In [1]:

```
import json
import os
from tqdm import tqdm_notebook
import gensim
from mermaid.embeddings import Embedding
from mermaid import preprocessing as pp
from mermaid.embeddings.word_index import WordIndex
from mermaid import plotting
from mermaid import models
import numpy as np
import mermaid
```

```
/home/guanxing/venv2/lib/python3.6/site-packages/h5py/__init__.py:36:
FutureWarning: Conversion of the second argument of issubdtype from `f
loat` to `np.floating` is deprecated. In future, it will be treated as
`np.float64 == np.dtype(float).type`.
  from ._conv import register_converters as _register_converters
Using TensorFlow backend.
/home/guanxing/venv2/lib/python3.6/importlib/_bootstrap.py:219: Runtim
eWarning: compiletime version 3.5 of module 'tensorflow.python.framewo
rk.fast_tensor_util' does not match runtime version 3.6
  return f(*args, **kwds)
```

In [2]:

```
def traverse_file_dir(file_path):
    a = os.walk(file_path)

    file_list = []
    file_list2 = []
    for item in a:
        file_list.append(item)
    for item in file_list[0][2]:
        file_list2.append(os.path.join(file_path,item))
    return file_list2

file_path = '/opt/share/dataset/enwiki/'
file_list = traverse_file_dir(file_path)
```

In [3]:

```
big_stream = pp.Stream()

for i in tqdm_notebook(range(6000),total=6000):
    with open (file_list[i]) as file:
        for single_line in file.readlines():
            a = json.loads(single_line)
            b={}
            b.setdefault('d',a['q'])
            b.setdefault('q',a['a'])
            b.setdefault('label',a['label'])
            big_stream.append(pp.Entry(**b))
```

100% 6000/6000 [00:39<00:00, 151.13it/s]

In [4]:

```
len(big_stream)
```

Out[4]:

```
3635176
```

In [4]:

```
new_stream = pp.Stream()
index = 0
for item in big_stream:
    if item['label']==1:
        new_stream.append(item)
        new_stream.append(big_stream[index+1])
    index+=1
```

In [5]:

```
len(new_stream)
```

Out[5]:

```
1199998
```

In [6]:

```
import os

os.environ['CUDA_VISIBLE_DEVICES'] = '0,1'

from mermaid import preprocessing as pp
from mermaid import sessions
from mermaid.factory import Factory

import json
```

In [7]:

```
train_stream, dev_stream, test_stream = new_stream.split()
```

In [8]:

```
class LoadedWordVecEmbedding(Embedding):
    def __init__(self, word_vector_file_path,dim):
        super().__init__()
        self._w2v = gensim.models.KeyedVectors.load_word2vec_format(word_vector_fil
        self._word_index = WordIndex({
            token:self._w2v.vocab[token].index + 2 for token in self._w2v.vocab
        })
        self._dim = dim


    def make_layer(self):
        embedding_matrix = np.zeros((self.vocab_size, self._dim))
        for key, index in self._word_index.items():
            embedding_matrix[index] = self._w2v[key]
        return keras.layers.Embedding(
                self.vocab_size,
                self._dim,
                weights=[embedding_matrix],
                trainable=self._trainable
        )

    @property
    def vocab_size(self):
        return len(self._word_index)+2

    @property
    def word_index(self):
        return self._word_index

    def __getitem__(self, item):
        return self._w2v[item]
```

In [9]:

```
glove_word_vector_embedding = LoadedWordVecEmbedding('/home/guanxing/glove_word_vec
glove_embedding = Factory.Embedding(glove_word_vector_embedding)
```

In [10]:

```
import keras
```

In [13]:

```
sess = sessions.DQTrainingSession()
sess.setup.train(train_stream)
sess.setup.dev(dev_stream)
sess.setup.test(test_stream)
sess.setup.preprocessor('split', 'split')
sess.setup.embedding(glove_embedding, name='glove_embedding')
sess.setup.model(models.ConvKNRM)
sess.setup.fit(epochs=100,batch_size=1024)
sess.compile()
sess.model.name='mermaid-convknrm-enwiki'
sess.fit()
sess.eval()

chart = plotting.ModelChart(sess._model.name)

chart.test_on(sess._model, *sess.XY(sess._test))
chart.add_result(sess._fit_history)
plotting.output_notebook()
chart.show()
```

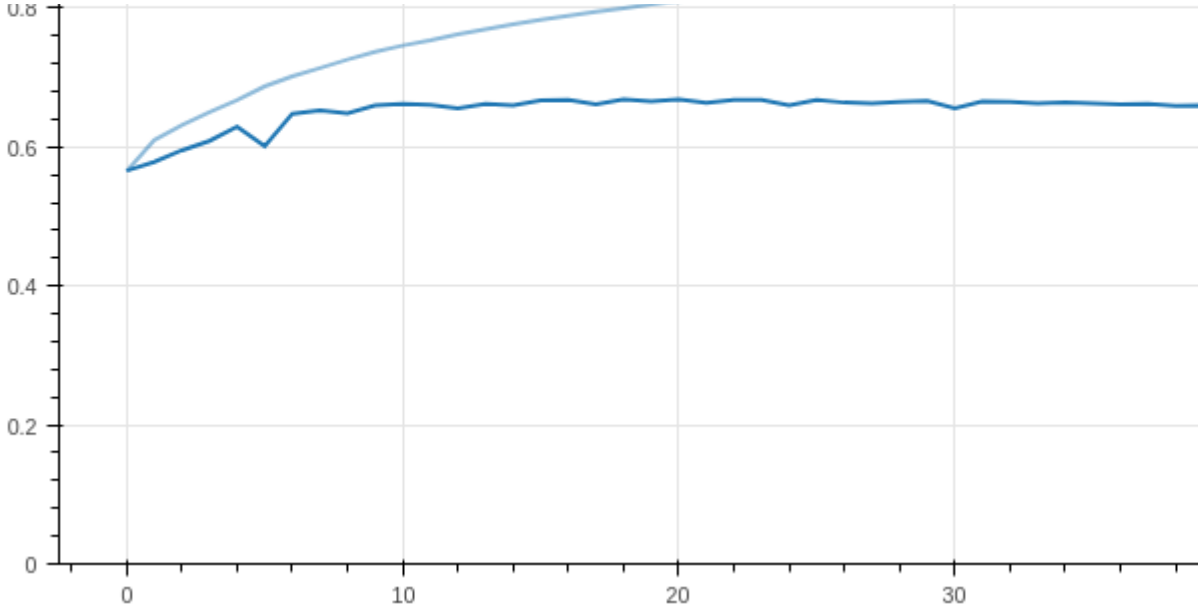
```
Train on 959998 samples, validate on 120000 samples
Epoch 1/50
959998/959998 [==============================] - 86s 90us/step - loss:
0.6495 - acc: 0.5653 - val_loss: 0.6814 - val_acc: 0.5659
Epoch 2/50
959998/959998 [==============================] - 84s 87us/step - loss:
0.6128 - acc: 0.6102 - val_loss: 0.6710 - val_acc: 0.5786
Epoch 3/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.5976 - acc: 0.6318 - val_loss: 0.6626 - val_acc: 0.5953
Epoch 4/50
959998/959998 [==============================] - 84s 87us/step - loss:
0.5832 - acc: 0.6503 - val_loss: 0.6525 - val_acc: 0.6088
Epoch 5/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.5685 - acc: 0.6677 - val_loss: 0.6358 - val_acc: 0.6291
Epoch 6/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.5511 - acc: 0.6873 - val_loss: 0.6486 - val_acc: 0.6011
Epoch 7/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.5362 - acc: 0.7019 - val_loss: 0.6207 - val_acc: 0.6479
Epoch 8/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.5232 - acc: 0.7136 - val_loss: 0.6168 - val_acc: 0.6529
Epoch 9/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.5099 - acc: 0.7259 - val_loss: 0.6200 - val_acc: 0.6487
Epoch 10/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4969 - acc: 0.7372 - val_loss: 0.6139 - val_acc: 0.6598
Epoch 11/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4859 - acc: 0.7460 - val_loss: 0.6171 - val_acc: 0.6620
Epoch 12/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4752 - acc: 0.7535 - val_loss: 0.6204 - val_acc: 0.6606
Epoch 13/50
959998/959998 [==============================] - 84s 88us/step - loss:
```

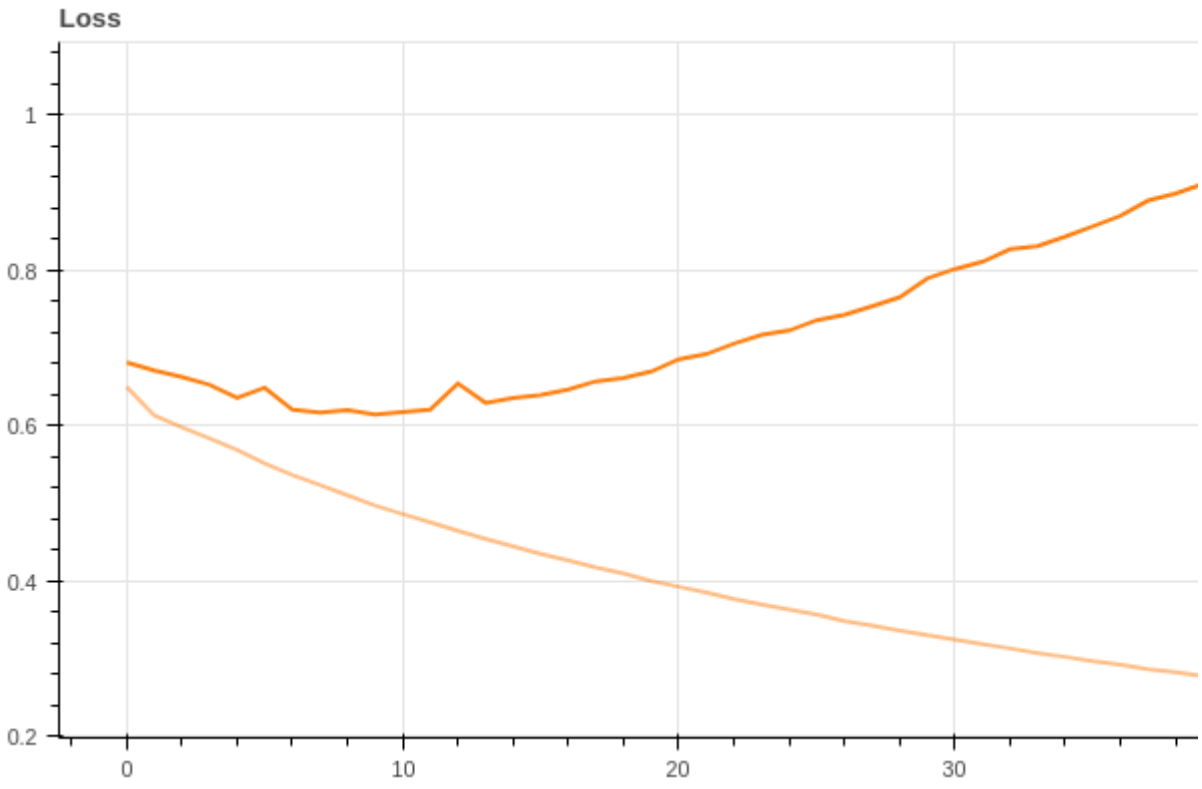
```
0.4642 - acc: 0.7621 - val_loss: 0.6542 - val_acc: 0.6555
Epoch 14/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4540 - acc: 0.7694 - val_loss: 0.6292 - val_acc: 0.6620
Epoch 15/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4441 - acc: 0.7766 - val_loss: 0.6353 - val_acc: 0.6602
Epoch 16/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4347 - acc: 0.7832 - val_loss: 0.6390 - val_acc: 0.6673
Epoch 17/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4259 - acc: 0.7886 - val_loss: 0.6462 - val_acc: 0.6680
Epoch 18/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4169 - acc: 0.7948 - val_loss: 0.6569 - val_acc: 0.6614
Epoch 19/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.4091 - acc: 0.7998 - val_loss: 0.6615 - val_acc: 0.6687
Epoch 20/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3997 - acc: 0.8056 - val_loss: 0.6693 - val_acc: 0.6661
Epoch 21/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3922 - acc: 0.8106 - val_loss: 0.6853 - val_acc: 0.6686
Epoch 22/50
959998/959998 [==============================] - 84s 87us/step - loss:
0.3848 - acc: 0.8150 - val_loss: 0.6919 - val_acc: 0.6634
Epoch 23/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3763 - acc: 0.8198 - val_loss: 0.7055 - val_acc: 0.6681
Epoch 24/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3693 - acc: 0.8240 - val_loss: 0.7167 - val_acc: 0.6676
Epoch 25/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3626 - acc: 0.8281 - val_loss: 0.7224 - val_acc: 0.6597
Epoch 26/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3561 - acc: 0.8317 - val_loss: 0.7357 - val_acc: 0.6677
Epoch 27/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3481 - acc: 0.8367 - val_loss: 0.7428 - val_acc: 0.6639
Epoch 28/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3422 - acc: 0.8397 - val_loss: 0.7535 - val_acc: 0.6629
Epoch 29/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3359 - acc: 0.8433 - val_loss: 0.7651 - val_acc: 0.6648
Epoch 30/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3297 - acc: 0.8467 - val_loss: 0.7895 - val_acc: 0.6663
Epoch 31/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3243 - acc: 0.8498 - val_loss: 0.8016 - val_acc: 0.6554
Epoch 32/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3181 - acc: 0.8530 - val_loss: 0.8109 - val_acc: 0.6656
Epoch 33/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3127 - acc: 0.8558 - val_loss: 0.8272 - val_acc: 0.6650
```

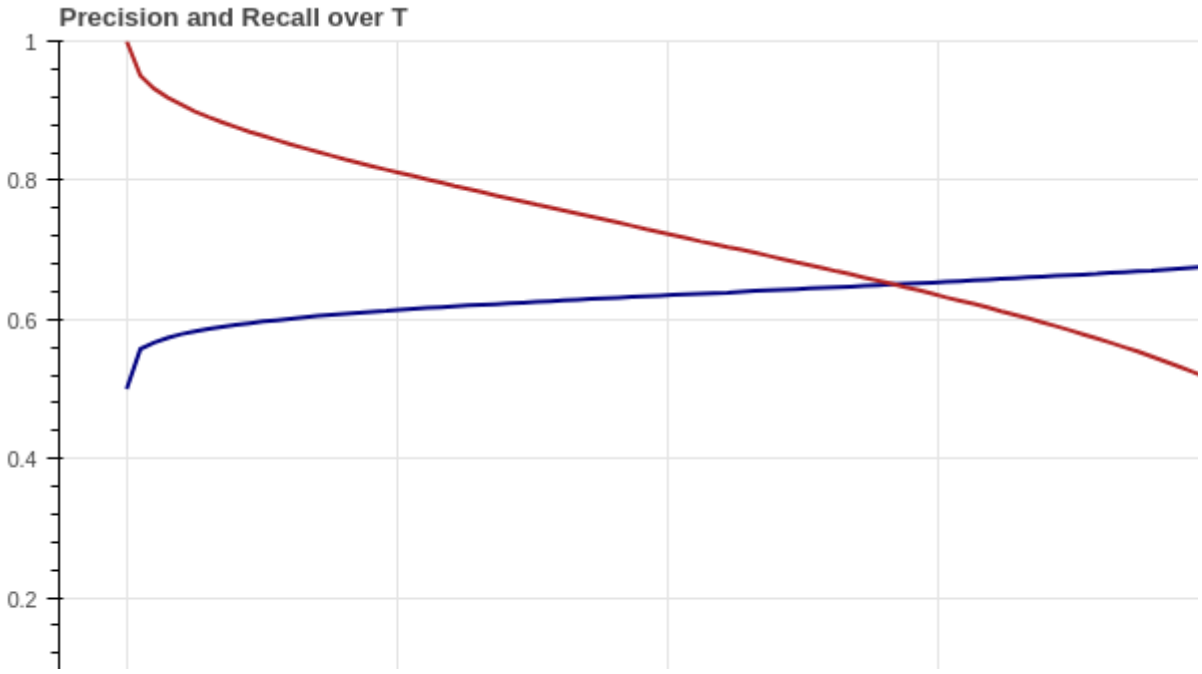
```
Epoch 34/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3069 - acc: 0.8592 - val_loss: 0.8310 - val_acc: 0.6628
Epoch 35/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.3017 - acc: 0.8615 - val_loss: 0.8433 - val_acc: 0.6644
Epoch 36/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2961 - acc: 0.8648 - val_loss: 0.8566 - val_acc: 0.6627
Epoch 37/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2915 - acc: 0.8673 - val_loss: 0.8700 - val_acc: 0.6617
Epoch 38/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2858 - acc: 0.8698 - val_loss: 0.8899 - val_acc: 0.6619
Epoch 39/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2823 - acc: 0.8720 - val_loss: 0.8988 - val_acc: 0.6593
Epoch 40/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2766 - acc: 0.8746 - val_loss: 0.9111 - val_acc: 0.6602
Epoch 41/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2730 - acc: 0.8766 - val_loss: 0.9316 - val_acc: 0.6582
Epoch 42/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2683 - acc: 0.8789 - val_loss: 0.9407 - val_acc: 0.6609
Epoch 43/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2639 - acc: 0.8814 - val_loss: 0.9537 - val_acc: 0.6618
Epoch 44/50
959998/959998 [==============================] - 84s 88us/step - loss:
0.2600 - acc: 0.8833 - val_loss: 0.9662 - val_acc: 0.6604
Epoch 45/50
959998/959998 [==============================] - 83s 87us/step - loss:
0.2561 - acc: 0.8851 - val_loss: 0.9882 - val_acc: 0.6618
Epoch 46/50
959998/959998 [==============================] - 83s 87us/step - loss:
0.2522 - acc: 0.8870 - val_loss: 0.9923 - val_acc: 0.6591
Epoch 47/50
959998/959998 [==============================] - 83s 87us/step - loss:
0.2486 - acc: 0.8885 - val_loss: 1.0295 - val_acc: 0.6533
Epoch 48/50
959998/959998 [==============================] - 83s 87us/step - loss:
0.2450 - acc: 0.8904 - val_loss: 1.0270 - val_acc: 0.6603
Epoch 49/50
959998/959998 [==============================] - 83s 87us/step - loss:
0.2412 - acc: 0.8925 - val_loss: 1.0393 - val_acc: 0.6600
Epoch 50/50
959998/959998 [==============================] - 83s 87us/step - loss:
0.2381 - acc: 0.8936 - val_loss: 1.0533 - val_acc: 0.6599
120000/120000 [==============================] - 10s 81us/step
Loss: 1.0701910430908204, Accuracy: 0.6520833333333333
```

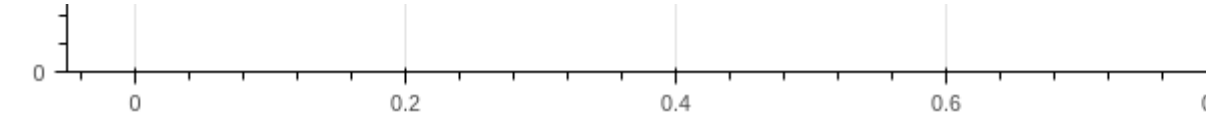
(https://bokeh.pydata.org) BokehJS 0.12.13 successfully loaded.

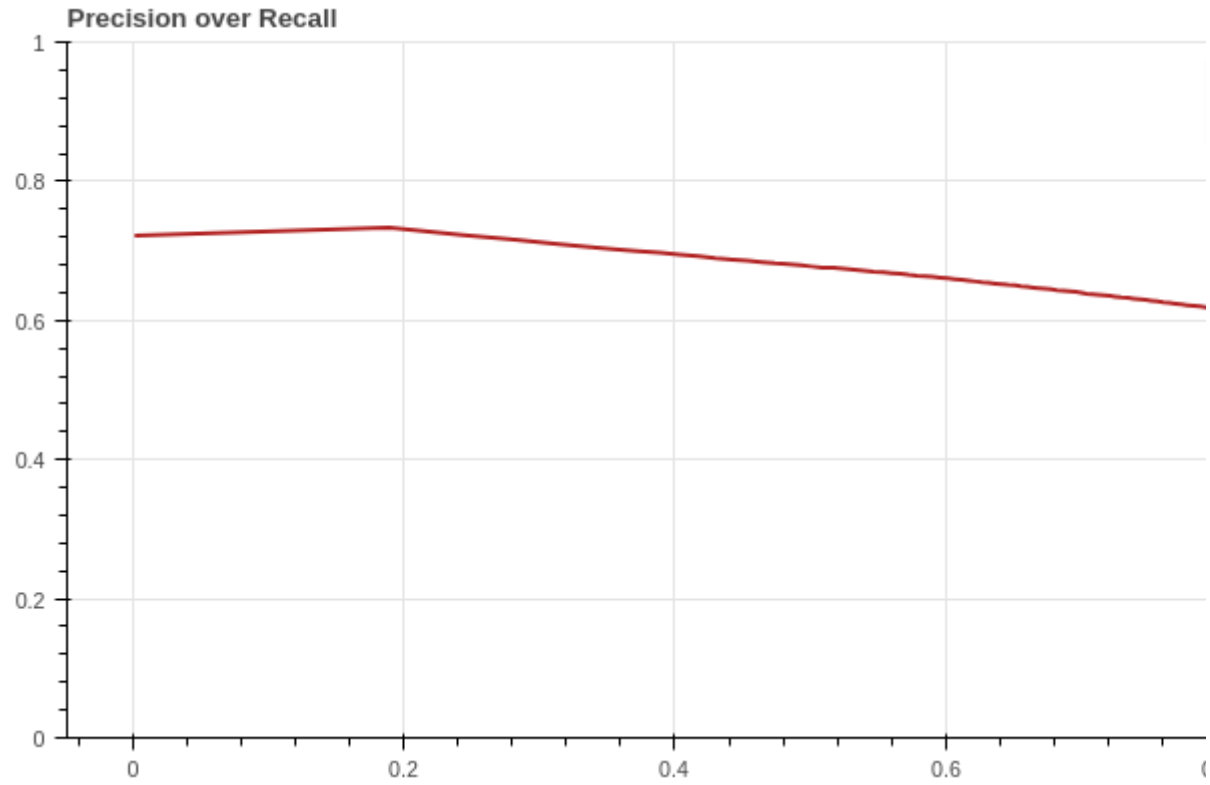


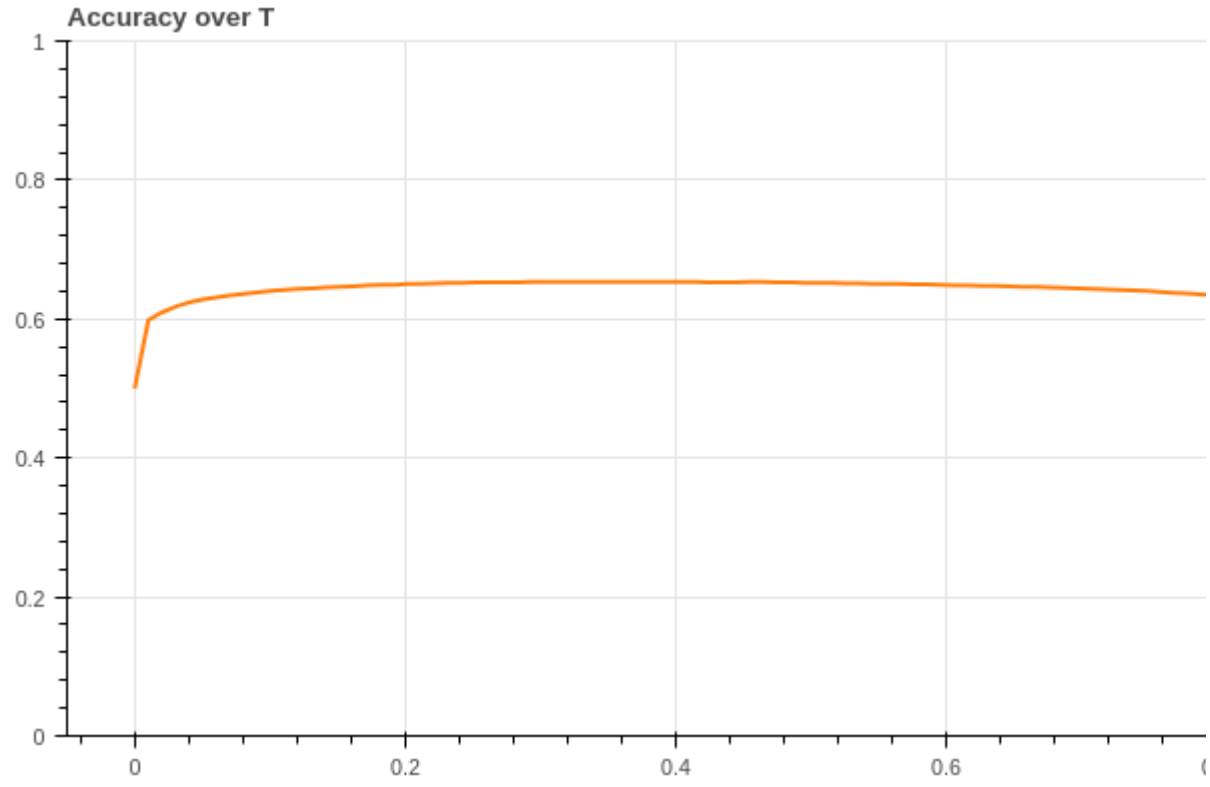

0.8
0.6
0.4
0.2
0
0
10
20
30


Loss
1
0.8
0.6
0.4
0.2
0
10
20
30


Precision and Recall over T
1
0.8
0.6
0.4
0.2


0
0
0.2
0.4
0.6


Precision over Recall
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6


Accuracy over T
1
0.8
0.6
0.4
0.2
0
0
0.2
0.4
0.6

In [ ]: