stock_comment
About Dataset

Originally we were planning to scrap conversation data from Yahoo finance (https://finance.yahoo.com/quote/AA/community). The conversation data is recent and diverse. But web-scrapping consumes a lot of time. After 20 hours running on local machine, only about 30MB of data has been generated. I also tried to run the script on a CIMS server, but Google chrome runs very slowly inside the virtual box. Thus, considering time and data size, I obtained the Stockwits dataset(about 194.3MB) collected by an Udacity Team as an alternative option. The dataset contains messages from Stockwits(a social media app), and those messages are similar to posts on twitter. This dataset is available in the pulic domain and contains sufficient data. More detailed description can be found here (https://vkontech.com/sentiment-analysis-of-stocktwits-messages-using-lstm-in-pytorch/).

Exploratory data analysis & cleansing

Here, I created a schema for the dataframe, called z.show() to present some rows of the dataset. In total, there are 4 columns, 1548010 rows. Column names and types are shown in the printSchema output.

index	message_body	sentiment
0	$FITB great buy at 26.00...ill wait	2
1	@StockTwits $MSFT	1
2	#STAAnalystAlert for $TDG : Jefferies Maintains with a rating of Hold setting target price at USD 350.00. Our own verdict is Buy http://www.stocktargetadvisor.com/toprating	2
3	$AMD I heard there’s a guy who knows someone who thinks somebody knows something - on StockTwits.	1
4	$AMD reveal yourself!	0

Output is truncated to 102400 bytes. Learn more about **ZEPPELIN_INTERPRETER_OUTPUT_LIMIT**

```
root
 |-- index: string (nullable = true)
 |-- message_body: string (nullable = true)
 |-- sentiment: integer (nullable = true)
 |-- timestamp: timestamp (nullable = true)
```

```
filePath2: String = project/output.csv
```

stock_comment

_c0	_c1
1002	$GOO
1003	$FCX
1004	$MOS
1005	$DFS
101	$WMT
102	$KMI
103	$AMD
104	$HBAI
105	$MSF

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult**

```
rawDF2: org.apache.spark.sql.DataFrame = [_c0: string, _c1: string]
filtered: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [_c0: string, _c1: string]
```

```
root
 |-- _c0: string (nullable = true)
 |-- _c1: string (nullable = true)
```

```
joinedDF: org.apache.spark.sql.DataFrame = [index: string, message_body: string ... 4 more fields]
```

index	message_body	sentiment	times
	ETUP&utm_campaign=social_tracking#/WYNN?key=5f008ceb-7e89-4999-a2da-c893680bc755		
644	Short sale volume (not short interest) for $WPX on 2018-06-28 is 44%. http://shortvolumes.com/?t=WPX via @shortvolumes	-1	2018-
381	$WMT is gonna be cheaper than a $BJ. ALSO $AST	1	2018-
90	$WMT Wal-Mart (Dividend Aristocrat)	0	2018-

Output is truncated to 102400 bytes. Learn more about **ZEPPELIN_INTERPRETER_OUTPUT_LIMIT**

```
root
 |-- index: string (nullable = true)
 |-- message_body: string (nullable = true)
 |-- sentiment: integer (nullable = true)
 |-- timestamp: timestamp (nullable = true)
 |-- _c0: string (nullable = true)
 |-- _c1: string (nullable = true)
```

stock_comment

```
newDF: org.apache.spark.sql.DataFrame = [index: string, message_body: string ... 3 more fields]
```

index	message_body	sentiment
0	$FITB great buy at 26.00...ill wait	2
1	@StockTwits $MSFT	1
2	#STAAnalystAlert for $TDG : Jefferies Maintains with a rating of Hold setting target price at USD 350.00. Our own verdict is Buy http://www.stocktargetadvisor.com/toprating	2
3	$AMD I heard there’s a guy who knows someone who thinks somebody knows something - on StockTwits.	1

Output is truncated to 102400 bytes. Learn more about **ZEPPELIN_INTERPRETER_OUTPUT_LIMIT**

```
import scala.collection.mutable.WrappedArray
convert_list: org.apache.spark.sql.expressions.UserDefinedFunction = SparkUserDefinedFunction($Lambda$4440/1186732433@7c3fdee7,ArrayType(StringType,true),List(Some(class[value[0]: array<string>])),Some(class[value[0]: array<string>]),None,true,true)
```

index	message_body	sentiment	times
0	$FITB great buy at 26.00...ill wait	2	2018-
1	@StockTwits $MSFT	1	2018-
2	#STAAnalystAlert for $TDG : Jefferies Maintains with a rating of Hold setting target price at USD 350.00. Our own verdict is Buy http://www.stocktargetadvisor.com/toprating	2	2018-
3	$AMD I heard there’s a guy who knows someone who thinks somebody knows something - on	1	2018-

Output is truncated to 102400 bytes. Learn more about **ZEPPELIN_INTERPRETER_OUTPUT_LIMIT**

```
flatted: org.apache.spark.sql.DataFrame = [list_of_symbols: array<string>, index: string ... 4 more fields]
```

stock_comment

list_of_symbols	index	message_body	se

Output is truncated to 102400 bytes. Learn more about **ZEPPELIN_INTERPRETER_OUTPUT_LIMIT**

flatted_symbol	timestamp
$FB	2018-07-01
$ACN	2018-07-01
$JCP	2018-07-01
$WHR	2018-07-01
$BXP	2018-07-01
$IRBT	2018-07-01
$ECL	2018-07-01
$TWX	2018-07-01

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult**

```
groupedDF: org.apache.spark.sql.DataFrame = [flatted_symbol: string, timestamp: date ... 1 more field]
```

flatted_symbol	timestamp
FB	2018-07-01
ACN	2018-07-01
JCP	2018-07-01
WHR	2018-07-01

BXP	2018-07-01
IRBT	2018-07-01
ECL	2018-07-01
TWX	2018-07-01

stock_comment

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult**

```
removeDF: org.apache.spark.sql.DataFrame = [flatted_symbol: string, timestamp: date ... 1 more field]
```

flatted_symbol	timestamp	avg(sentiment)
FB	2018-07-01	0.3764705882352941
ACN	2018-07-01	-0.3333333333333333
JCP	2018-07-01	0.5
WHR	2018-07-01	0.0
BXP	2018-07-01	0.0
IRBT	2018-07-01	0.0
ECL	2018-07-01	0.0
TWX	2018-07-01	-1.0

Output is truncated to 1000 rows. Learn more about **zeppelin.spark.maxResult**

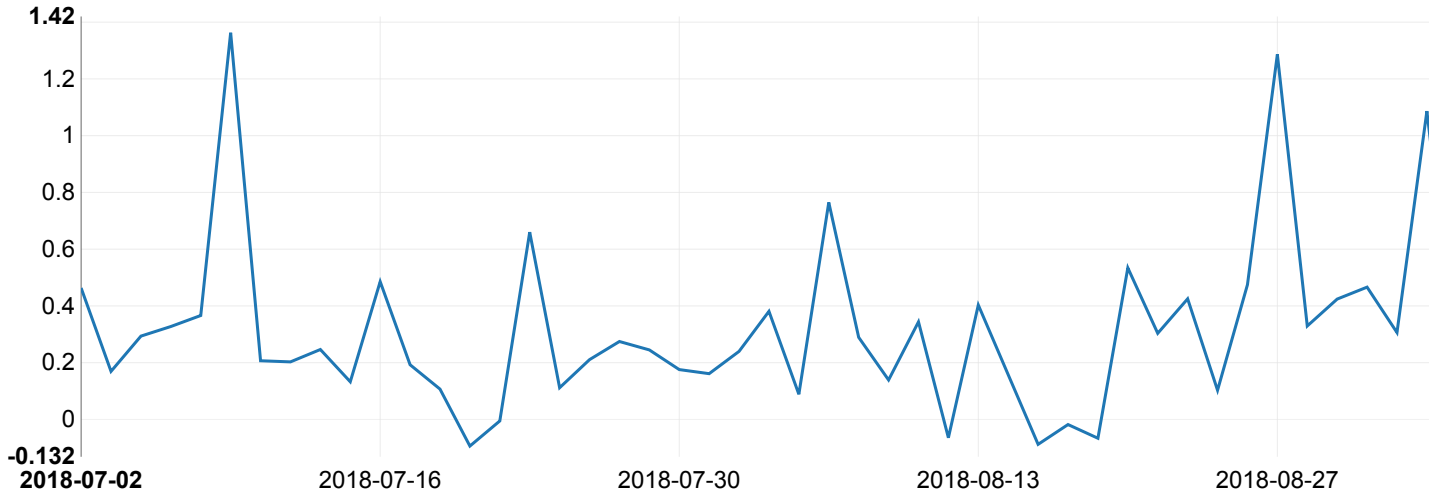
```
dfWithWeekNumber: org.apache.spark.sql.DataFrame = [flatted_symbol: string, timestamp: date ... 2 mor
e fields]
df4: org.apache.spark.sql.DataFrame = [flatted_symbol: string, timestamp: date ... 3 more fields]


nflx: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [shiftedDate: date, avg(sentiment): do
uble]
```

shiftedDate	avg(s

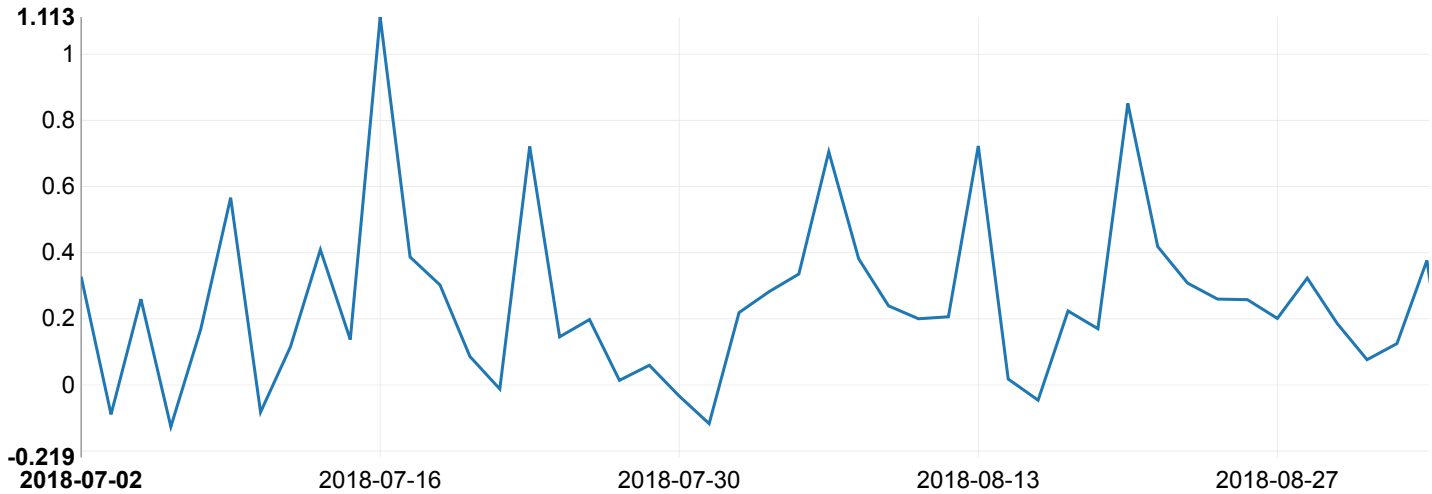
```
finalDF: org.apache.spark.sql.DataFrame = [flatted_symbol: string, shiftedDate: date ... 4 more field
s]
```

stock_comment




```
res62: Int = 4
```

```
TSLA: org.apache.spark.sql.Dataset[org.apache.spark.sql.Row] = [shiftedDate: date, avg(sentiment): do
uble]
```




```
outputPath: String = project/cleanedComments.csv
```