awalnya per ##ut saya sakit bagian atas lalu sk ##rg sakit ##nya
menjadi di dada , itu kn ##apa ? [SEP]

INFO:tensorflow:input_ids: 101 117 25453 64981 27669 10115 119 25147 125 18370
10592 64981 93843 10706 57236 10120 11551 42020 57177 12718 23031 20853 11163
51045 16214 102696 90804 10107 117 63699 10178 11159 64981 57236 11551 16695
31288 66998 20251 57236 10676 11999 10120 42020 117 11910 96820 46757 136 102 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0

INFO:tensorflow:input_ids: 101 117 25453 64981 27669 10115 119 25147 125 18370
10592 64981 93843 10706 57236 10120 11551 42020 57177 12718 23031 20853 11163
51045 16214 102696 90804 10107 117 63699 10178 11159 64981 57236 11551 16695
31288 66998 20251 57236 10676 11999 10120 42020 117 11910 96820 46757 136 102 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: serangan-jantung (id = 2)

INFO:tensorflow:label: serangan-jantung (id = 2)

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] , , saya ada masalah dengan ira ##ma jan ##tung
saya dalam 2 minggu terakhir ini , yang saya rasa ##kan pertama kali , yaitu
sensa ##si ke ##mbung di per ##ut saya , kemudian ses ##eka ##li ada skip atau
satu det ##akan lebih awal pada det ##ak jan ##tung saya . setelah saya
melakukan pe ##meri ##ksa ##an ke jan ##tung dan melakukan ek ##g , ge ##jala
tersebut tidak muncul , dan pada saat melakukan ek ##g tre ##ad ##mill , ge
##jala tersebut hanya sekali muncul pada saat perubahan posisi dari dud ##uk ke
berdiri , dan pada saat dilakukan te ##s hingga 8 menit , ge ##jala tersebut
tidak muncul sama sekali , sehingga men ##gang ##gap kondisi jan ##tung saya
normal . namun , semi ##nggu terakhir ini ge ##jala tersebut selalu muncul
hingga be ##rul ##ang - ulang kali bahkan sampai saat ini saya masih merasa
##kan nya . kala ##u di gambar ##kan det ##ak jan ##tung nya kira ##2 seperti
ini : - - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - Ã ##¢ ##Â ##Â - -
- - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - Ã ##¢ ##Â ##Â - - - - Ã
##¢ ##Â ##Â - Ã ##¢ ##Â ##Â - - - - Ã ##¢ ##Â ##Â - - untuk tekanan darah normal
di angka 120 / 80 apa ##kah itu [SEP]

INFO:tensorflow:tokens: [CLS] , , saya ada masalah dengan ira ##ma jan ##tung
saya dalam 2 minggu terakhir ini , yang saya rasa ##kan pertama kali , yaitu
sensa ##si ke ##mbung di per ##ut saya , kemudian ses ##eka ##li ada skip atau
satu det ##akan lebih awal pada det ##ak jan ##tung saya . setelah saya
melakukan pe ##meri ##ksa ##an ke jan ##tung dan melakukan ek ##g , ge ##jala
tersebut tidak muncul , dan pada saat melakukan ek ##g tre ##ad ##mill , ge
##jala tersebut hanya sekali muncul pada saat perubahan posisi dari dud ##uk ke
berdiri , dan pada saat dilakukan te ##s hingga 8 menit , ge ##jala tersebut
tidak muncul sama sekali , sehingga men ##gang ##gap kondisi jan ##tung saya
normal . namun , semi ##nggu terakhir ini ge ##jala tersebut selalu muncul
```

```
hingga be ##rul ##ang - ulang kali bahkan sampai saat ini saya masih merasa
##kan nya . kala ##u di gambar ##kan det ##ak jan ##tung nya kira ##2 seperti
ini : - - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - Ã ##¢ ##Â ##Â - -
- - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - - Ã ##¢ ##Â ##Â - Ã ##¢ ##Â ##Â - - - - Ã
##¢ ##Â ##Â - Ã ##¢ ##Â ##Â - - - - Ã ##¢ ##Â ##Â - - untuk tekanan darah normal
di angka 120 / 80 apa ##kah itu [SEP]

INFO:tensorflow:input_ids: 101 117 117 64981 15290 42697 10659 95190 10369 63923
23091 64981 10663 123 74646 36357 10592 117 10265 64981 45772 10706 14253 16384
117 20131 63175 10449 11163 110448 10120 10178 11159 64981 117 16113 10974 31519
10390 15290 52124 11754 12591 10349 27125 13394 23605 10585 10349 10710 63923
23091 64981 119 18044 64981 27286 11161 85137 42430 10206 11163 63923 23091
10215 27286 16334 10240 117 46503 30216 12848 11868 35187 117 10215 10585 16214
27286 16334 10240 11617 11488 100496 117 46503 30216 12848 18029 46233 35187
10585 16214 55442 44251 10397 64519 13013 11163 76965 117 10215 10585 16214
28920 10361 10107 18295 129 90256 117 46503 30216 12848 11868 35187 14469 46233
117 19793 10588 13755 76320 90582 63923 23091 64981 16626 119 22736 117 15900
49443 36357 10592 46503 30216 12848 56894 35187 18295 10347 24849 11889 118
74910 16384 57177 20853 16214 10592 64981 20535 93843 10706 24091 119 84844
10138 10120 60022 10706 10349 10710 63923 23091 24091 32105 10729 13908 10592
131 118 118 228 110883 110904 110904 118 118 228 110883 110904 110904 118 118
228 110883 110904 110904 118 228 110883 110904 110904 118 118 118 118 228 110883
110904 110904 118 118 228 110883 110904 110904 118 118 228 110883 110904 110904
118 228 110883 110904 110904 118 118 118 118 228 110883 110904 110904 118 228
110883 110904 110904 118 118 118 118 228 110883 110904 110904 118 118 10782
93131 43947 16626 10120 73853 12048 120 10832 32500 28977 11910 102
```

```
INFO:tensorflow:input_ids: 101 117 117 64981 15290 42697 10659 95190 10369 63923
23091 64981 10663 123 74646 36357 10592 117 10265 64981 45772 10706 14253 16384
117 20131 63175 10449 11163 110448 10120 10178 11159 64981 117 16113 10974 31519
10390 15290 52124 11754 12591 10349 27125 13394 23605 10585 10349 10710 63923
23091 64981 119 18044 64981 27286 11161 85137 42430 10206 11163 63923 23091
10215 27286 16334 10240 117 46503 30216 12848 11868 35187 117 10215 10585 16214
27286 16334 10240 11617 11488 100496 117 46503 30216 12848 18029 46233 35187
10585 16214 55442 44251 10397 64519 13013 11163 76965 117 10215 10585 16214
28920 10361 10107 18295 129 90256 117 46503 30216 12848 11868 35187 14469 46233
117 19793 10588 13755 76320 90582 63923 23091 64981 16626 119 22736 117 15900
49443 36357 10592 46503 30216 12848 56894 35187 18295 10347 24849 11889 118
74910 16384 57177 20853 16214 10592 64981 20535 93843 10706 24091 119 84844
10138 10120 60022 10706 10349 10710 63923 23091 24091 32105 10729 13908 10592
131 118 118 228 110883 110904 110904 118 118 228 110883 110904 110904 118 118
228 110883 110904 110904 118 228 110883 110904 110904 118 118 118 118 228 110883
110904 110904 118 118 228 110883 110904 110904 118 118 228 110883 110904 110904
118 228 110883 110904 110904 118 118 118 118 228 110883 110904 110904 118 228
110883 110904 110904 118 118 118 118 228 110883 110904 110904 118 118 10782
93131 43947 16626 10120 73853 12048 120 10832 32500 28977 11910 102
```

```
INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:label: aritmia (id = 1)
```

```
INFO:tensorflow:label: aritmia (id = 1)
```

```
INFO:tensorflow:*** Example ***
```

```
INFO:tensorflow:*** Example ***
```

```
INFO:tensorflow:guid: None
```

```
INFO:tensorflow:guid: None
```

```
INFO:tensorflow:tokens: [CLS] , akhir " ini ketika saya mena ##iki tangga jan
##tung saya be ##rde ##bar keras hingga ter ##asa tanpa di ##pe ##gang dan ten
##gku ##k le ##her juga demikian , juga na ##fas ter ##asa ng ##os " an . ge
##jala ya itu ? [SEP]
```

```
INFO:tensorflow:tokens: [CLS] , akhir " ini ketika saya mena ##iki tangga jan
##tung saya be ##rde ##bar keras hingga ter ##asa tanpa di ##pe ##gang dan ten
##gku ##k le ##her juga demikian , juga na ##fas ter ##asa ng ##os " an . ge
##jala ya itu ? [SEP]
```

```
INFO:tensorflow:input_ids: 101 117 26814 107 10592 19940 64981 54779 20897 74735
63923 23091 64981 10347 17229 12867 54235 18295 12718 23031 29498 10120 11355
```

```
13755 10215 11769 88714 10174 10141 14206 11535 57164 117 11535 10132 57797
12718 23031 10743 10310 107 10151 119 46503 30216 10549 11910 136 102 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0

INFO:tensorflow:input_ids: 101 117 26814 107 10592 19940 64981 54779 20897 74735
63923 23091 64981 10347 17229 12867 54235 18295 12718 23031 29498 10120 11355
13755 10215 11769 88714 10174 10141 14206 11535 57164 117 11535 10132 57797
12718 23031 10743 10310 107 10151 119 46503 30216 10549 11910 136 102 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: aritmia (id = 1)

INFO:tensorflow:label: aritmia (id = 1)

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] saya berumur 34 th ##n . saya 3 hr y ##g lalu ken
##a penyakit seperti orang y ##g lu ##mpu ##h . tangan dan kaki tidak bisa dig
##era ##kan . itu terjadi tiba ##2 . saya be ##ro ##bat alternatif , ka saya ter
##ken ##a darah din ##gin jadi darah me ##mbe ##ku dan tidak men ##gali ##r .
yang ingin saya kan apa ##kah benar saya ken ##a penyakit itu ? apa tin ##dak
lan ##jut y ##g harus saya la ##kuk ##an ? bagaimana saya dapat men ##gant ##isi
##pasi agar tidak terjadi lagi ? [SEP]

INFO:tensorflow:tokens: [CLS] saya berumur 34 th ##n . saya 3 hr y ##g lalu ken
##a penyakit seperti orang y ##g lu ##mpu ##h . tangan dan kaki tidak bisa dig
##era ##kan . itu terjadi tiba ##2 . saya be ##ro ##bat alternatif , ka saya ter
##ken ##a darah din ##gin jadi darah me ##mbe ##ku dan tidak men ##gali ##r .
yang ingin saya kan apa ##kah benar saya ken ##a penyakit itu ? apa tin ##dak
lan ##jut y ##g harus saya la ##kuk ##an ? bagaimana saya dapat men ##gant ##isi
##pasi agar tidak terjadi lagi ? [SEP]

INFO:tensorflow:input_ids: 101 64981 105994 11069 77586 10115 119 64981 124
40399 193 10240 31288 67680 10113 64951 13908 12430 193 10240 14657 98443 10237
119 48371 10215 45340 11868 17103 80592 12015 10706 119 11910 28742 52986 10729
119 64981 10347 10567 18234 84329 117 10730 64981 12718 11062 10113 43947 10595
18823 17760 43947 10911 35216 10853 10215 11868 10588 53740 10129 119 10265
54419 64981 10905 32500 28977 51990 64981 67680 10113 64951 11910 136 32500
21629 37052 12228 58851 193 10240 29062 64981 10109 76666 10206 136 100831 64981
13377 10588 44762 14553 104565 36769 11868 28742 21129 136 102 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_ids: 101 64981 105994 11069 77586 10115 119 64981 124
40399 193 10240 31288 67680 10113 64951 13908 12430 193 10240 14657 98443 10237
119 48371 10215 45340 11868 17103 80592 12015 10706 119 11910 28742 52986 10729
119 64981 10347 10567 18234 84329 117 10730 64981 12718 11062 10113 43947 10595
18823 17760 43947 10911 35216 10853 10215 11868 10588 53740 10129 119 10265
54419 64981 10905 32500 28977 51990 64981 67680 10113 64951 11910 136 32500
21629 37052 12228 58851 193 10240 29062 64981 10109 76666 10206 136 100831 64981
13377 10588 44762 14553 104565 36769 11868 28742 21129 136 102 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: stroke (id = 4)

INFO:tensorflow:label: stroke (id = 4)

INFO:tensorflow:*** Example ***

INFO:tensorflow:*** Example ***

INFO:tensorflow:guid: None

INFO:tensorflow:guid: None

INFO:tensorflow:tokens: [CLS] saya berusia 19 tahun . . ya ##a setiap saya
melakukan aktiv ##itas ola ##hraga saya merasa ##kan det ##ak jan ##tung seperti
be ##rhenti sebe ##ntar dan badan seperti dia ##lir ##i ha ##wa din ##gin . .
ketika isti ##rah ##at det ##ak jan ##tung ter ##asa cepat dan la ##mbat . .
sebelumnya saya sudah per ##iks ##a ke beberapa dan hasil ##nya semua normal . .
hanya saja saya belum sempat re ##kam jan ##tung di rs [SEP]
```

```
INFO:tensorflow:tokens: [CLS] saya berusia 19 tahun . . ya ##a setiap saya
melakukan aktiv ##itas ola ##hraga saya merasa ##kan det ##ak jan ##tung seperti
be ##rhenti sebe ##ntar dan badan seperti dia ##lir ##i ha ##wa din ##gin . .
ketika isti ##rah ##at det ##ak jan ##tung ter ##asa cepat dan la ##mbat . .
sebelumnya saya sudah per ##iks ##a ke beberapa dan hasil ##nya semua normal . .
hanya saja saya belum sempat re ##kam jan ##tung di rs [SEP]
```

```
INFO:tensorflow:input_ids: 101 64981 56440 10270 10989 119 119 10549 10113 24590
64981 27286 22275 18491 76893 79767 64981 93843 10706 10349 10710 63923 23091
13908 10347 107329 35669 30632 10215 51463 13908 10671 24328 10116 10228 11037
10595 18823 119 119 19940 61186 23497 10526 10349 10710 63923 23091 12718 23031
65160 10215 10109 90444 119 119 35252 64981 25147 10178 40670 10113 11163 15334
10215 31102 10676 23367 16626 119 119 18029 44725 64981 48118 77806 11639 46750
63923 23091 10120 48495 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:input_ids: 101 64981 56440 10270 10989 119 119 10549 10113 24590
64981 27286 22275 18491 76893 79767 64981 93843 10706 10349 10710 63923 23091
13908 10347 107329 35669 30632 10215 51463 13908 10671 24328 10116 10228 11037
10595 18823 119 119 19940 61186 23497 10526 10349 10710 63923 23091 12718 23031
65160 10215 10109 90444 119 119 35252 64981 25147 10178 40670 10113 11163 15334
10215 31102 10676 23367 16626 119 119 18029 44725 64981 48118 77806 11639 46750
63923 23091 10120 48495 102 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:input_mask: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:segment_ids: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

INFO:tensorflow:label: aritmia (id = 1)

INFO:tensorflow:label: aritmia (id = 1)
```

[ ]:
```python
#Example on first observation in the training set
print("Sentence : ", train_InputExamples.iloc[0].text_a)
print("-"*30)
print("Tokens : ", tokenizer.tokenize(train_InputExamples.iloc[0].text_a))
print("-"*30)
print("Input IDs : ", train_features[0].input_ids)
print("-"*30)
print("Input Masks : ", train_features[0].input_mask)
print("-"*30)
print("Segment IDs : ", train_features[0].segment_ids)
```

```
Sentence :  ,suami saya mengalami pembengkakan jantung ,9 bln terakhir ini dan
kita sdhmenjalani ekg,treadmill dll,dah hasilnya ef jantung hnya 13%,yang ingin
saya kan apakah suami saya bisa sembuh kembali dan bgmn cara n
penapenangananannya,kaki jg membengkak ,mohon  petunjuk nya pengobatan dan
perawatannter,
------------------------------
Tokens :  [',', 'sua', '##mi', 'saya', 'mengalami', 'pe', '##mbe', '##ng',
'##kak', '##an', 'jan', '##tung', ',', '9', 'bl', '##n', 'terakhir', 'ini',
'dan', 'kita', 's', '##dh', '##men', '##jalan', '##i', 'ek', '##g', ',', 'tre',
'##ad', '##mill', 'dl', '##l', ',', 'da', '##h', 'hasil', '##nya', 'ef', 'jan',
'##tung', 'h', '##nya', '13', '%', ',', 'yang', 'ingin', 'saya', 'kan', 'apa',
'##kah', 'sua', '##mi', 'saya', 'bisa', 'sem', '##bu', '##h', 'kembali', 'dan',
'bg', '##mn', 'cara', 'n', 'pena', '##pena', '##ngan', '##annya', ',', 'kaki',
'j', '##g', 'me', '##mbe', '##ng', '##kak', ',', 'moh', '##on', 'pet', '##un',
'##juk', 'nya', 'pen', '##go', '##batan', 'dan', 'per', '##awa', '##tan',
'##nter', ',']
------------------------------
Input IDs :  [101, 117, 10603, 10500, 64981, 42060, 11161, 35216, 10376, 71442,
10206, 63923, 23091, 117, 130, 21484, 10115, 36357, 10592, 10215, 40091, 187,
20193, 11418, 95947, 10116, 16334, 10240, 117, 11617, 11488, 100496, 63940,
10161, 117, 10143, 10237, 31102, 10676, 56331, 63923, 23091, 176, 10676, 10249,
110, 117, 10265, 54419, 64981, 10905, 32500, 28977, 10603, 10500, 64981, 17103,
11531, 12177, 10237, 20879, 10215, 91542, 47929, 15903, 182, 39465, 53303,
15728, 44328, 117, 45340, 178, 10240, 10911, 35216, 10376, 71442, 117, 49234,
10263, 32784, 11107, 23150, 24091, 66558, 10797, 47693, 10215, 10178, 27593,
12059, 25446, 117, 102, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0]
------------------------------
Input Masks :  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
------------------------------
Segment IDs :  [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

##Creating A Multi-Class Classifier Model

```
[ ]: def create_model(is_predicting, input_ids, input_mask, segment_ids, labels,
                      num_labels):

       bert_module = hub.Module(
           BERT_MODEL_HUB,
           trainable=True)
       bert_inputs = dict(
           input_ids=input_ids,
           input_mask=input_mask,
           segment_ids=segment_ids)
       bert_outputs = bert_module(
           inputs=bert_inputs,
           signature="tokens",
           as_dict=True)

       # Use "pooled_output" for classification tasks on an entire sentence.
       # Use "sequence_outputs" for token-level output.
       output_layer = bert_outputs["pooled_output"]

       hidden_size = output_layer.shape[-1].value

       # Create our own layer to tune for politeness data.
       output_weights = tf.get_variable(
           "output_weights", [num_labels, hidden_size],
           initializer=tf.truncated_normal_initializer(stddev=0.02))

       output_bias = tf.get_variable(
           "output_bias", [num_labels], initializer=tf.zeros_initializer())

       with tf.variable_scope("loss"):

         # Dropout helps prevent overfitting
         output_layer = tf.nn.dropout(output_layer, keep_prob=0.9)

         logits = tf.matmul(output_layer, output_weights, transpose_b=True)
         logits = tf.nn.bias_add(logits, output_bias)
         log_probs = tf.nn.log_softmax(logits, axis=-1)

         # Convert labels into one-hot encoding
         one_hot_labels = tf.one_hot(labels, depth=num_labels, dtype=tf.float32)

         predicted_labels = tf.squeeze(tf.argmax(log_probs, axis=-1, output_type=tf.
      ↪int32))
         # If we're predicting, we want predicted labels and the probabiltiies.
         if is_predicting:
           return (predicted_labels, log_probs)

         # If we're train/eval, compute loss between predicted and actual label
         per_example_loss = -tf.reduce_sum(one_hot_labels * log_probs, axis=-1)
         loss = tf.reduce_mean(per_example_loss)
         return (loss, predicted_labels, log_probs)
```

[ ]:
```
#A function that adapts our model to work for training, evaluation, and␣
 ↪prediction.

# model_fn_builder actually creates our model function
# using the passed parameters for num_labels, learning_rate, etc.
def model_fn_builder(num_labels, learning_rate, num_train_steps,
                     num_warmup_steps):
```

```
  """Returns `model_fn` closure for TPUEstimator."""
  def model_fn(features, labels, mode, params):  # pylint:␣
↪disable=unused-argument
    """The `model_fn` for TPUEstimator."""

    input_ids = features["input_ids"]
    input_mask = features["input_mask"]
    segment_ids = features["segment_ids"]
    label_ids = features["label_ids"]

    is_predicting = (mode == tf.estimator.ModeKeys.PREDICT)

    # TRAIN and EVAL
    if not is_predicting:

      (loss, predicted_labels, log_probs) = create_model(
        is_predicting, input_ids, input_mask, segment_ids, label_ids,␣
↪num_labels)

      train_op = bert.optimization.create_optimizer(
          loss, learning_rate, num_train_steps, num_warmup_steps, use_tpu=False)

      # Calculate evaluation metrics.
      def metric_fn(label_ids, predicted_labels):
        accuracy = tf.metrics.accuracy(label_ids, predicted_labels)
        true_pos = tf.metrics.true_positives(
            label_ids,
            predicted_labels)
        true_neg = tf.metrics.true_negatives(
            label_ids,
            predicted_labels)
        false_pos = tf.metrics.false_positives(
            label_ids,
            predicted_labels)
        false_neg = tf.metrics.false_negatives(
            label_ids,
            predicted_labels)

        return {
            "eval_accuracy": accuracy,
            "true_positives": true_pos,
            "true_negatives": true_neg,
            "false_positives": false_pos,
            "false_negatives": false_neg
            }

      eval_metrics = metric_fn(label_ids, predicted_labels)
      if mode == tf.estimator.ModeKeys.TRAIN:
        return tf.estimator.EstimatorSpec(mode=mode,
          loss=loss,
          train_op=train_op)
      else:
          return tf.estimator.EstimatorSpec(mode=mode,
            loss=loss,
            eval_metric_ops=eval_metrics)
    else:
      (predicted_labels, log_probs) = create_model(
        is_predicting, input_ids, input_mask, segment_ids, label_ids,␣
↪num_labels)

      predictions = {
          'probabilities': log_probs,
          'labels': predicted_labels
      }
      return tf.estimator.EstimatorSpec(mode, predictions=predictions)

  # Return the actual model function in the closure
  return model_fn
```

[ ]:
```
# Compute train and warmup steps from batch size
# These hyperparameters are copied from this colab notebook (https://colab.
↪sandbox.google.com/github/tensorflow/tpu/blob/master/tools/colab/
↪bert_finetuning_with_cloud_tpus.ipynb)
BATCH_SIZE = 16
LEARNING_RATE = 2e-5
NUM_TRAIN_EPOCHS = 5
# Warmup is a period of time where the learning rate is small and gradually␣
↪increases--usually helps training.
WARMUP_PROPORTION = 0.1
# Model configs
SAVE_CHECKPOINTS_STEPS = 300
SAVE_SUMMARY_STEPS = 100

# Compute train and warmup steps from batch size
num_train_steps = int(len(train_features) / BATCH_SIZE * NUM_TRAIN_EPOCHS)
num_warmup_steps = int(num_train_steps * WARMUP_PROPORTION)

# Specify output directory and number of checkpoint steps to save
run_config = tf.estimator.RunConfig(
    model_dir=OUTPUT_DIR,
    save_summary_steps=SAVE_SUMMARY_STEPS,
    save_checkpoints_steps=SAVE_CHECKPOINTS_STEPS)
```

```
# Specify output directory and number of checkpoint steps to save
run_config = tf.estimator.RunConfig(
    model_dir=OUTPUT_DIR,
    save_summary_steps=SAVE_SUMMARY_STEPS,
    save_checkpoints_steps=SAVE_CHECKPOINTS_STEPS)
```

[ ]:
```
#Initializing the model and the estimator
model_fn = model_fn_builder(
  num_labels=len(label_list),
  learning_rate=LEARNING_RATE,
  num_train_steps=num_train_steps,
  num_warmup_steps=num_warmup_steps)

estimator = tf.estimator.Estimator(
  model_fn=model_fn,
  config=run_config,
  params={"batch_size": BATCH_SIZE})
```

```
INFO:tensorflow:Using config: {'_model_dir': '/GD/My Drive/Colab
Notebooks/LifeHackHeart/', '_tf_random_seed': None, '_save_summary_steps': 100,
'_save_checkpoints_steps': 300, '_save_checkpoints_secs': None,
'_session_config': allow_soft_placement: true
graph_options {
  rewrite_options {
    meta_optimizer_iterations: ONE
  }
}
, '_keep_checkpoint_max': 5, '_keep_checkpoint_every_n_hours': 10000,
'_log_step_count_steps': 100, '_train_distribute': None, '_device_fn': None,
'_protocol': None, '_eval_distribute': None, '_experimental_distribute': None,
'_experimental_max_worker_delay_secs': None, '_session_creation_timeout_secs':
7200, '_service': None, '_cluster_spec':
<tensorflow.python.training.server_lib.ClusterSpec object at 0x7faa4e893748>,
'_task_type': 'worker', '_task_id': 0, '_global_id_in_cluster': 0, '_master':
'', '_evaluation_master': '', '_is_chief': True, '_num_ps_replicas': 0,
'_num_worker_replicas': 1}

INFO:tensorflow:Using config: {'_model_dir': '/GD/My Drive/Colab
Notebooks/LifeHackHeart/', '_tf_random_seed': None, '_save_summary_steps': 100,
'_save_checkpoints_steps': 300, '_save_checkpoints_secs': None,
'_session_config': allow_soft_placement: true
graph_options {
  rewrite_options {
    meta_optimizer_iterations: ONE
  }
}
, '_keep_checkpoint_max': 5, '_keep_checkpoint_every_n_hours': 10000,
'_log_step_count_steps': 100, '_train_distribute': None, '_device_fn': None,
'_protocol': None, '_eval_distribute': None, '_experimental_distribute': None,
'_experimental_max_worker_delay_secs': None, '_session_creation_timeout_secs':
7200, '_service': None, '_cluster_spec':
<tensorflow.python.training.server_lib.ClusterSpec object at 0x7faa4e893748>,
'_task_type': 'worker', '_task_id': 0, '_global_id_in_cluster': 0, '_master':
'', '_evaluation_master': '', '_is_chief': True, '_num_ps_replicas': 0,
'_num_worker_replicas': 1}
```

we will now create an input builder function that takes our training feature set (`train_features`) and produces a generator. This is a pretty standard design pattern for working with Tensorflow Estimators.

[ ]:
```python
# Create an input function for training. drop_remainder = True for using TPUs.
train_input_fn = bert.run_classifier.input_fn_builder(
    features=train_features,
    seq_length=MAX_SEQ_LENGTH,
    is_training=True,
    drop_remainder=False)

# Create an input function for validating. drop_remainder = True for using TPUs.
val_input_fn = run_classifier.input_fn_builder(
    features=val_features,
    seq_length=MAX_SEQ_LENGTH,
    is_training=False,
    drop_remainder=False)
```

##Training & Evaluating

[ ]:
```python
#Training the model
print(f'Beginning Training!')
current_time = datetime.now()
estimator.train(input_fn=train_input_fn, max_steps=num_train_steps)
print("Training took time ", datetime.now() - current_time)
```

```
Beginning Training!
INFO:tensorflow:Calling model_fn.

INFO:tensorflow:Calling model_fn.

INFO:tensorflow:Saver not created because there are no variables in the graph to
restore

INFO:tensorflow:Saver not created because there are no variables in the graph to
restore
/usr/local/lib/python3.6/dist-
packages/tensorflow_core/python/framework/indexed_slices.py:424: UserWarning:
Converting sparse IndexedSlices to a dense Tensor of unknown shape. This may
consume a large amount of memory.
  "Converting sparse IndexedSlices to a dense Tensor of unknown shape. "

INFO:tensorflow:Done calling model_fn.
```

```
INFO:tensorflow:Done calling model_fn.

INFO:tensorflow:Create CheckpointSaverHook.

INFO:tensorflow:Create CheckpointSaverHook.

INFO:tensorflow:Graph was finalized.

INFO:tensorflow:Graph was finalized.

INFO:tensorflow:Running local_init_op.

INFO:tensorflow:Running local_init_op.

INFO:tensorflow:Done running local_init_op.

INFO:tensorflow:Done running local_init_op.

INFO:tensorflow:Saving checkpoints for 0 into /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt.

INFO:tensorflow:Saving checkpoints for 0 into /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt.

INFO:tensorflow:loss = 2.221311, step = 0

INFO:tensorflow:loss = 2.221311, step = 0

INFO:tensorflow:global_step/sec: 1.45645

INFO:tensorflow:global_step/sec: 1.45645

INFO:tensorflow:loss = 1.6858985, step = 100 (68.664 sec)

INFO:tensorflow:loss = 1.6858985, step = 100 (68.664 sec)

INFO:tensorflow:global_step/sec: 1.96087

INFO:tensorflow:global_step/sec: 1.96087

INFO:tensorflow:loss = 0.5017297, step = 200 (50.995 sec)

INFO:tensorflow:loss = 0.5017297, step = 200 (50.995 sec)

INFO:tensorflow:Saving checkpoints for 300 into /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt.

INFO:tensorflow:Saving checkpoints for 300 into /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt.

INFO:tensorflow:global_step/sec: 1.25624

INFO:tensorflow:global_step/sec: 1.25624

INFO:tensorflow:loss = 0.4210082, step = 300 (79.604 sec)

INFO:tensorflow:loss = 0.4210082, step = 300 (79.604 sec)

INFO:tensorflow:global_step/sec: 1.96069

INFO:tensorflow:global_step/sec: 1.96069
```

```
INFO:tensorflow:loss = 0.31012008, step = 400 (51.002 sec)

INFO:tensorflow:loss = 0.31012008, step = 400 (51.002 sec)

INFO:tensorflow:Saving checkpoints for 453 into /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt.

INFO:tensorflow:Saving checkpoints for 453 into /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt.

INFO:tensorflow:Loss for final step: 0.13591668.

INFO:tensorflow:Loss for final step: 0.13591668.

Training took time  0:06:29.224072
```

[ ]:
```python
#Evaluating the model with Validation set
eval_results = estimator.evaluate(input_fn=val_input_fn, steps=None)
```

```
INFO:tensorflow:Calling model_fn.

INFO:tensorflow:Calling model_fn.

INFO:tensorflow:Saver not created because there are no variables in the graph to
restore

INFO:tensorflow:Saver not created because there are no variables in the graph to
restore
/usr/local/lib/python3.6/dist-
packages/tensorflow_core/python/framework/indexed_slices.py:424: UserWarning:
Converting sparse IndexedSlices to a dense Tensor of unknown shape. This may
consume a large amount of memory.
  "Converting sparse IndexedSlices to a dense Tensor of unknown shape. "

INFO:tensorflow:Done calling model_fn.

INFO:tensorflow:Done calling model_fn.

INFO:tensorflow:Starting evaluation at 2020-07-27T20:37:00Z

INFO:tensorflow:Starting evaluation at 2020-07-27T20:37:00Z

INFO:tensorflow:Graph was finalized.

INFO:tensorflow:Graph was finalized.

INFO:tensorflow:Restoring parameters from /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt-453

INFO:tensorflow:Restoring parameters from /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt-453

INFO:tensorflow:Running local_init_op.

INFO:tensorflow:Running local_init_op.

INFO:tensorflow:Done running local_init_op.
```

```
INFO:tensorflow:Done running local_init_op.

INFO:tensorflow:Finished evaluation at 2020-07-27-20:38:05

INFO:tensorflow:Finished evaluation at 2020-07-27-20:38:05

INFO:tensorflow:Saving dict for global step 453: eval_accuracy = 0.630854,
false_negatives = 43.0, false_positives = 19.0, global_step = 453, loss =
1.2297496, true_negatives = 31.0, true_positives = 270.0

INFO:tensorflow:Saving dict for global step 453: eval_accuracy = 0.630854,
false_negatives = 43.0, false_positives = 19.0, global_step = 453, loss =
1.2297496, true_negatives = 31.0, true_positives = 270.0

INFO:tensorflow:Saving 'checkpoint_path' summary for global step 453: /GD/My
Drive/Colab Notebooks/LifeHackHeart/model.ckpt-453

INFO:tensorflow:Saving 'checkpoint_path' summary for global step 453: /GD/My
Drive/Colab Notebooks/LifeHackHeart/model.ckpt-453
```

```
[ ]: eval_results
```

```
[ ]: {'eval_accuracy': 0.630854,
      'false_negatives': 43.0,
      'false_positives': 19.0,
      'global_step': 453,
      'loss': 1.2297496,
      'true_negatives': 31.0,
      'true_positives': 270.0}
```

```
[ ]: predictions = estimator.predict(val_input_fn)
```

```
[ ]: preds_result = []
     for prediction in predictions:
       preds_result.append((prediction['probabilities'], prediction['labels']))
```

```
INFO:tensorflow:Calling model_fn.

INFO:tensorflow:Calling model_fn.

INFO:tensorflow:Saver not created because there are no variables in the graph to
restore

INFO:tensorflow:Saver not created because there are no variables in the graph to
restore

INFO:tensorflow:Done calling model_fn.

INFO:tensorflow:Done calling model_fn.

INFO:tensorflow:Graph was finalized.

INFO:tensorflow:Graph was finalized.
```

```
INFO:tensorflow:Restoring parameters from /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt-453
```

```
INFO:tensorflow:Restoring parameters from /GD/My Drive/Colab
Notebooks/LifeHackHeart/model.ckpt-453
```

```
INFO:tensorflow:Running local_init_op.
```

```
INFO:tensorflow:Running local_init_op.
```

```
INFO:tensorflow:Done running local_init_op.
```

```
INFO:tensorflow:Done running local_init_op.
```

```
[ ]: y_pred = list(map(lambda x: x[1], preds_result))
```

```
[ ]: mapping = dict()

     for i in range(len(label_list)):
       mapping[label_list[i]] = i

     y_actual = list(map(lambda x: mapping[x], val['category'].tolist()))
```

```
[ ]: from sklearn.metrics import confusion_matrix

     confusion_matrix(y_actual, y_pred)
```

```
[ ]: array([[32,  2, 11,  0,  4,  1,  0,  0,  0,  0],
            [ 2, 38, 13,  2,  2,  1,  0,  0,  0,  0],
            [12,  8, 86,  2,  6,  4,  1,  0,  0,  0],
            [ 5,  0,  0, 14,  0,  0,  1,  0,  0,  0],
            [ 8,  1,  3,  2, 46,  0,  1,  0,  0,  0],
            [ 7,  2,  9,  0,  1,  7,  1,  0,  0,  0],
            [ 3,  0,  2,  3,  0,  0,  8,  0,  0,  0],
            [ 0,  0,  0,  0,  1,  0,  0,  0,  0,  0],
            [ 4,  0,  1,  0,  0,  1,  0,  0,  0,  0],
            [ 1,  2,  1,  0,  0,  1,  0,  0,  0,  0]])
```

```
[ ]: val_pred = val.copy()
     val_pred['pred'] = list(map(lambda x: label_list[x], y_pred))
     val_pred.to_csv('prediction.csv')
```

```
[ ]: val_pred.head()
```

```
[ ]:       Unnamed: 0  …             pred
     1758        1758  …  serangan-jantung
     1283        1283  …           aritmia
     1404        1404  …  serangan-jantung
     31            31  …     gagal-jantung
     625          625  …           aritmia
```

```
[5 rows x 7 columns]
```

#Reference: Most of the code has been taken from the following resource:

- https://colab.research.google.com/github/google-research/bert/blob/master/predicting_movie_reviews_wi