In [27]:
```
from sklearn import metrics
from matplotlib.pylab import rcParams
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [28]:
```
#Data Collection and Reading
data = pd.read_csv("OnlineNewsPopularity.csv")
data.head()
```

Out[28]:

	url	timedelta	n_tokens_title	n_tokens_content	n_unique_t
0	http://mashable.com/2013/01/07/amazon-instant-...	731.0	12.0	219.0	0.6
1	http://mashable.com/2013/01/07/ap-samsung-spon...	731.0	9.0	255.0	0.6
2	http://mashable.com/2013/01/07/apple-40-billio...	731.0	9.0	211.0	0.5
3	http://mashable.com/2013/01/07/astronaut-notre...	731.0	9.0	531.0	0.5
4	http://mashable.com/2013/01/07/att-u-verse-apps/	731.0	13.0	1072.0	0.4

5 rows × 61 columns

In [29]:
```
data.shape
```

Out[29]:
```
(39644, 61)
```

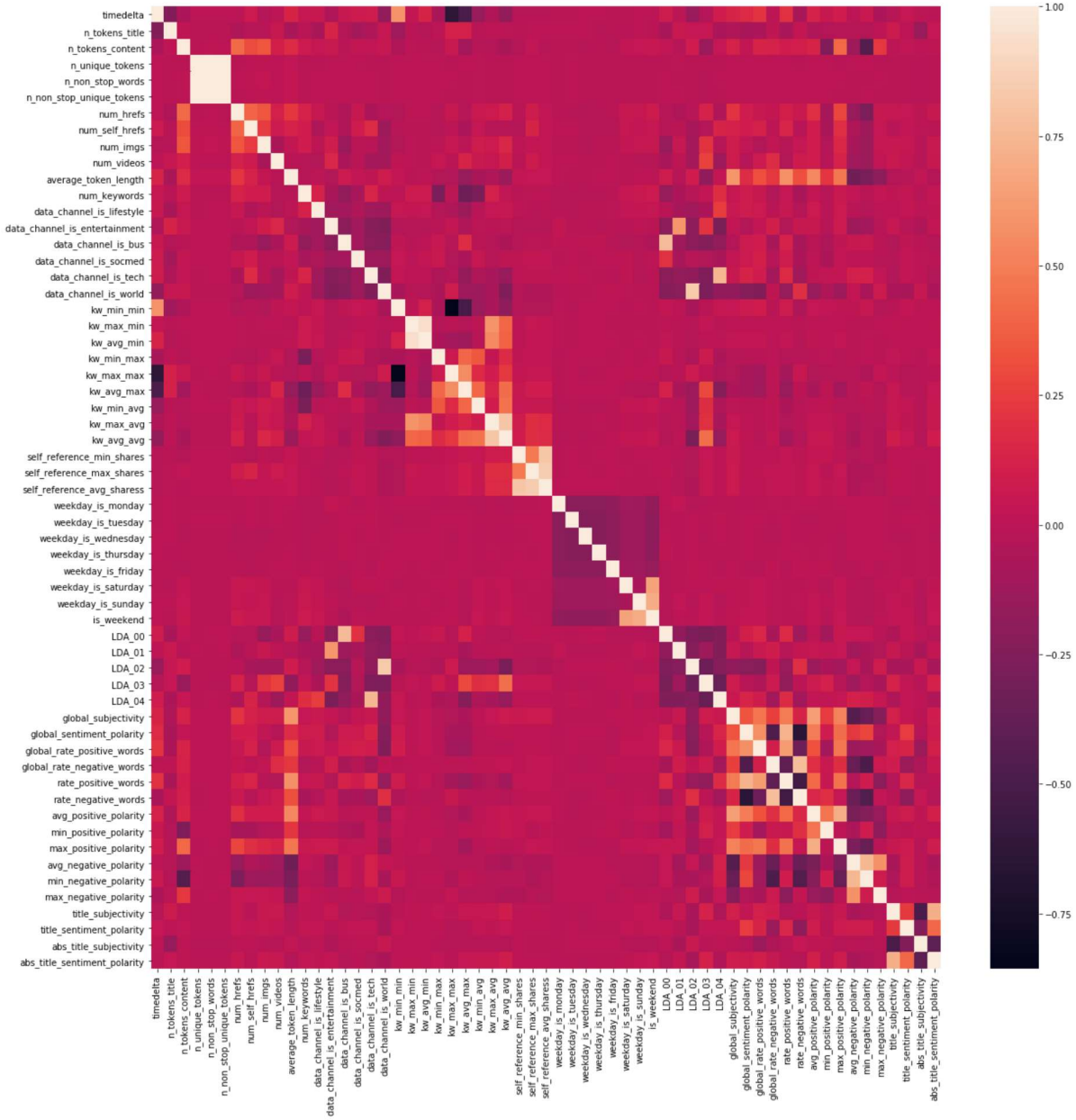
In [30]:
```
#Analysing data and finding dependent and independent variable
X = data.iloc[:, 1:60]
X.head()
```

Out[30]:

	timedelta	n_tokens_title	n_tokens_content	n_unique_tokens	n_non_stop_words	n_non_stop_
0	731.0	12.0	219.0	0.663594	1.0	
1	731.0	9.0	255.0	0.604743	1.0	
2	731.0	9.0	211.0	0.575130	1.0	
3	731.0	9.0	531.0	0.503788	1.0	
4	731.0	13.0	1072.0	0.415646	1.0	

5 rows × 59 columns

In [31]:
```
corr = X.corr()
sns.heatmap(corr)
rcParams['figure.figsize'] = 20, 20
```



In [20]:
```
Y = data["shares"]
Y.head()
```

Out[20]:
```
0     593
1     711
2    1500
3    1200
4     505
Name: shares, dtype: int64
```

In [21]:
```
#Splitting data into training and testing
x_train, x_test, y_train, y_test = train_test_split(X, Y, random_state=5, test
_size=0.20)
```

In [22]:
```
#initializing model and training model
lin_model = LinearRegression()
lin_model.fit(x_train, y_train)

#Predict data
predicted_value = lin_model.predict(x_test)

#Mean Squared Errors
metrics.mean_squared_error(predicted_value, y_test)

#Plotting predicted and test data
plt.scatter(predicted_value, y_test)
plt.show()
```