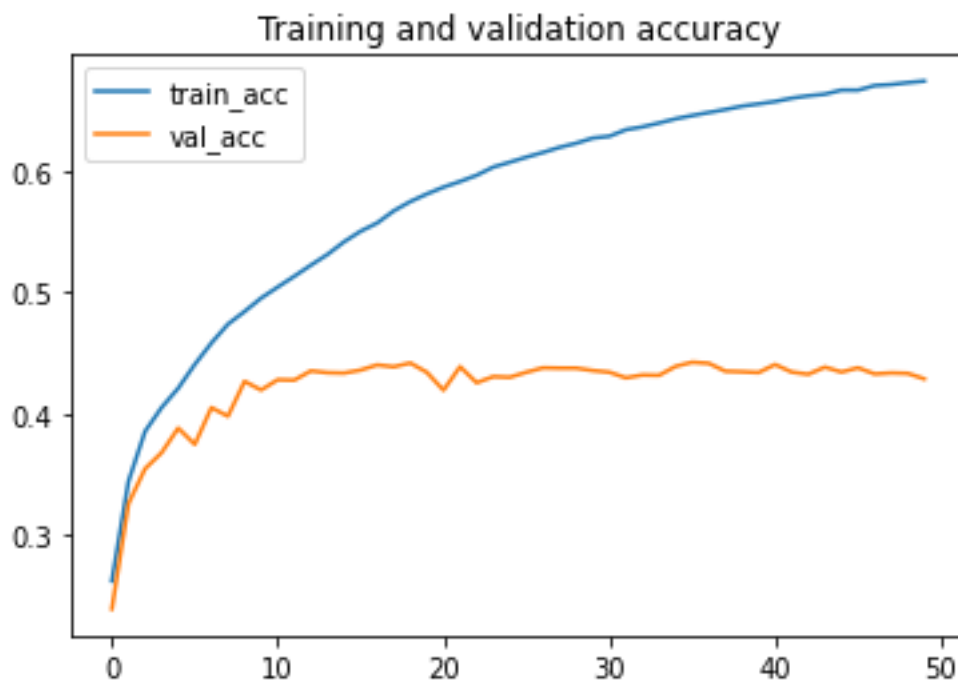
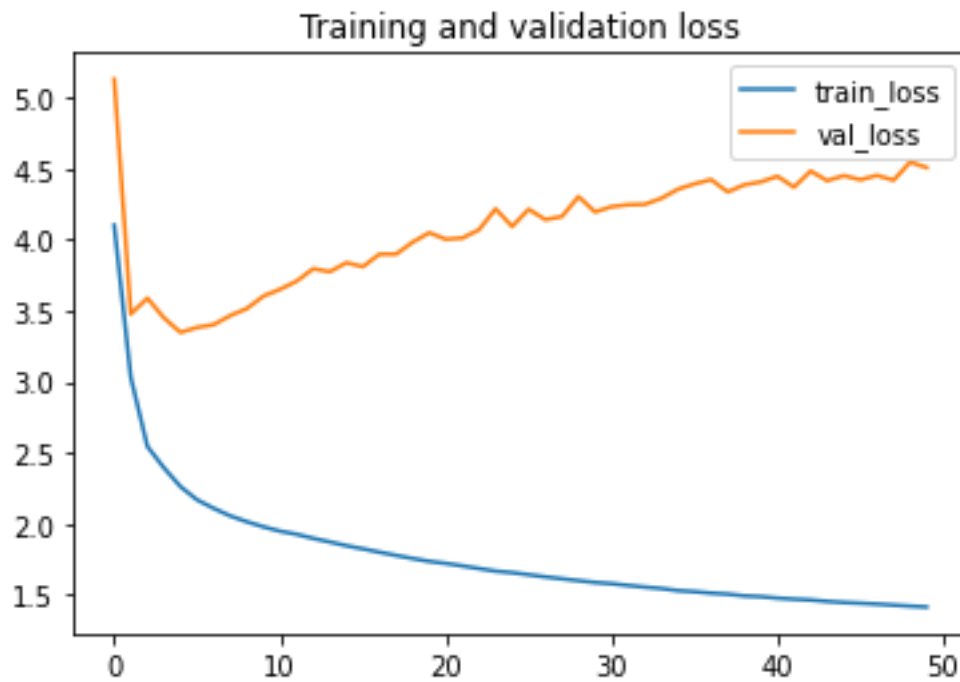


Note : All of these tests are preliminary and are not mentioned in results_final_phase.xlsx

EX 1:

```
DROPOUT_RATE = 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 256
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 TrainedByUs
Train_size = 215359
Val_size = 121512
Network = default-baseline
After VGG Normalization = False
optimizer='rmsprop', loss='categorical_crossentropy'
Filter in tokenizer= true
```





Ex2:

BATCH_SIZE = 256

SEQ_LENGTH = 26

VOCAB_SIZE = --

EMBEDDING_TYPE = Glove.6b.300d.e

EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Paper Pretrained

Train_size = 215359

Val_size = 121512

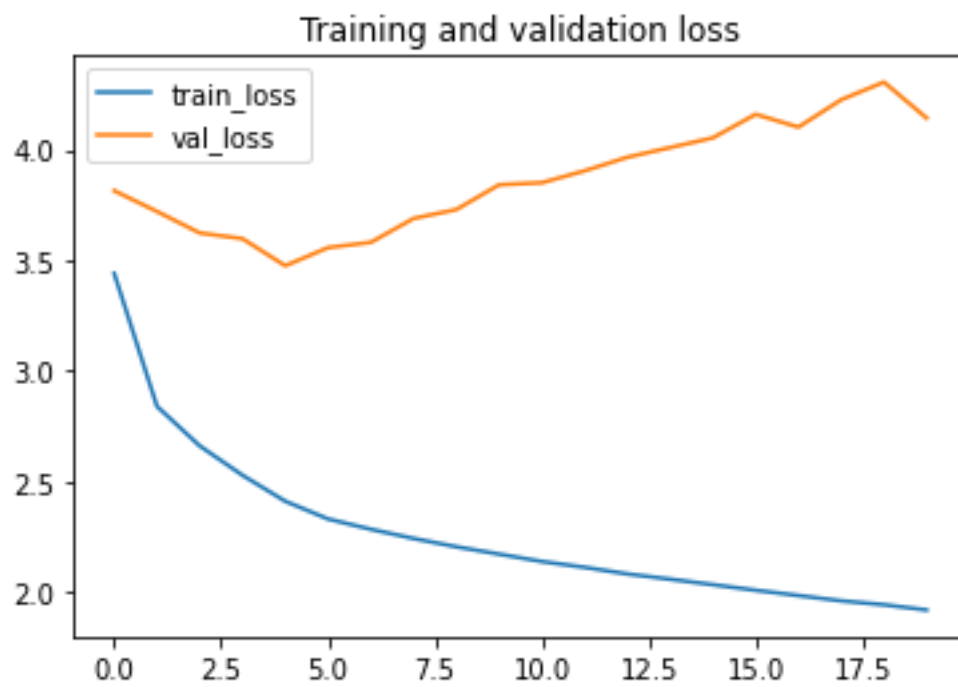
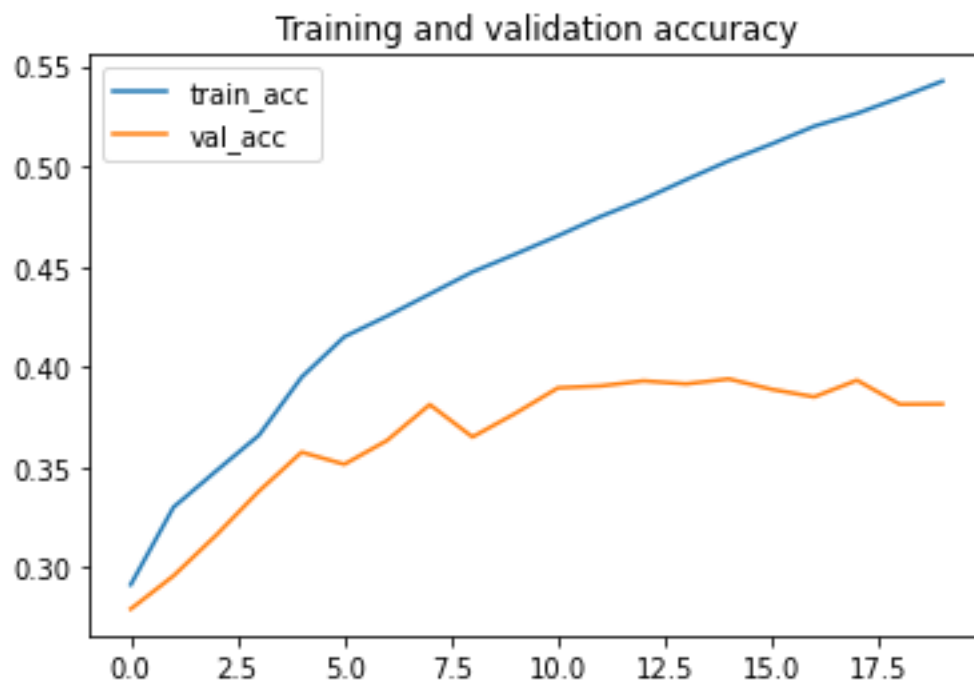
Network = default-baseline

After VGG Normalization = False

optimizer='rmsprop', loss='categorical_crossentropy'

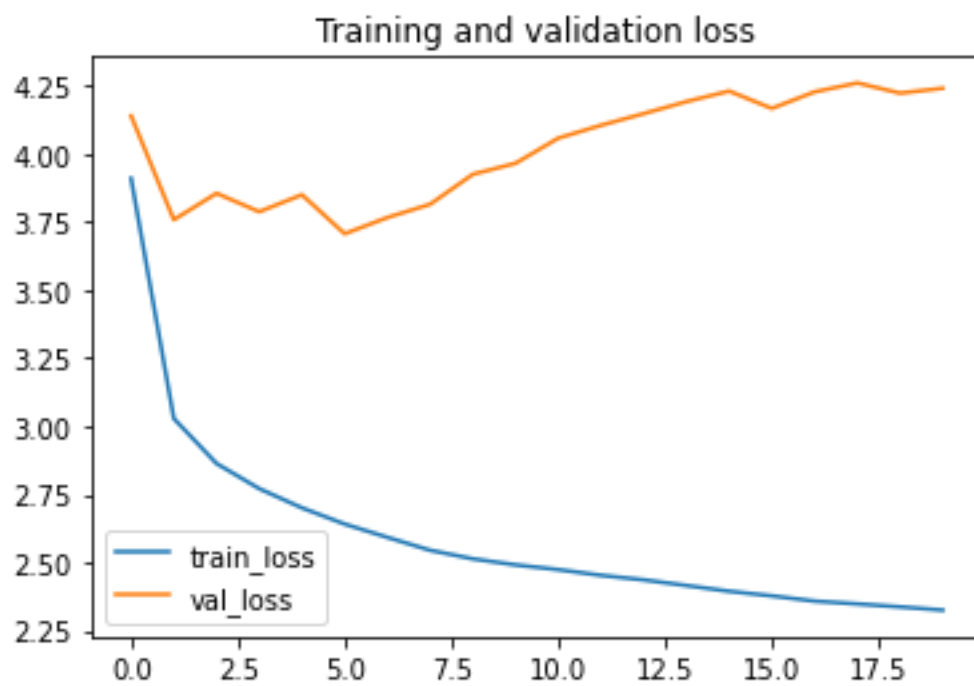
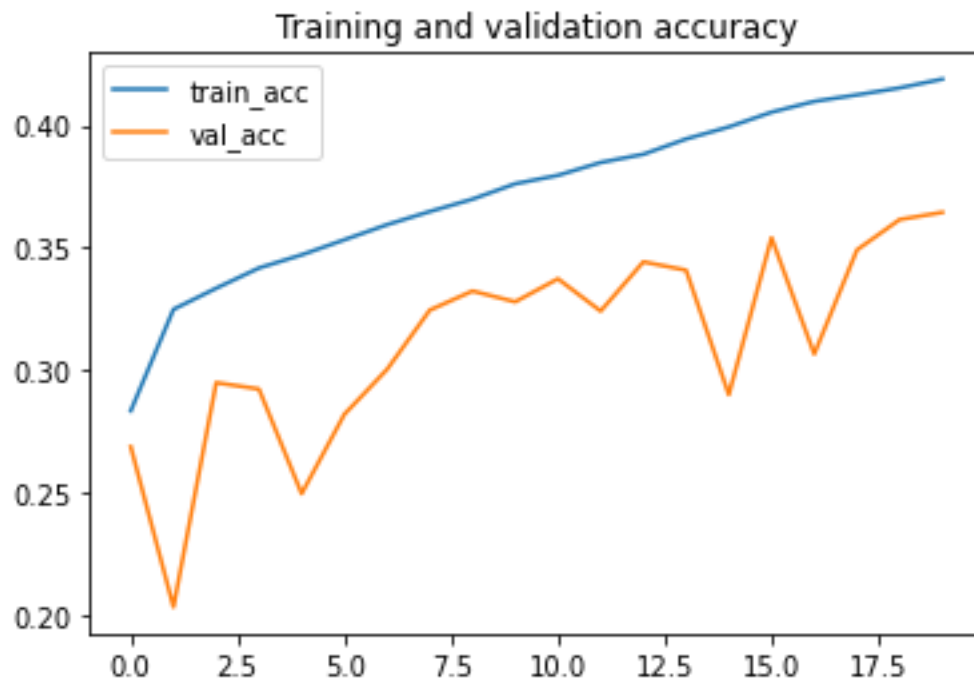
Filter in tokenizer= False

Epoch = 20



Ex3:

=ex2 without any embedding



Ex4:

Vanilla DROPOUT_RATE = 0.4 → 0.6
EMBEDDING_DIM = 300

```
EPOCHS = 20
BATCH_SIZE = 64
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en
```

Image_Features = VGG19 Pretrained

Train_size = 20000

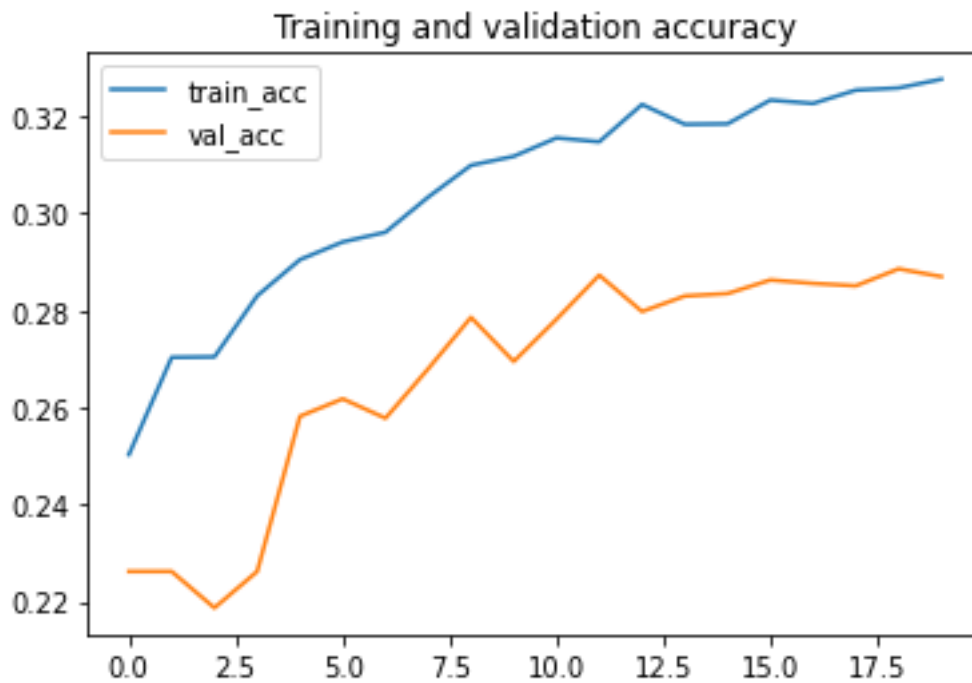
Val_size = 10000

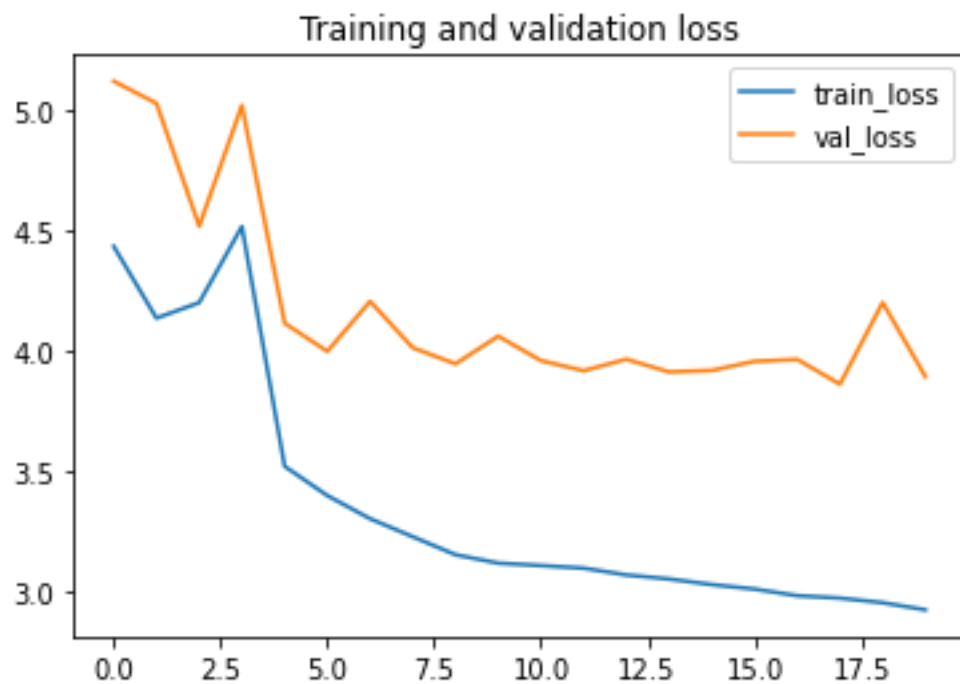
Network = default-baseline

After VGG Normalization = False

optimizer='rmsprop', loss='categorical_crossentropy'

Filter in tokenizer= false





Ex4:

Vanilla DROPOUT_RATE = 0.4 → 0.6

EMBEDDING_DIM = 300

EPOCHS = 20

BATCH_SIZE = 64

SEQ_LENGTH = 26

VOCAB_SIZE = --

EMBEDDING_TYPE = Glove.6b.300d.e

EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained + L2 Normalized

Train_size = 50000

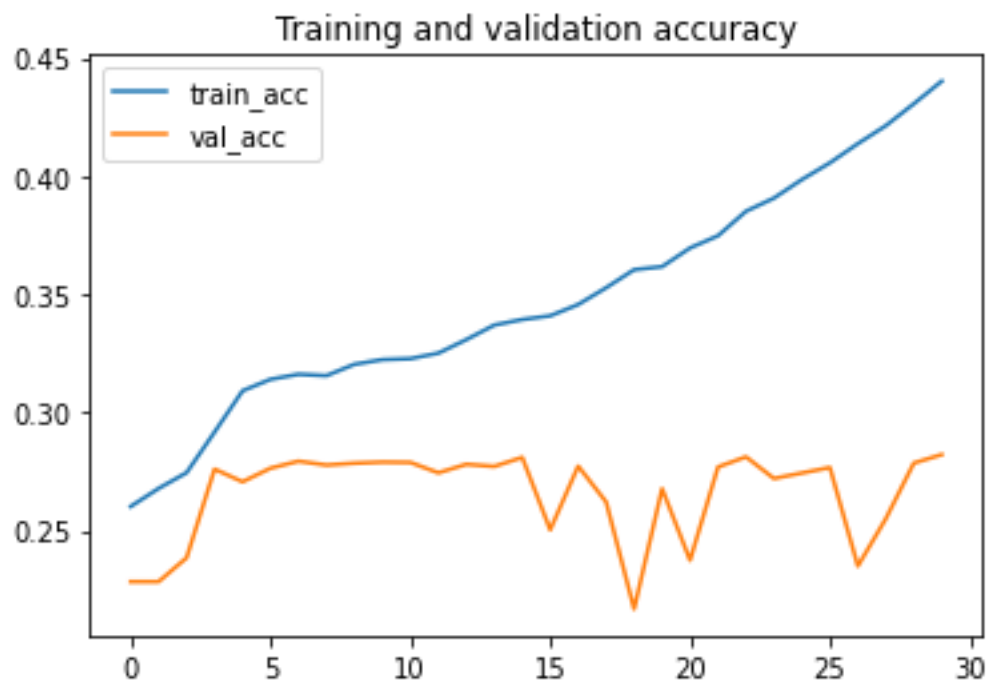
Val_size = 25000

Network = default-baseline

After VGG Normalization = true

optimizer='rmsprop', loss='categorical_crossentropy'

Filter in tokenizer= false



Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 500

```

```
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained + L2 Normalized

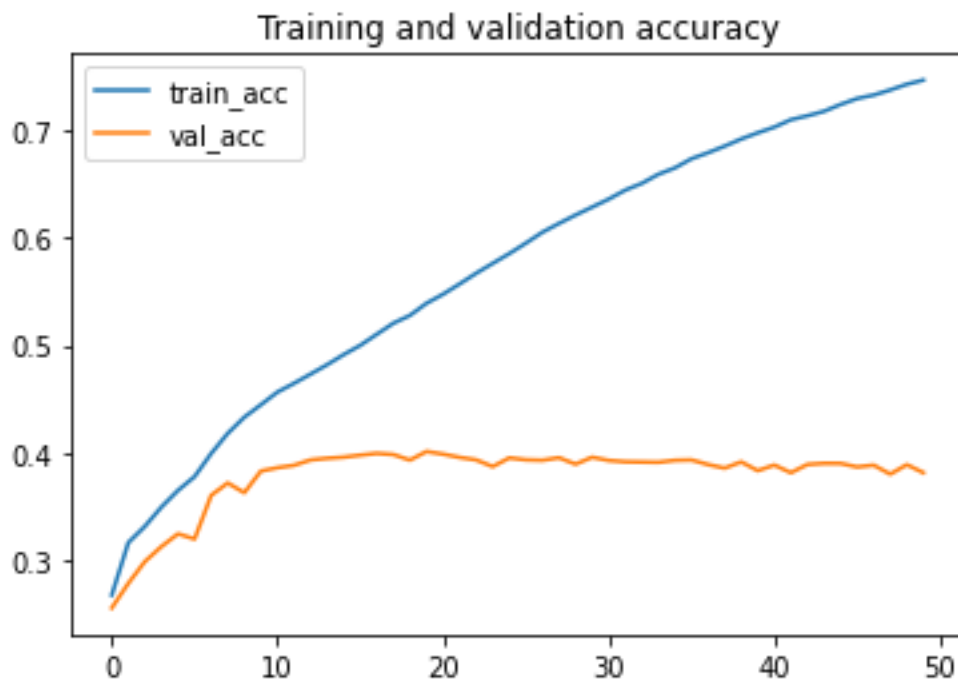
Train_size = all

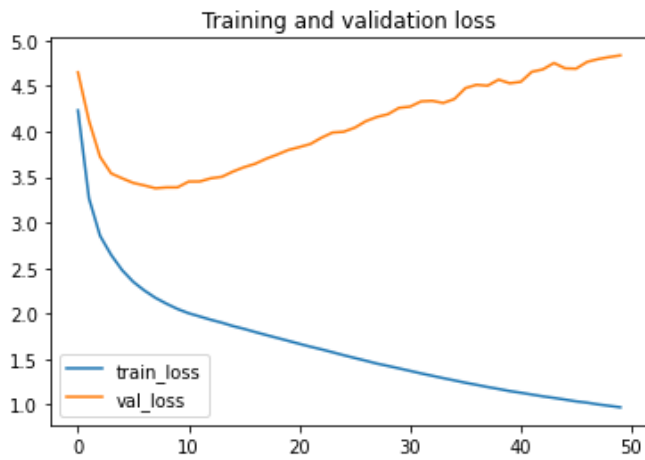
Val_size = all

Network = default-baseline

After VGG Normalization = true

optimizer='rmsprop', loss='categorical_crossentropy'
Filter in tokenizer= false
```





Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 25
BATCH_SIZE = 500
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained + L2 Normalized

Train_size = 20k

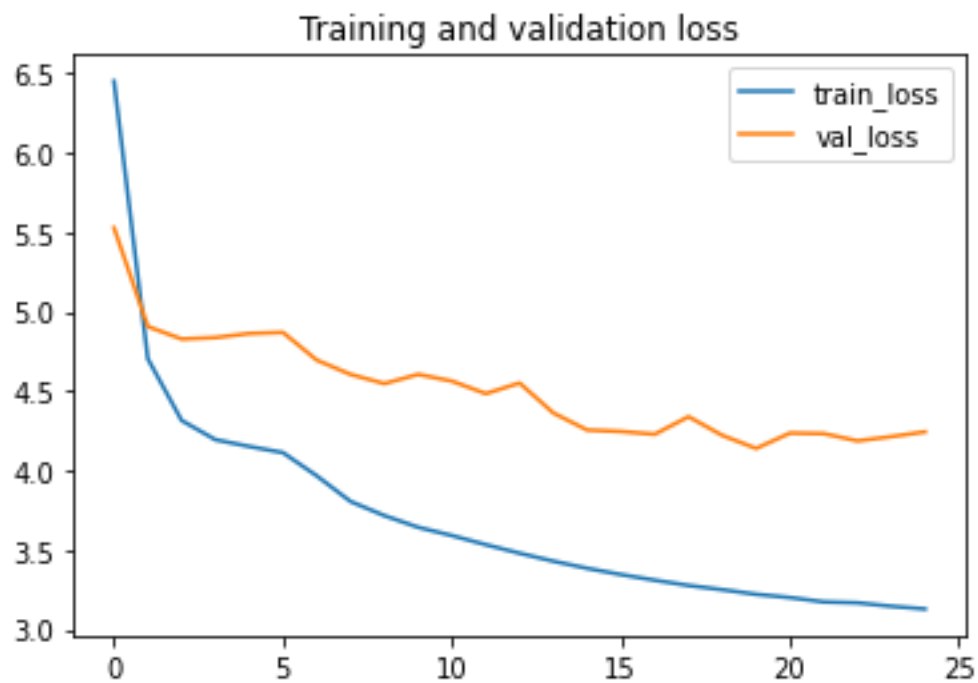
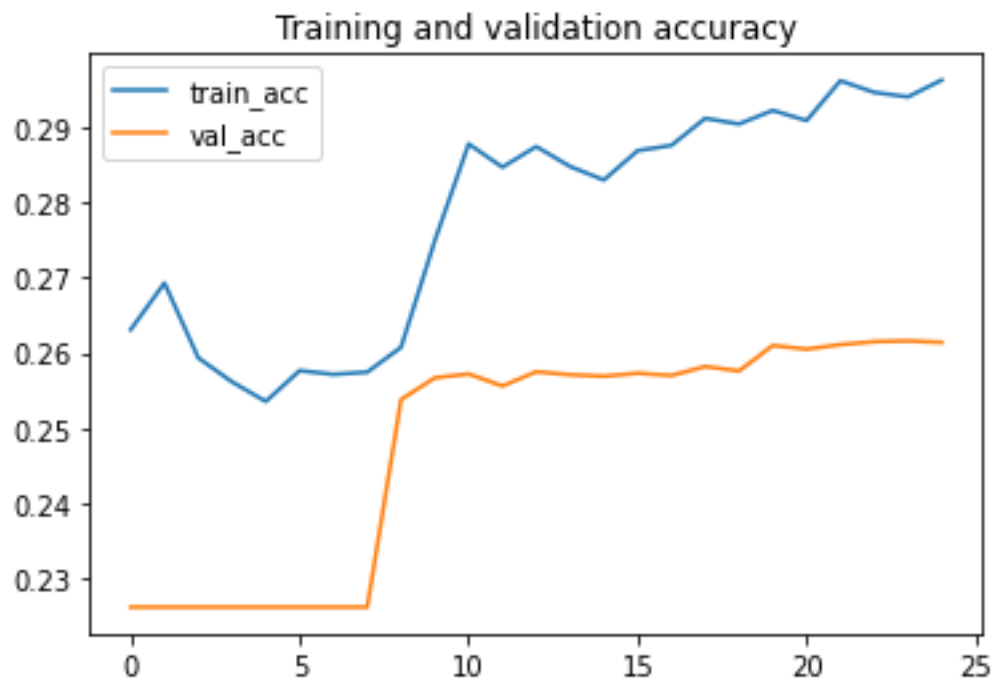
Val_size = 10k

Network = default-baseline

After VGG Normalization = true

optimizer='rmsprop', loss='categorical_crossentropy'
Filter in tokenizer= false

```



Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 75
  
```

```
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en
```

Image_Features = VGG19 Pretrained + L2 Normalized

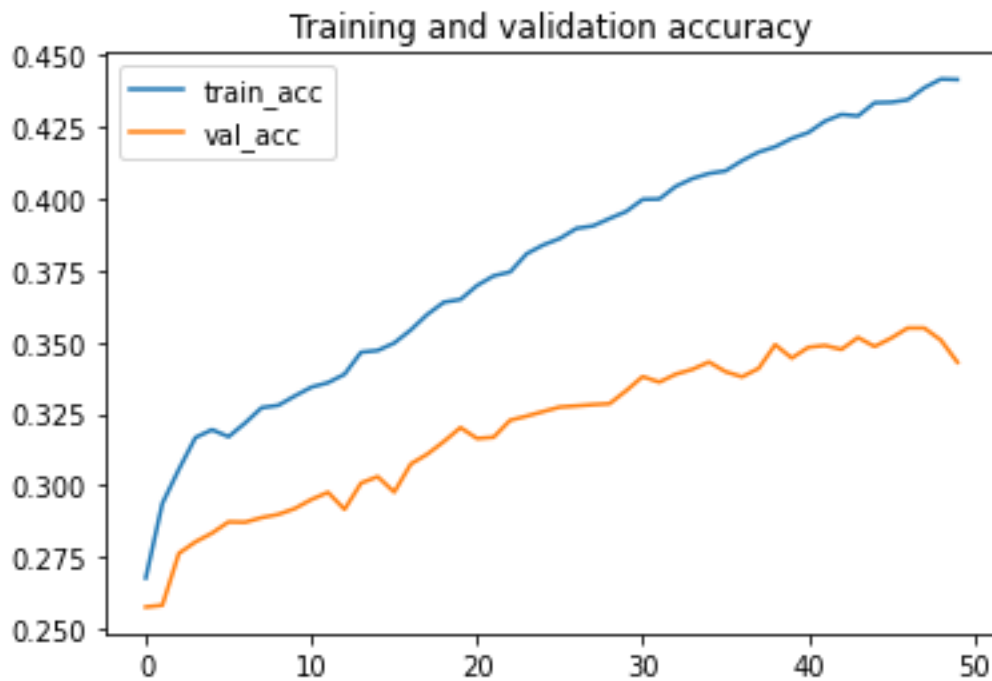
Train_size = 50k

Val_size = 25k

Network = default-baseline

After VGG Normalization = true

```
opt=tf.keras.optimizers.RMSprop(lr=0.00004)
, loss='categorical_crossentropy'
Filter in tokenizer= false
```





loss: 2.4463 - accuracy: 0.4414 - val_loss: 3.4854 - val_accuracy: 0.3429

Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 1024
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained + L2 Normalized

Train_size = 50k

Val_size = 25k

```

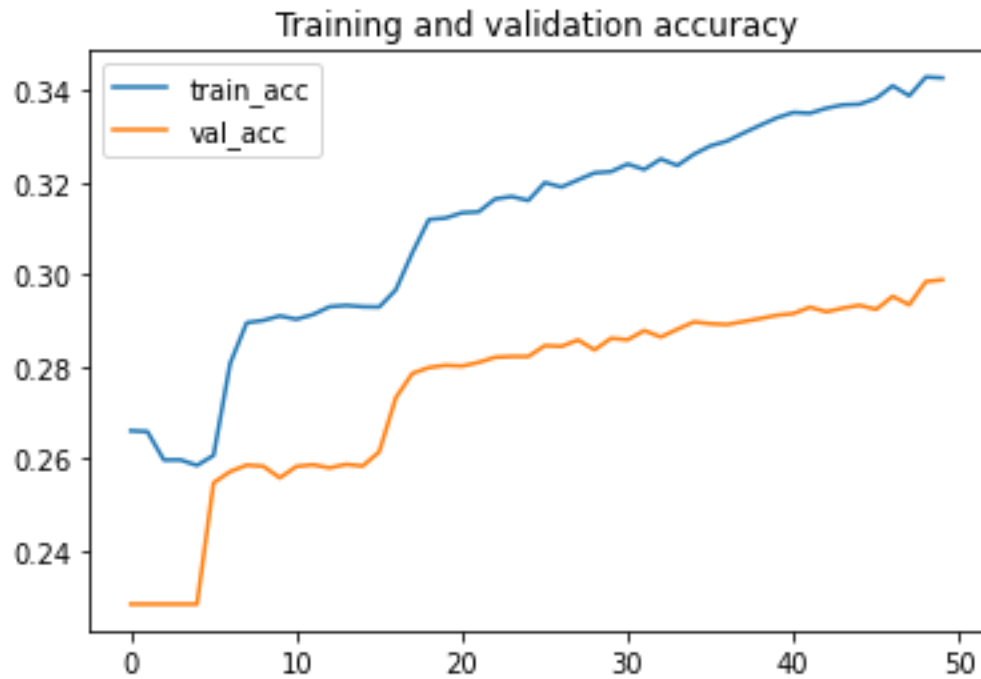
Network = default-baseline

After VGG Normalization = true

```

opt=tf.keras.optimizers.RMSprop(lr=0.00004)
, loss='categorical_crossentropy'
Filter in tokenizer= false

```



loss: 2.6415 - accuracy: 0.3427 - val_loss: 3.8126 - val_accuracy: 0.2989

Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --

```

```
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained + L2 Normalized

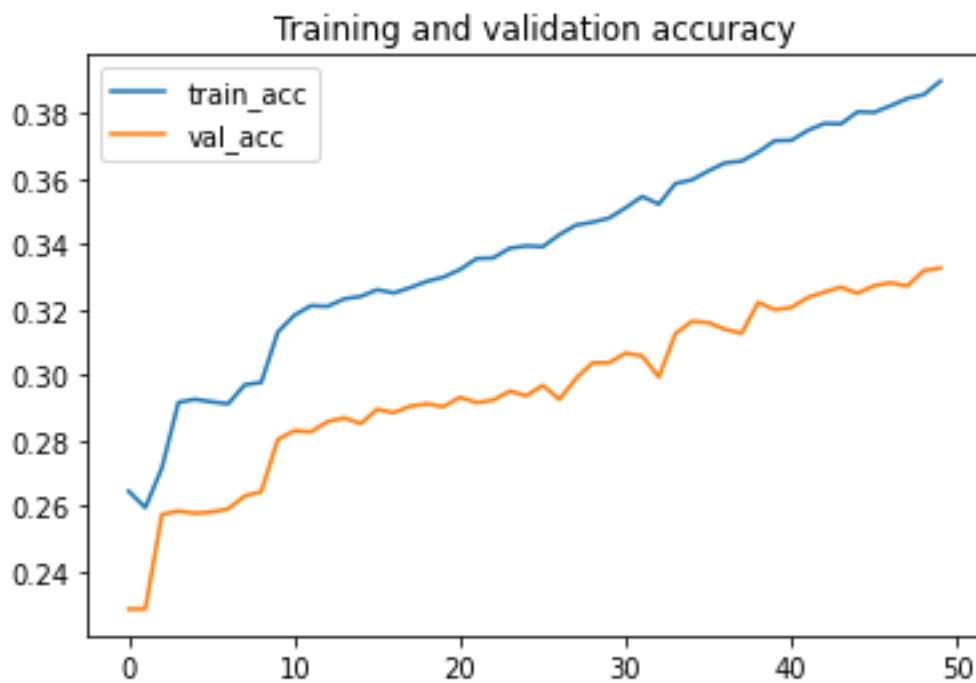
Train_size = 50k

Val_size = 25k

Network = default-baseline

After VGG Normalization = true

opt=tf.keras.optimizers.RMSprop(lr=0.00004)
, loss='categorical_crossentropy'
Filter in tokenizer= false
```





loss: 2.4531 - accuracy: 0.3899 - val_loss: 3.4347 - val_accuracy: 0.3327

Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.e
EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained + L2 Normalized

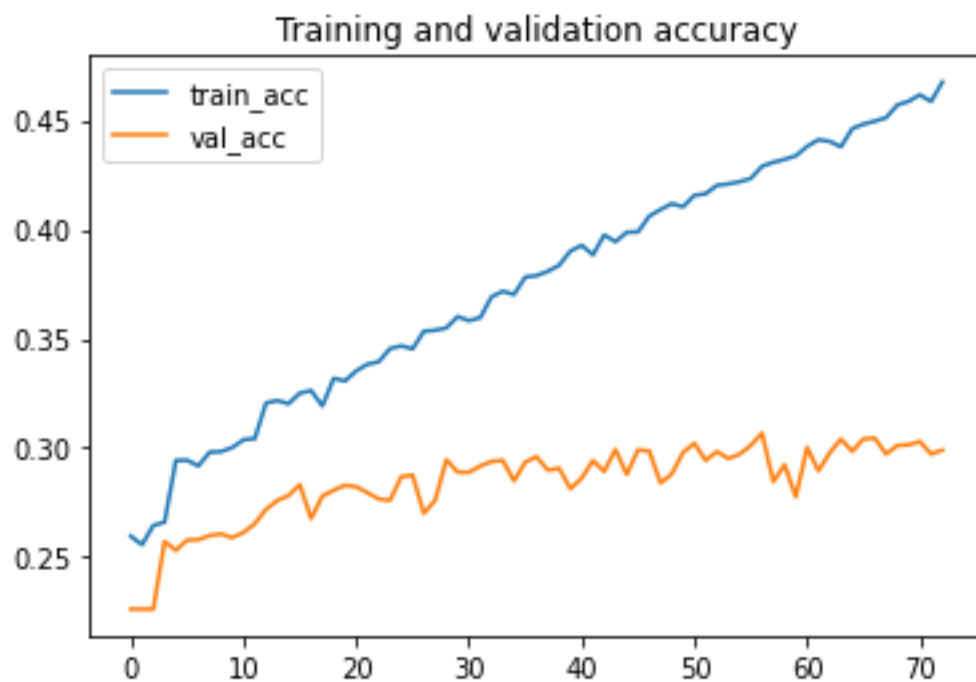
Train_size = 20k
Val_size = 10k

Network = default-baseline

After VGG Normalization = true

opt= lr_schedule =true
, loss='categorical_crossentropy'
Filter in tokenizer= false

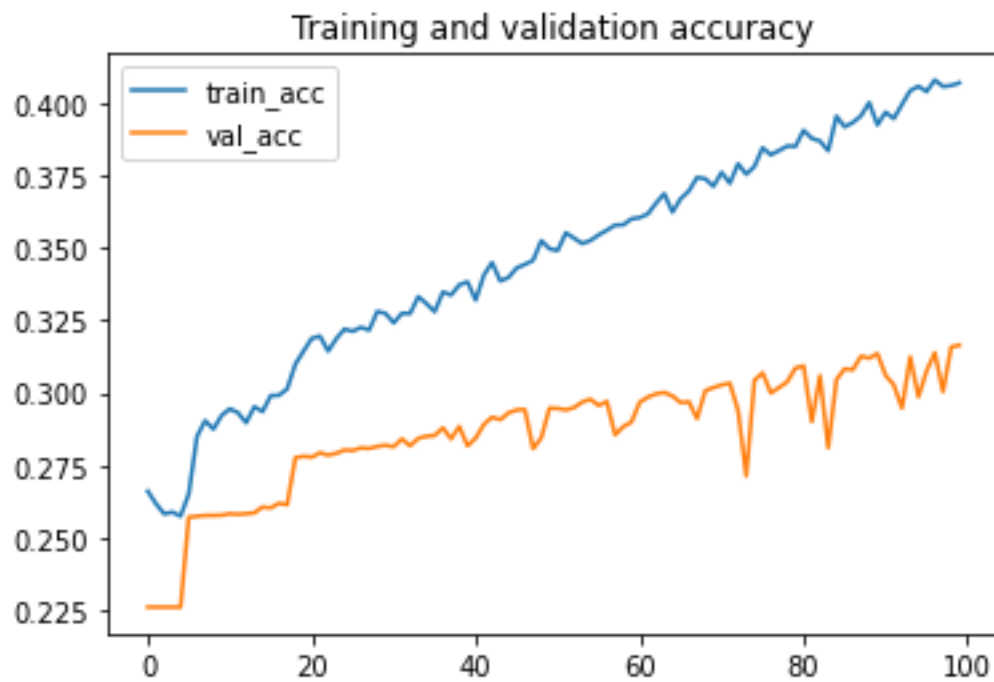
```



loss: 2.2042 - accuracy: 0.4676 - val_loss: 3.7185 - val_accuracy: 0.2990

Ex:

```
opt=tf.keras.optimizers.RMSprop(lr=0.00004)
```

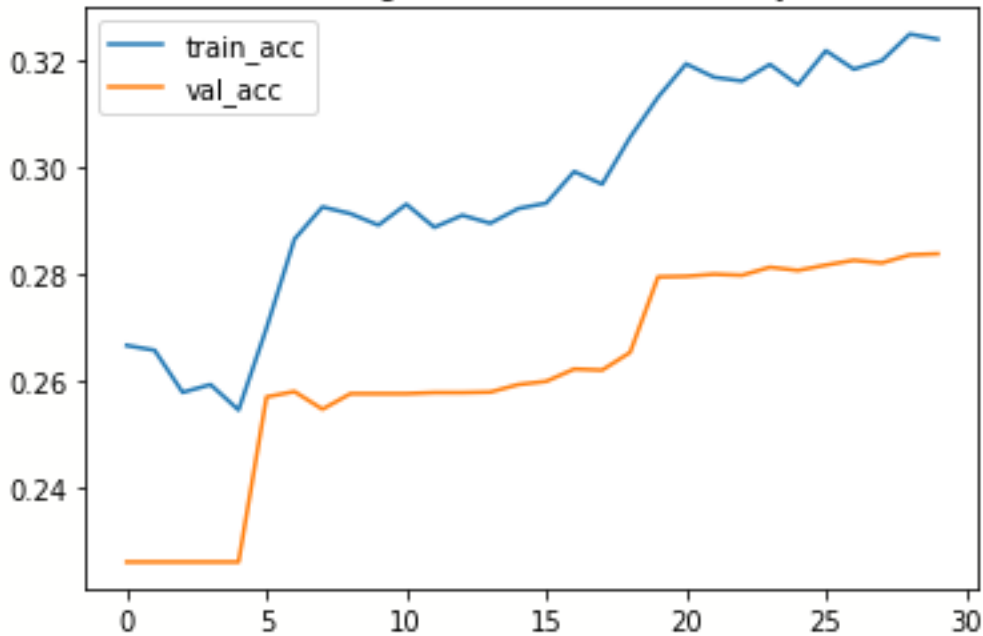
accuracy: 0.3956 - val_loss: 3.7801 - val_accuracy: 0.3127

Ex:

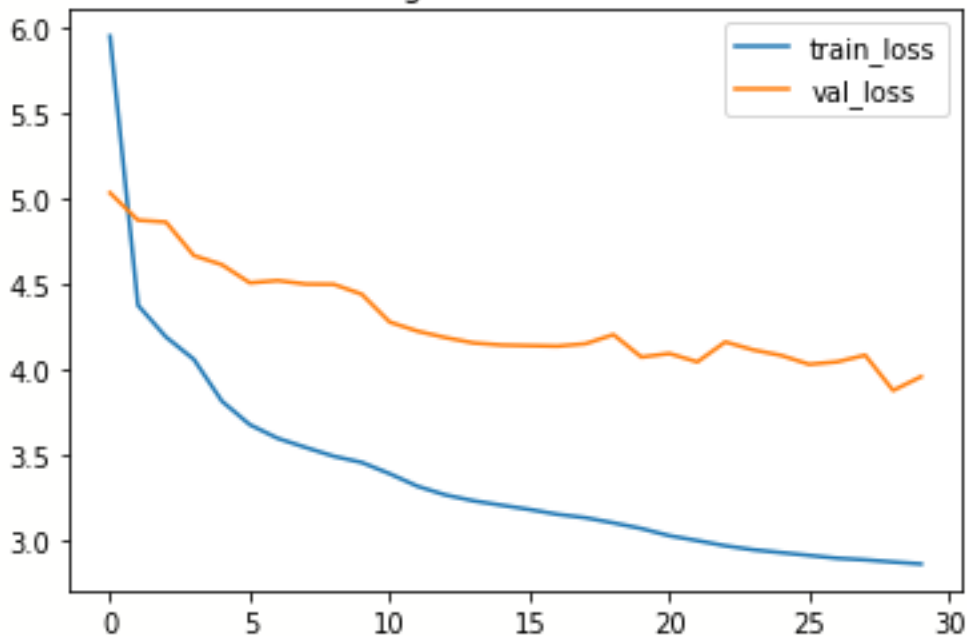
Vanilla DROPOUT_RATE = 0.5 → 0.5
 EMBEDDING_DIM = 300
 EPOCHS = 30

```
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 Pretrained
Train_size = 20k
Val_size = 10k
Network = default-baseline
After VGG Normalization = true
opt= lr =0.0001 , loss='categorical_crossentropy'
Filter in tokenizer= false
```

Training and validation accuracy



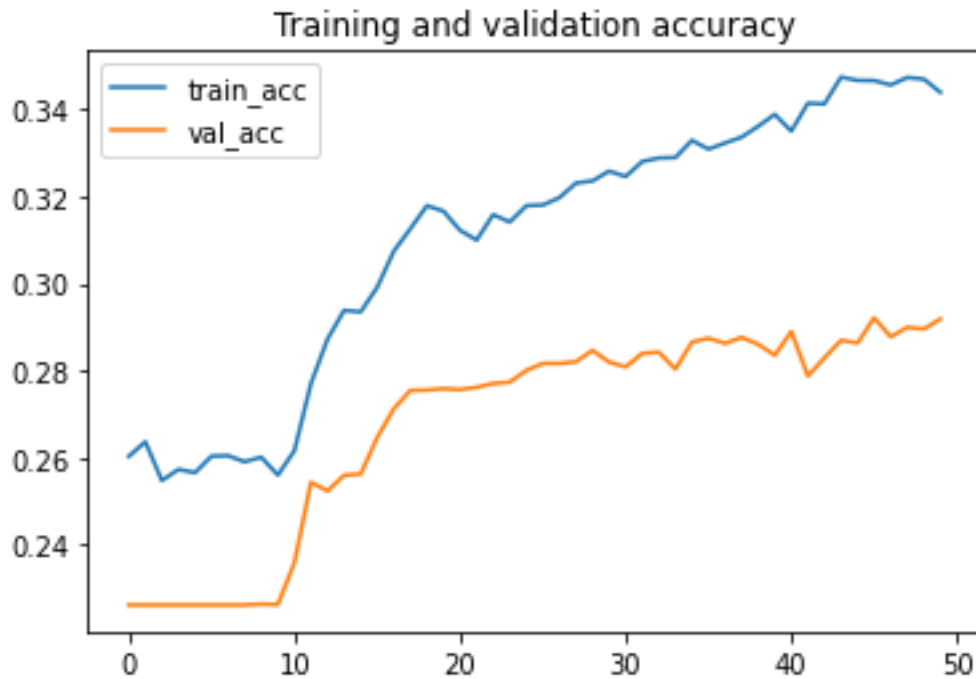
Training and validation loss



Val acc= 28.2

Ex:

```
Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 50
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 Pretrained
Train_size = 20k
Val_size = 10k
Network = default-baseline
After VGG Normalization = true
opt= lr=0.0001 , loss='categorical_crossentropy'
Filter in tokenizer= false
Text padding = pre
```





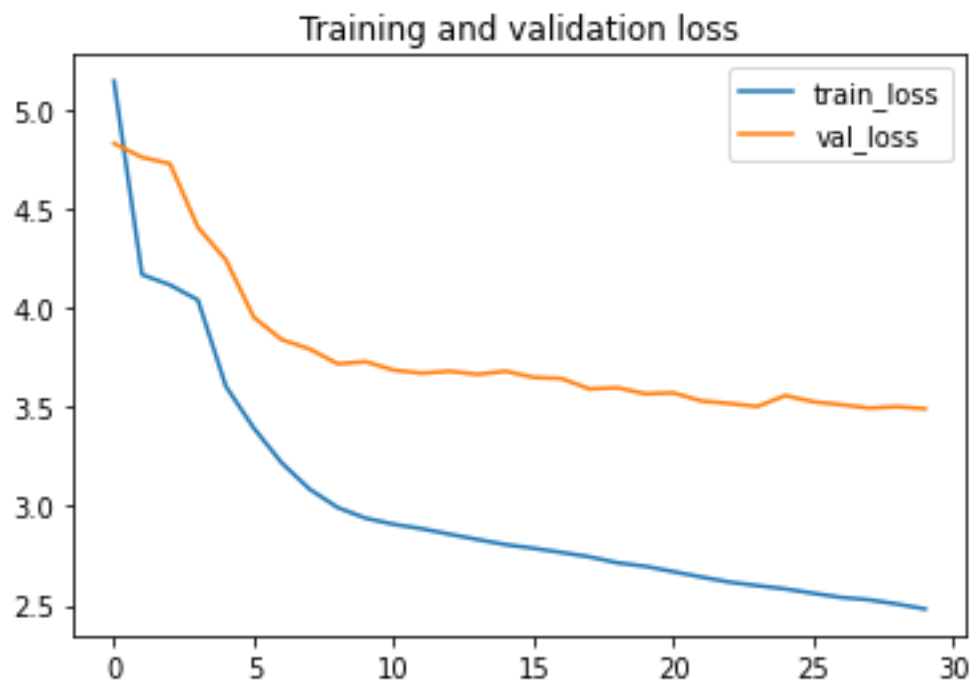
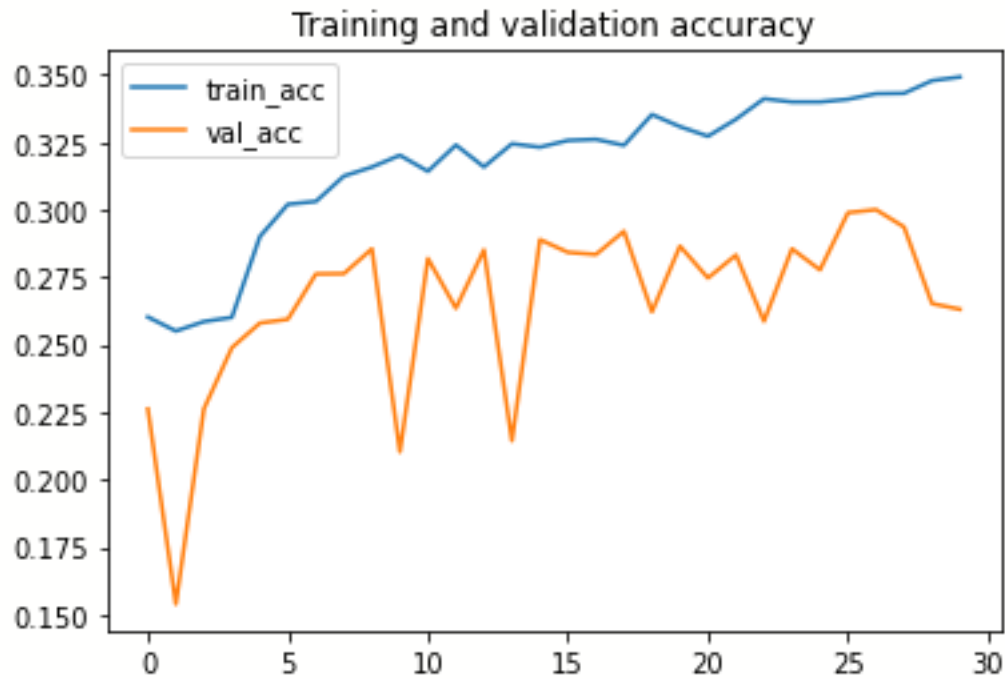
loss: 2.8339 - accuracy: 0.3329 - val_loss: 3.7751 - val_accuracy: 0.2866

Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 30
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 Pretrained
Train_size = 20k
Val_size = 10k
Network = default-baseline
After VGG Normalization = true
opt= lr=0.0001 , loss='categorical_crossentropy'
Filter in tokenizer= false
Text padding = pre
opt=tf.keras.optimizers.Adadelta(lr=1)

```



loss: 2.4811 - accuracy: 0.3492 - val_loss: 3.4903 - val_accuracy: 0.2631

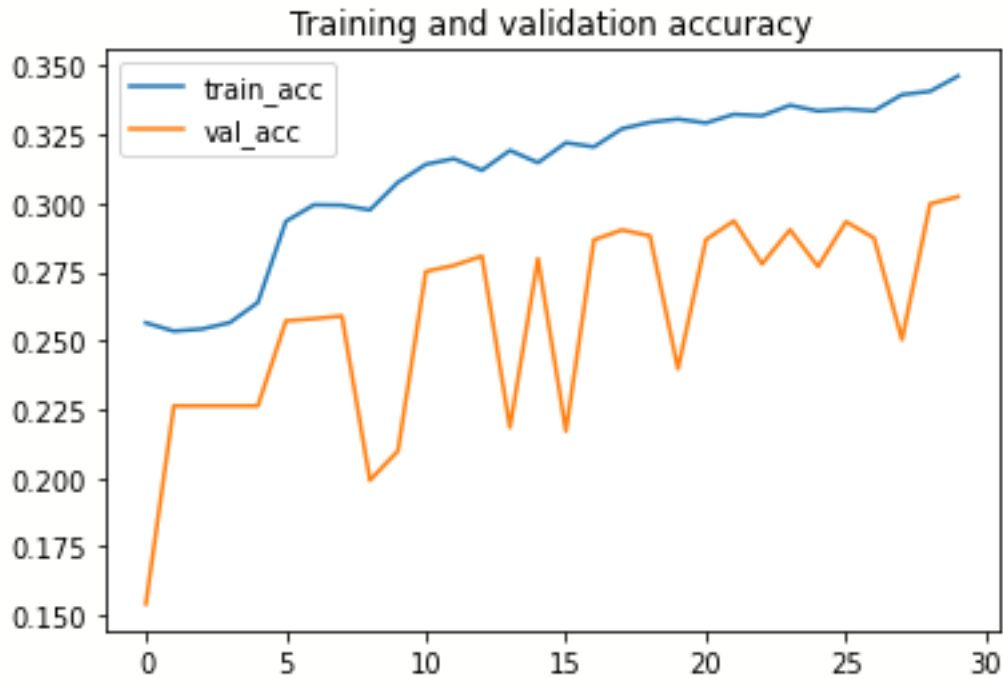
Ex:

```

Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 30
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 Pretrained
Train_size = 20k

```

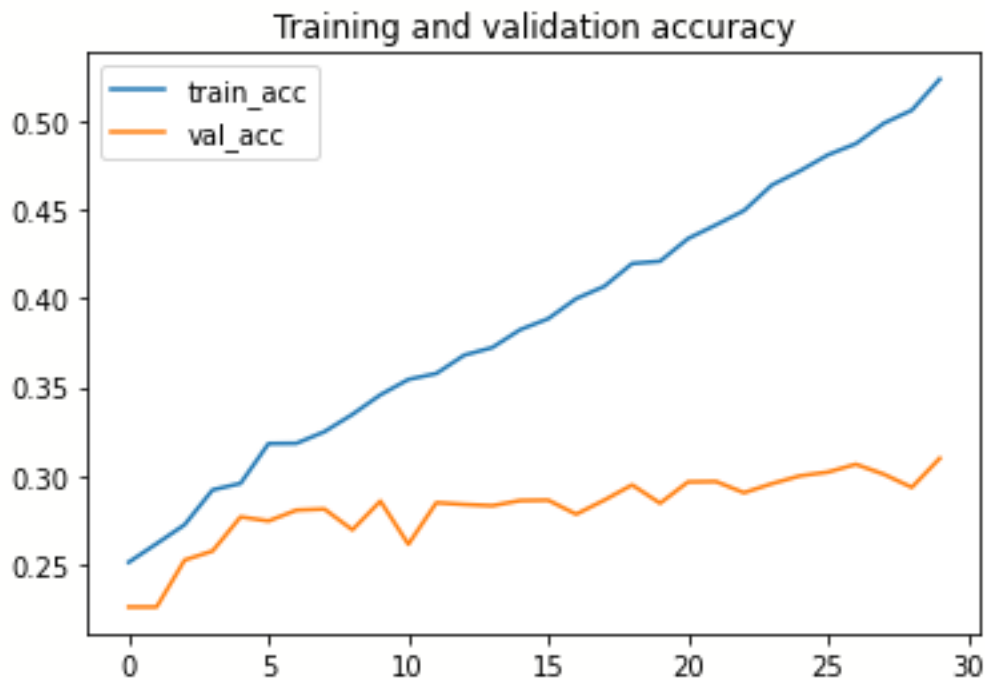
```
Val_size = 10k
Network = default-baseline
After VGG Normalization = true
opt= lr =0.0001 , loss='categorical_crossentropy'
Filter in tokenizer= false
Text padding = pre
opt=tf.keras.optimizers.Adadelta(lr=1)
tex data feeded to network (questions)= exactly the same as paper
```

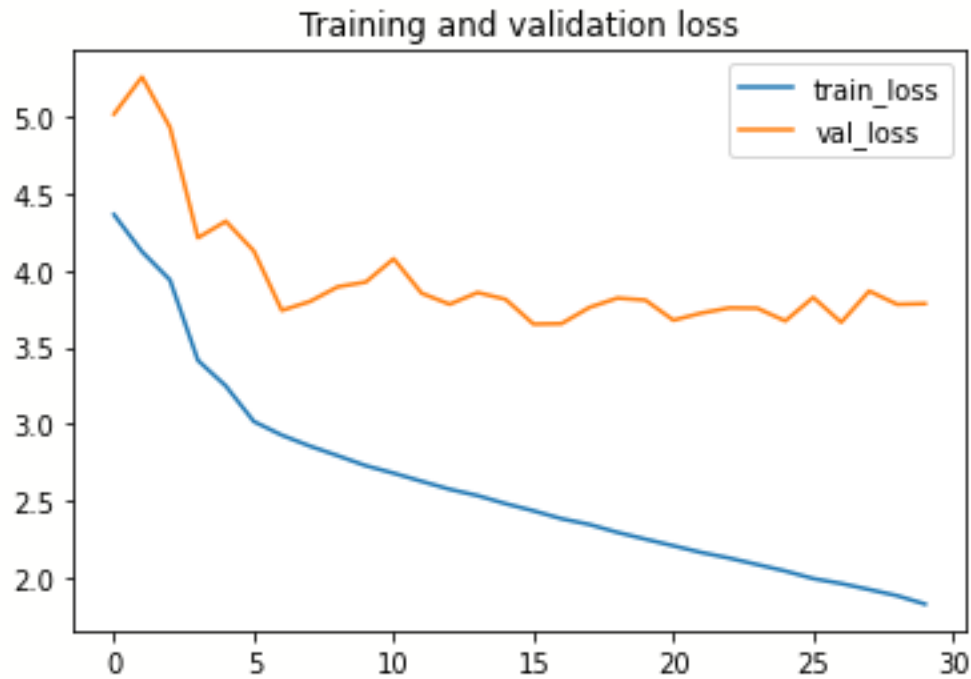


accuracy: 0.3462 - val_loss: 3.4552 - val_accuracy: 0.3023

ex:

```
Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 30
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 Pretrained
Train_size = 20k
Val_size = 10k
Network = default-baseline
After VGG Normalization = true
opt= lr =0.0001 , loss='categorical_crossentropy'
Filter in tokenizer= false
Text padding = pre
opt=tf.keras.optimizers.rmsprop(lr=0.0004)
tex data fedded to network (questions)= exactly the same as paper
```





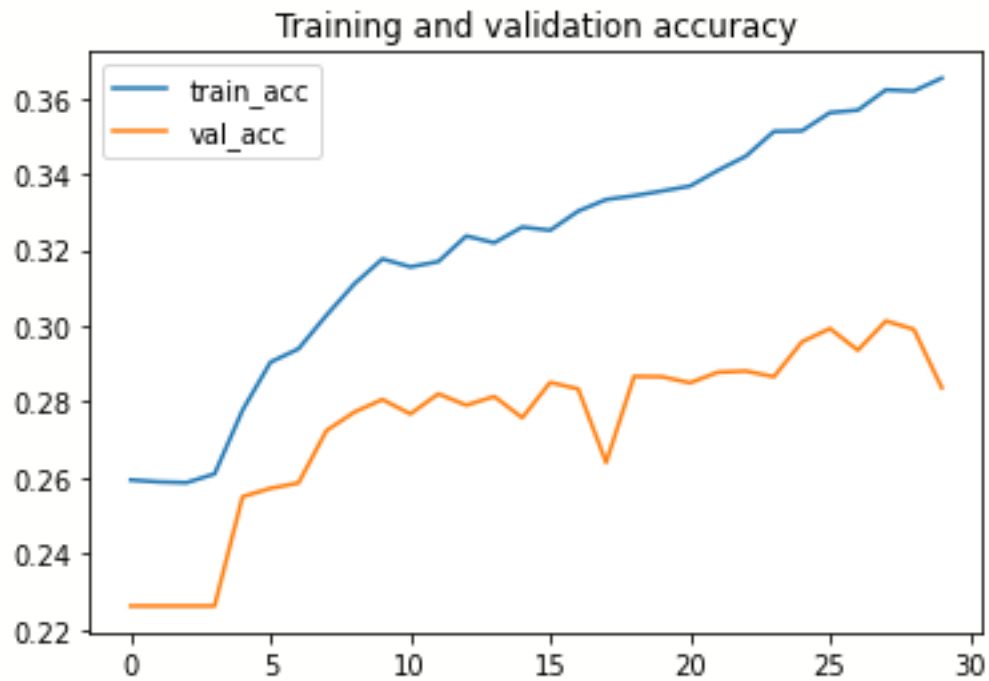
accuracy: 0.5233 - val_loss: 3.7841 - val_accuracy: 0.3099

ex:

```
lr_schedule = tf.keras.optimizers.schedules.ExponentialDecay(
    initial_learning_rate=1e-4,
    decay_steps=300,
    decay_rate=math.exp(math.log(0.1)/learning_rate_decay_every/steps_per_epoch))#0.9
opt = tf.keras.optimizers.RMSprop(learning_rate=lr_schedule)
```

```
Vanilla DROPOUT_RATE = 0.5 → 0.5
EMBEDDING_DIM = 300
EPOCHS = 30
BATCH_SIZE = 300
SEQ_LENGTH = 26
VOCAB_SIZE = --
EMBEDDING_TYPE = Glove.6b.300d.en
Image_Features = VGG19 Pretrained
Train_size = 20k
Val_size = 10k
Network = default-baseline
After VGG Normalization = true
loss='categorical_crossentropy'
Filter in tokenizer= false
Text padding = pre
```


tex data feeded to network (questions)= exactly the same as paper



val loss, val acc: [3.8172922134399414, 0.3048176169395447]

Ex:

```
lr_schedule = tf.keras.optimizers.schedules.ExponentialDecay(  
    initial_learning_rate=1e-4,
```

```
decay_steps=300,  
decay_rate=math.exp(math.log(0.1)/learning_rate_decay_every/steps_per_epoch))#0.9  
opt = tf.keras.optimizers.RMSprop(learning_rate=lr_schedule)
```

Vanilla DROPOUT_RATE = 0.5 → 0.5

EMBEDDING_DIM = 300

EPOCHS = 30

BATCH_SIZE = 300

SEQ_LENGTH = 26

VOCAB_SIZE = --

EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained

Train_size = all

Val_size = all

Network = default-baseline

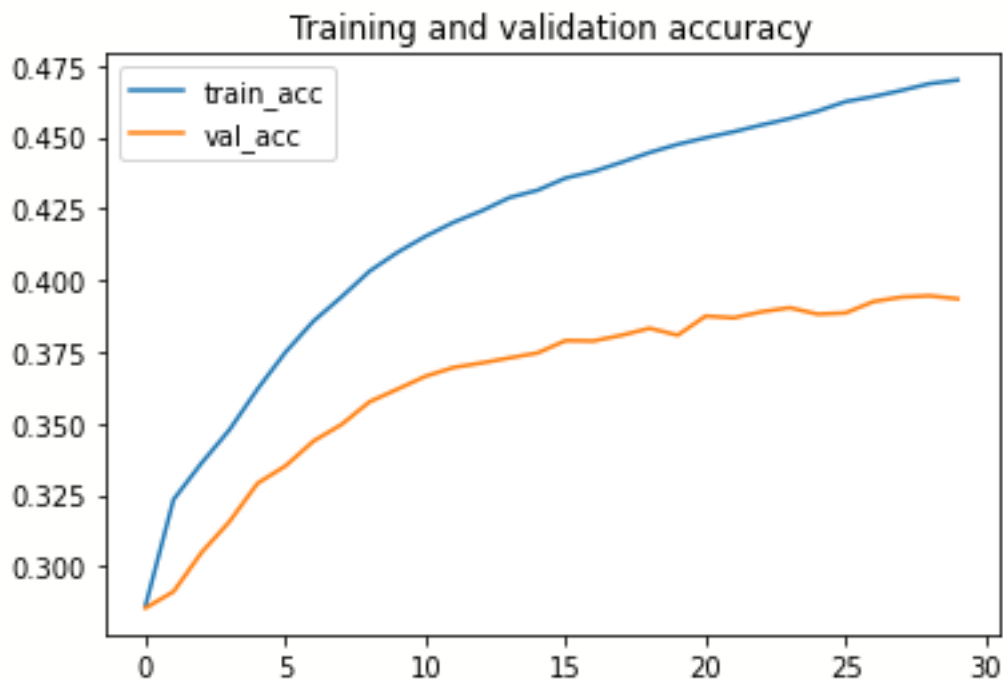
After VGG Normalization = true

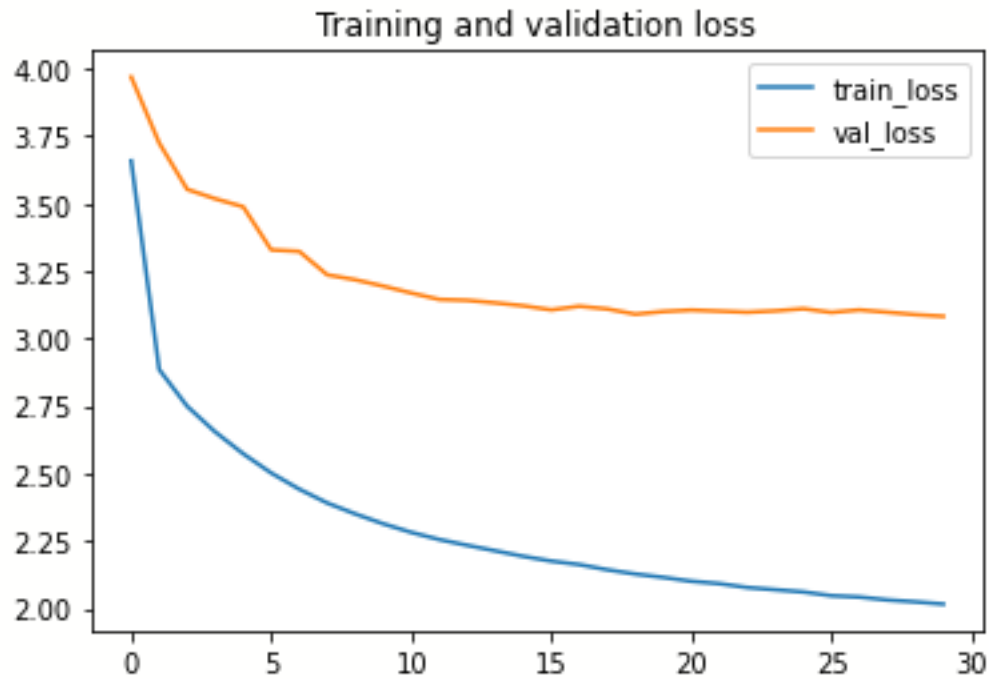
loss='categorical_crossentropy'

Filter in tokenizer= false

Text padding = pre

tex data feeded to network (questions)= exactly the same as paper





loss: 2.0158 - accuracy: 0.4699 - val_loss: 3.0818 - val_accuracy: 0.3935

val loss, val acc: [3.0818161964416504, 0.3935002386569977]

ex x1:

```
model.save(BASE_PATH_parssoft+'model_30ep_all.h5')
```

```
opt=tf.keras.optimizers.Adadelta(lr=1)
```

Vanilla DROPOUT_RATE = 0.5 → 0.5

EMBEDDING_DIM = 300

EPOCHS = 30

BATCH_SIZE = 300

SEQ_LENGTH = 26

VOCAB_SIZE = --

EMBEDDING_TYPE = Glove.6b.300d.en

Image_Features = VGG19 Pretrained

Train_size = all

Val_size = all

Network = default-baseline

After VGG Normalization = true

loss='categorical_crossentropy'

Filter in tokenizer= false

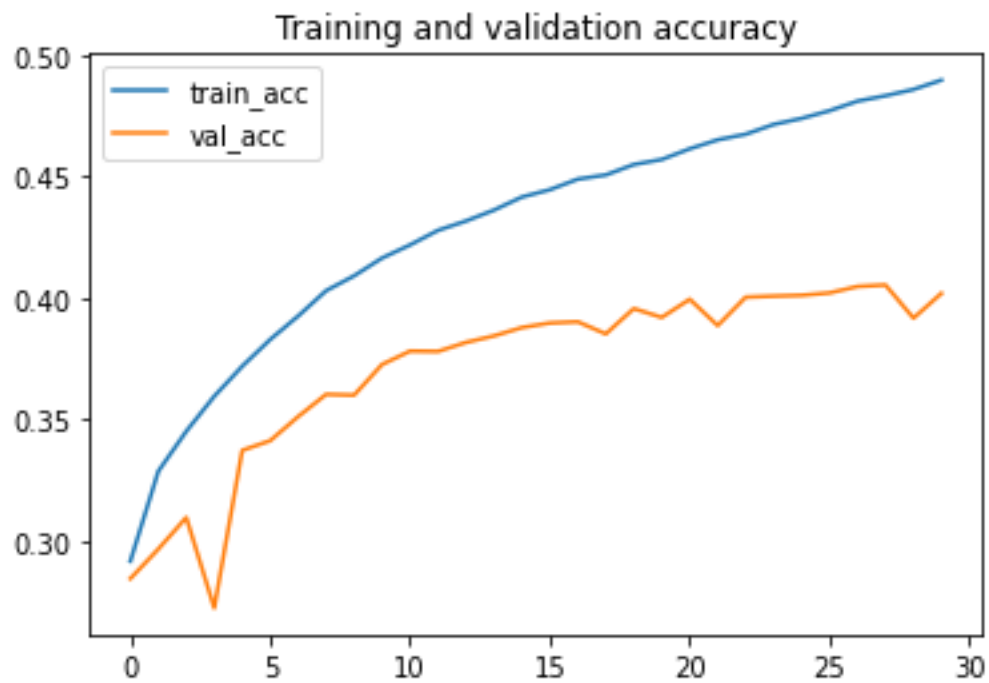
Text padding = pre

tex data feeded to network (questions)= exactly the same as paper

loading VQA annotations and questions into memory...

```
0:00:04.829438
creating index...
index created!
Loading and preparing results...
DONE (t=0.34s)
creating index...
index created!
computing accuracy
Finished Percent: [#####] 99% Done computing accuracy
```

accuracy: 0.4898 - val_loss: 2.8407 - val_accuracy: 0.4019





val loss, val acc: [2.840735673904419, 0.4019191563129425]

Overall Accuracy is: 46.57

Per Question Type Accuracy is the following:

what is the : 24.83
is the : 72.85
what : 26.64
is this an : 76.61
are there : 81.83
what is in the : 22.87
is it : 75.83
what is : 17.64
how many : 33.58
is this : 74.43
is there a : 89.25
is the person : 72.69
is this a : 74.88
what color is the : 35.67
how : 19.17
do : 75.02
how many people are in : 22.49
why : 13.42
are the : 74.69
none of the above : 46.59
are : 73.81
does this : 78.56
what is the person : 18.02
what are : 24.07
who is : 22.03
what brand : 33.27
has : 77.57
what kind of : 26.38
is the man : 77.23
are they : 75.11
what type of : 29.41

is : 76.54
is this person : 71.89
what is the man : 23.30
are these : 73.40
how many people are : 28.44
what is on the : 23.82
is there : 83.35
do you : 81.58
what color are the : 36.31
where is the : 17.15
what time : 12.29
does the : 77.95
are there any : 76.68
what room is : 43.54
what sport is : 45.65
is he : 77.77
is the woman : 73.68
which : 32.70
what does the : 17.14
what color : 26.84
what is the name : 4.34
where are the : 20.18
was : 75.95
what animal is : 20.04
what is the color of the : 44.48
what is the woman : 18.83
could : 89.82
what are the : 22.74
what is this : 22.44
can you : 75.77
what number is : 4.56
is that a : 76.03
what color is : 29.44
why is the : 14.09

Per Answer Type Accuracy is the following:
other : 27.31
yes/no : 78.50
number : 28.13

ground truth answers
Question: Is this a healthy meal?
Answer 1: no
Answer 2: no
Answer 3: no
Answer 4: no
Answer 5: no
Answer 6: no
Answer 7: no
Answer 8: no
Answer 9: no
Answer 10: no

generated answer (accuracy 0.00)
<Figure size 640x480 with 1 Axes>