In [6]:

```
pip install geopandas
```

```
Collecting geopandas
  Using cached https://files.pythonhosted.org/packages/83/c5/3cf9cdc39a6f2552922f79915f36b45a95b71fd343cfc51
2.py3-none-any.whl (https://files.pythonhosted.org/packages/83/c5/3cf9cdc39a6f2552922f79915f36b45a95b71fd343cfc
y2.py3-none-any.whl)
Collecting fiona
  Downloading https://files.pythonhosted.org/packages/ec/20/4e63bc5c6e62df889297b382c3ccd4a7a488b00946aaaf
36-cp36m-manylinux1_x86_64.whl (https://files.pythonhosted.org/packages/ec/20/4e63bc5c6e62df889297b382c3ccd4a
na-1.8.13.post1-cp36-cp36m-manylinux1_x86_64.whl) (14.7MB)
     |████████████████████████████████| 14.7MB 6.3MB/s eta 0:00:01
Requirement already satisfied: pyproj>=2.2.0 in /home/lechen/.local/lib/python3.6/site-packages
Requirement already satisfied: pandas>=0.23.0 in /home/lechen/anaconda3/lib/python3.6/site-packa
Collecting shapely
  Downloading https://files.pythonhosted.org/packages/20/fa/c96d3461fda99ed8e82ff0b219ac2c8384694b4e640a611a
36m-manylinux1_x86_64.whl (https://files.pythonhosted.org/packages/20/fa/c96d3461fda99ed8e82ff0b219ac2c8384694b
7.0-cp36-cp36m-manylinux1_x86_64.whl) (1.8MB)
     |████████████████████████████████| 1.8MB 13.7MB/s eta 0:00:01
Collecting cligj>=0.5
  Using cached https://files.pythonhosted.org/packages/e4/be/30a58b4b0733850280d01f8bd132591b4668ed5c70467
ne-any.whl (https://files.pythonhosted.org/packages/e4/be/30a58b4b0733850280d01f8bd132591b4668ed5c7046761098c
whl)
Requirement already satisfied: six>=1.7 in /home/lechen/anaconda3/lib/python3.6/site-packages (f
Requirement already satisfied: click<8,>=4.0 in /home/lechen/anaconda3/lib/python3.6/site-packag
0)
Collecting munch
```

In [1]:

```
import numpy as np
import pandas as pd
import geopandas as gpd
from shapely.geometry import Point
import os
import tensorflow as tf
from tqdm import tqdm
from sklearn.utils import shuffle
from sklearn.metrics import mean_squared_log_error
from math import sqrt
from datetime import datetime
from datetime import timedelta
from tensorflow.keras import layers
from keras.layers import Dense
from keras.layers import LSTM
from keras.layers import Bidirectional
from keras.layers import Masking
from keras.layers import BatchNormalization
from keras.layers import ZeroPadding2D
from keras.layers import Activation
from tensorflow.keras import Input
from tensorflow.keras.models import Model
from tensorflow.keras.callbacks import ModelCheckpoint, ReduceLROnPlate
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_error
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()


```

```
/home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflow/python/framework/dtype
ng (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it wil
(1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
/home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflow/python/framework/dtype
ng (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it wil
(1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
/home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflow/python/framework/dtype
ng (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it wil
(1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
```

```
/home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflow/python/framework/dtype
ng (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it wil
(1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
/home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflow/python/framework/dtype
ng (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it wil
(1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
/home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflow/python/framework/dtype
ng (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it wil
(1,)) / '(1,)type'.
```

In [2]:

```python
train_df = gpd.read_file("enriched_covid_19_week_2.csv")
train_df["Country_Region"] = [country_name.replace("'","") for country_
train_df["restrictions"] = train_df["restrictions"].astype("int")
train_df["quarantine"] = train_df["quarantine"].astype("int")
train_df["schools"] = train_df["schools"].astype("int")
train_df["total_pop"] = train_df["total_pop"].astype("float")
train_df["density"] = train_df["density"].astype("float")
train_df["hospibed"] = train_df["hospibed"].astype("float")
train_df["lung"] = train_df["lung"].astype("float")
train_df["total_pop"] = train_df["total_pop"]/max(train_df["total_pop"]
train_df["density"] = train_df["density"]/max(train_df["density"])
train_df["hospibed"] = train_df["hospibed"]/max(train_df["hospibed"])
train_df["lung"] = train_df["lung"]/max(train_df["lung"])
train_df.head()
```

	Id	Province_State	Country_Region	Date	ConfirmedCases	Fatalities	age_0-4	
0	1		Afghanistan	2020-01-22	0.0	0.0	0.1457166900587929	(
1	2		Afghanistan	2020-01-23	0.0	0.0	0.1457166900587929	(
2	3		Afghanistan	2020-01-24	0.0	0.0	0.1457166900587929	(
3	4		Afghanistan	2020-01-25	0.0	0.0	0.1457166900587929	(
4	5		Afghanistan	2020-01-26	0.0	0.0	0.1457166900587929	(

5 rows × 39 columns

In [3]:

```
train_df.columns
```

```
Index(['Id', 'Province_State', 'Country_Region', 'Date', 'ConfirmedCases',
       'Fatalities', 'age_0-4', 'age_5-9', 'age_10-14', 'age_15-19',
       'age_20-24', 'age_25-29', 'age_30-34', 'age_35-39', 'age_40-44',
       'age_45-49', 'age_50-54', 'age_55-59', 'age_60-64', 'age_65-69',
       'age_70-74', 'age_75-79', 'age_80-84', 'age_85-89', 'age_90-94',
       'age_95-99', 'age_100+', 'total_pop', 'smokers_perc', 'density',
       'urbanpop', 'hospibed', 'lung', 'femalelung', 'malelung',
       'restrictions', 'quarantine', 'schools', 'geometry'],
      dtype='object')
```

In [ ]:

In [4]:

```
trend_df = pd.DataFrame(columns={"infection_trend","fatality_trend","qu
train_df = train_df.query("Date>'2020-01-22'and Date<='2020-03-18'")
days_in_sequence = 14
trend_list = []
with tqdm(total=len(list(train_df.Country_Region.unique()))) as pbar:
    for country in train_df.Country_Region.unique():
        for province in train_df.query(f"Country_Region=='{country}'").
            province_df = train_df.query(f"Country_Region=='{country}'
            for i in range(0,len(province_df),int(days_in_sequence/2)):
                if i+days_in_sequence<=len(province_df):
                    #prepare all the temporal inputs
                    infection_trend = [float(x) for x in province_df[i:
                    fatality_trend = [float(x) for x in province_df[i:i
                    restriction_trend = [float(x) for x in province_df[
                    quarantine_trend = [float(x) for x in province_df[i
                    school_trend = [float(x) for x in province_df[i:i+d
                    #preparing all the demographic inputs
                    total_population = float(province_df.iloc[i].total_
                    density = float(province_df.iloc[i].density)
                    hospibed = float(province_df.iloc[i].hospibed)
                    lung = float(province_df.iloc[i].lung)
                    expected_cases = float(province_df.iloc[i+days_in_s
                    expected_fatalities = float(province_df.iloc[i+days

                    trend_list.append({"infection_trend":infection_trer
                                     "fatality_trend":fatality_trend,
                                     "restriction_trend":restriction_tr
                                     "quarantine_trend":quarantine_trer
                                     "school_trend":school_trend,
                                     "demographic_inputs":[total_popula
                                     "expected_cases":expected_cases,
                                     "expected_fatalities":expected_fat
        pbar.update(1)
trend_df = pd.DataFrame(trend_list)
```

```
100%|██████████| 294/294 [00:16<00:00, 17.59it/s]
```

In [ ]:

```

```

In [5]:

```python
trend_df["temporal_inputs"] = [np.asarray([trends["infection_trend"],tr
trend_df = shuffle(trend_df)
i=0
y=0
temp_df = pd.DataFrame()
for idx,row in trend_df.iterrows():
    if sum(row.infection_trend)>0:
        temp_df = temp_df.append(row)
    else:
        if i<25:
            temp_df = temp_df.append(row)
            i+=1
trend_df = temp_df
```

In [6]:

```python
# Splitting my dataset with 80% for training and 10% for validation
sequence_length = 13
training_percentage = 0.8
training_item_count = int(len(trend_df)*training_percentage)
validation_item_count = len(trend_df)-int(len(trend_df)*training_percen
training_df = trend_df[:training_item_count]
validation_df = trend_df[training_item_count:]
```

In [7]:

```python
X_temporal_train = np.asarray(np.reshape(np.asarray([np.asarray(x) for
X_demographic_train = np.asarray([np.asarray(x) for x in training_df["d
Y_cases_train = np.asarray([np.asarray(x) for x in training_df["expecte
Y_fatalities_train = np.asarray([np.asarray(x) for x in training_df["ex
```

In [8]:

```python
X_temporal_test = np.asarray(np.reshape(np.asarray([np.asarray(x) for x
X_demographic_test = np.asarray([np.asarray(x) for x in validation_df["
Y_cases_test = np.asarray([np.asarray(x) for x in validation_df["expect
Y_fatalities_test = np.asarray([np.asarray(x) for x in validation_df["e
```

In [9]:

```
training_df.head()
```

	demographic_inputs	expected_cases	expected_fatalities	fatality_trend	infection_tren
1417	[0.006070468756114634, 0.003796939666628697, 0...	0.0	0.0	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0. 0.0, 0.0, ...
1575	[0.2322267255206195, 0.003607092683297262, 0.1...	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, ...
297	[0.026222145205808296, 0.003607092683297262, 0...	4.0	0.0	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[3.0, 3.0, 3.0, 3.0, 3.0, 3.0, 3. 3.0, 3.0, ...
73	[0.017716570420520272, 0.003607092683297262, 0...	0.0	0.0	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0. 0.0, 0.0, ...
899	[0.0005464781546782123, 0.00015187758666514788...	0.0	0.0	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, ...	[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0. 0.0, 0.0, ...

In [10]:

```
sequence_length = 13
training_percentage = 0.9
#temporal input branch
temporal_input_layer = Input(shape=(5,sequence_length))
main_rnn_layer = layers.LSTM(128, return_sequences=True, recurrent_drop
#demographic input branch
demographic_input_layer = Input(shape=(4))
demographic_dense = layers.Dense(16)(demographic_input_layer)
demographic_dropout = layers.Dropout(0.2)(demographic_dense)
#cases output branch
blstm1_c = layers.Bidirectional(layers.LSTM(50,return_sequences=True,
                                         kernel_initializer='he_normal'
                                         dropout=0.02,recurrent_dropout
dense1_c = layers.Dense(100, activation = 'relu')(blstm1_c)
blstm2_c = layers.Bidirectional(layers.LSTM(50,return_sequences=True,
                                         kernel_initializer='he_normal'
                                         dropout=0.02,recurrent_dropout
# blstm3_c = layers.Bidirectional(layers.LSTM(50,return_sequences=True)
blstm3_c = layers.Bidirectional(layers.LSTM(50,return_sequences=True,
                                            kernel_initializer='he_norm
                                            dropout=0.02,recurrent_drop
blstm4_c = layers.Bidirectional(layers.LSTM(30,return_sequences=True,
                                            kernel_initializer='he_norm
                                            dropout=0.02,recurrent_drop
dense2_c = layers.Dense(50, activation = 'relu')(blstm4_c)
lstm1_c = layers.LSTM(50, kernel_initializer='he_normal',
                      dropout=0.02,recurrent_dropout = 0.02)(dense2_c)
merge_c = layers.Concatenate(axis=-1)([lstm1_c,demographic_dropout])
dense3_c = layers.Dense(100)(merge_c)
dropout_c = layers.Dropout(0.3)(dense3_c)
cases = layers.Dense(1, activation = 'relu',name="cases")(dropout_c)
#fatality output branch
blstm1_f = layers.Bidirectional(layers.LSTM(50,return_sequences=True,
                                         kernel_initializer='he_normal'
                                         dropout=0.02,recurrent_dropout
dense1_f = layers.Dense(100, activation = 'relu')(blstm1_f)
blstm2_f = layers.Bidirectional(layers.LSTM(50,return_sequences=True,
                                         kernel_initializer='he_normal'
                                         dropout=0.02,recurrent_dropout
blstm3_f = layers.Bidirectional(layers.LSTM(50,return_sequences=True,
                                            kernel_initializer='he_norm
                                            dropout=0.02,recurrent_drop
blstm4_f = layers.Bidirectional(layers.LSTM(30,return_sequences=True,
                                            kernel_initializer='he_norm
                                            dropout=0.02,recurrent_drop
dense2_f = layers.Dense(50, activation = 'relu')(blstm4_f)
lstm1_f = layers.LSTM(50, kernel_initializer='he_normal',
                      dropout=0.02,recurrent_dropout = 0.02)(dense2_f)
merge_f = layers.Concatenate(axis=-1)([lstm1_f,demographic_dropout])
dense3_f = layers.Dense(100)(merge_f)
dropout_f = layers.Dropout(0.3)(dense3_f)
cases = layers.Dense(1, activation = 'relu',name="cases")(dropout_f)
fatalities = layers.Dense(1, activation = 'relu', name="fatalities")(dr
model = Model([temporal_input_layer,demographic_input_layer], [cases,fa

model.summary()
```

```
WARNING:tensorflow:From /home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflo
calling VarianceScaling.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated
version.
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
WARNING:tensorflow:From /home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflo
lling Orthogonal.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and wil
n.
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
WARNING:tensorflow:From /home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflo
lling Zeros.__init__ (from tensorflow.python.ops.init_ops) with dtype is deprecated and will be
Instructions for updating:
Call initializer instance with the dtype argument instead of passing it to the constructor
Model: "model"
_______________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
===============================================================================================
input_1 (InputLayer)            [(None, 5, 13)]      0
_______________________________________________________________________________________________
lstm (LSTM)                     (None, 5, 128)       72704       input_1[0][0]
_______________________________________________________________________________________________
bidirectional_4 (Bidirectional) (None, 5, 100)       71600       lstm[0][0]
_______________________________________________________________________________________________
dense_4 (Dense)                 (None, 5, 100)       10100       bidirectional_4[0][0]
_______________________________________________________________________________________________
bidirectional_5 (Bidirectional) (None, 5, 100)       60400       dense_4[0][0]
_______________________________________________________________________________________________
bidirectional_6 (Bidirectional) (None, 5, 100)       60400       bidirectional_5[0][0]
_______________________________________________________________________________________________
bidirectional_7 (Bidirectional) (None, 5, 60)        31440       bidirectional_6[0][0]
```

```
________________________________________________________________________________________________
input_2 (InputLayer)            [(None, 4)]          0
________________________________________________________________________________________________
dense_5 (Dense)                 (None, 5, 50)        3050        bidirectional_7[0][0]
________________________________________________________________________________________________
dense (Dense)                   (None, 16)           80          input_2[0][0]
________________________________________________________________________________________________
lstm_10 (LSTM)                  (None, 50)           20200       dense_5[0][0]
________________________________________________________________________________________________
dropout (Dropout)               (None, 16)           0           dense[0][0]
________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 66)           0           lstm_10[0][0]
                                                                 dropout[0][0]
________________________________________________________________________________________________
dense_6 (Dense)                 (None, 100)          6700        concatenate_1[0][0]
________________________________________________________________________________________________
dropout_2 (Dropout)             (None, 100)          0           dense_6[0][0]
________________________________________________________________________________________________
cases (Dense)                   (None, 1)            101         dropout_2[0][0]
________________________________________________________________________________________________
fatalities (Dense)              (None, 1)            101         dropout_2[0][0]
================================================================================================
Total params: 336,876
Trainable params: 336,876
Non-trainable params: 0
________________________________________________________________________________________________
```

In [11]:

```python
################################### set GPU #############################
import os
import tensorflow as tf
import keras.backend.tensorflow_backend as KTF
os.environ["CUDA_VISIBLE_DEVICES"]="2"
config = tf.ConfigProto()
# config.gpu_options.per_process_gpu_memory_fraction = 0.6
session = tf.Session(config=config)
KTF.set_session(session )
```

In [13]:

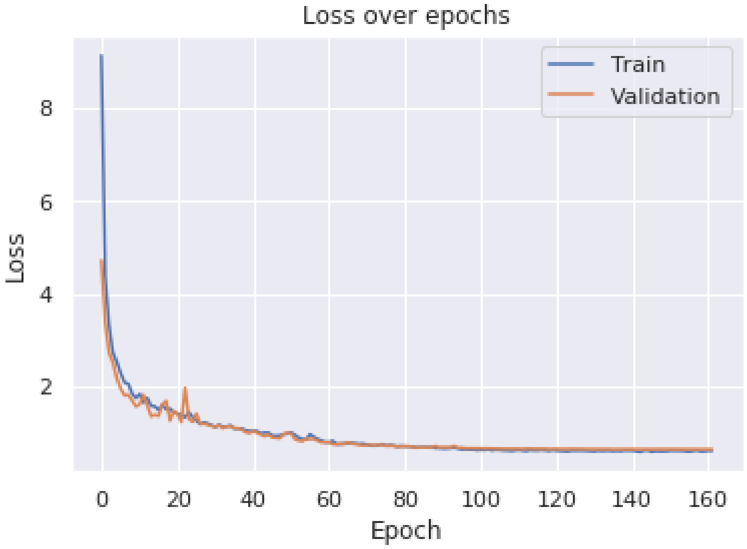
```
callbacks = [ReduceLROnPlateau(monitor='val_loss', patience=4, verbose=
             EarlyStopping(monitor='val_loss', patience=20),
             ModelCheckpoint(filepath='best_model.h5', monitor='val_los
model.compile(loss=[tf.keras.losses.MeanSquaredLogarithmicError(),tf.ke
history = model.fit([X_temporal_train,X_demographic_train], [Y_cases_tr
          epochs = 50,
          batch_size = 16,
          validation_data=([X_temporal_test,X_demographic_test],  [Y_ca
          callbacks=callbacks)
```

```
Train on 680 samples, validate on 171 samples
WARNING:tensorflow:From /home/lechen/anaconda3/envs/lechen/lib/python3.7/site-packages/tensorflo
0: add_dispatch_support.<locals>.wrapper (from tensorflow.python.ops.array_ops) is deprecated ar
version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
Epoch 1/50
680/680 [==============================] - 40s 59ms/sample - loss: 8.9888 - cases_loss: 7.6035 -
_loss: 7.6444 - val_cases_loss: 5.8887 - val_fatalities_loss: 2.0179
Epoch 2/50
680/680 [==============================] - 17s 24ms/sample - loss: 4.2394 - cases_loss: 3.1099 -
_loss: 5.1371 - val_cases_loss: 3.8278 - val_fatalities_loss: 1.3103
Epoch 3/50
680/680 [==============================] - 19s 28ms/sample - loss: 3.3431 - cases_loss: 2.3822 -
_loss: 4.2824 - val_cases_loss: 2.9693 - val_fatalities_loss: 1.2396
Epoch 4/50
680/680 [==============================] - 13s 20ms/sample - loss: 2.7325 - cases_loss: 1.9466 -
_loss: 3.7160 - val_cases_loss: 2.7297 - val_fatalities_loss: 0.9624
Epoch 5/50
680/680 [==============================] - 9s 14ms/sample - loss: 2.3302 - cases_loss: 1.7151 -
loss: 3.4413 - val_cases_loss: 2.5015 - val_fatalities_loss: 1.1100
Epoch 6/50
680/680 [==============================] - 8s 12ms/sample - loss: 2.0503 - cases_loss: 1.4966 -
loss: 3.1443 - val_cases_loss: 2.1828 - val_fatalities_loss: 0.9198
```

In [76]:

```
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('Loss over epochs')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train', 'Validation'], loc='best')
plt.show()

```



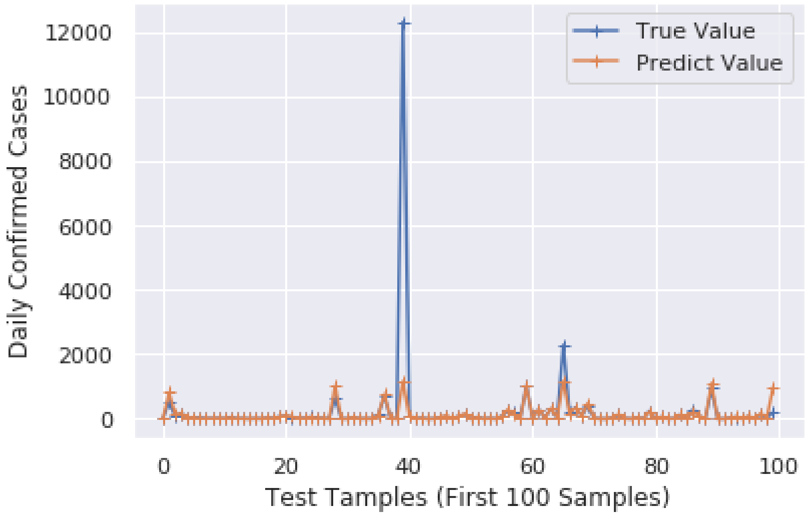
In [44]:

```
predictions = model.predict([X_temporal_test,X_demographic_test])
Y_cases_prediction = predictions[0]
Y_fatalities_prediction = predictions[1]
```

In [93]:

```
plt.plot([x for x in range(len(Y_cases_test[0:100])) ],Y_cases_test[0:1
plt.xlabel('Test Tamples (First 100 Samples)')
plt.ylabel('Daily Confirmed Cases')
plt.legend(['True Value', 'Predict Value'])
rmse = sqrt(mean_squared_error(Y_cases_test, Y_cases_prediction))
print('Test RMSE: %.3f' % rmse)
```

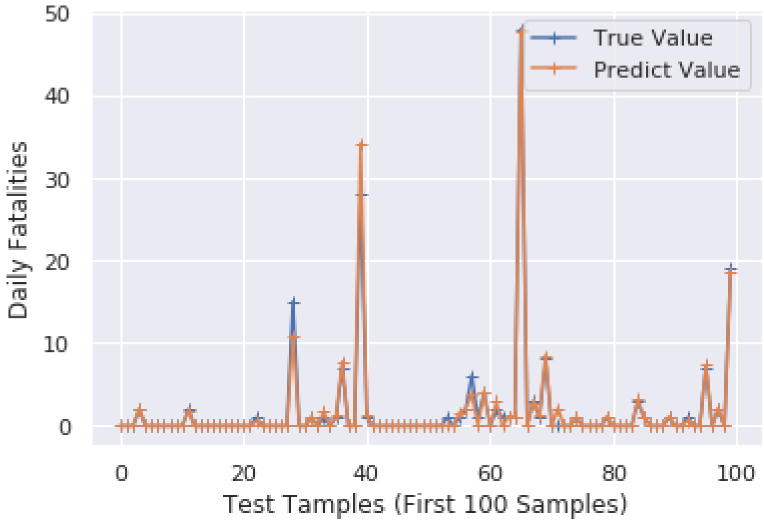
```
Test RMSE: 1127.061
```



In [94]:

```
plt.plot([x for x in range(len(Y_fatalities_test[0:100])) ],Y_fatalitie
plt.xlabel('Test Tamples (First 100 Samples)')
plt.ylabel('Daily Fatalities')
plt.legend(['True Value', 'Predict Value'])
rmse = sqrt(mean_squared_error(Y_fatalities_test[0:100], Y_fatalities_p
print('Test RMSE: %.3f' % rmse)
```

```
Test RMSE: 0.845
```


True Value
Predict Value
Daily Fatalities
Test Tamples (First 100 Samples)