Insights and Explanations--Text Analysis Python Part

After cleaning the data, I want to do some analysis with **Hive**.

First, upload the *Cleaned_News_Dataset.csv* in Hive view as table *news_data*.

Then, do the following 3 steps of analysis:

1.Calculate average headline length and body length:

```
SELECT
    source,
    AVG(LENGTH(title)) AS avg_headline_length
FROM
    news_data
GROUP BY
   source;


SELECT
    source,
    AVG(LENGTH(text)) AS avg_body_length
FROM
    news_data
GROUP BY
   source;
```

Outcome:

```
| source | avg_headline_length |
|---------|--------------------|
| fake     | 94.194054                |
| true     | 64.658291                |

| source | avg_body_length |
|---------|----------------|
| fake       | 2488.619601               |
| true       | 2319.029796               |
```

2.Monthly analysis and weekly analysis of fake news:

```
SELECT
    DATE_FORMAT(TO_DATE(FROM_UNIXTIME(UNIX_TIMESTAMP(date,
'MMMM dd, yyyy'))), 'yyyy-MM') AS year_month,
    source,
    COUNT(*) AS count
FROM
    news_data
WHERE
   date IS NOT NULL AND date != ''
   AND source = 'fake'
GROUP BY
    DATE_FORMAT(TO_DATE(FROM_UNIXTIME(UNIX_TIMESTAMP(date,
'MMMM dd, yyyy'))), 'yyyy-MM'),
    source
ORDER BY
    year_month ASC, count DESC;

SELECT
    DATE_FORMAT(TO_DATE(FROM_UNIXTIME(UNIX_TIMESTAMP(date,
'MMMM dd, yyyy'))), 'EEEE') AS day_of_week,
    source,
    COUNT(*) AS count
FROM
    news_data
WHERE
   date IS NOT NULL AND date != ''
```

```
    AND source = 'fake'
GROUP BY
    DATE_FORMAT(TO_DATE(FROM_UNIXTIME(UNIX_TIMESTAMP(date,
'MMMM dd, yyyy'))), 'EEEE'),
    source
ORDER BY
    FIELD(day_of_week, 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday',
'Saturday', 'Sunday');
```

Outcome:

```
| year_month | source | count |
|------------|--------|-------|
| 2015-05    | fake   | 338   |
| 2016-01    | fake   | 695   |
| 2016-02    | fake   | 687   |
| 2016-03    | fake   | 679   |
| 2016-04    | fake   | 610   |
| 2016-05    | fake   | 1012  |
| 2016-06    | fake   | 477   |
| 2016-07    | fake   | 465   |
| 2016-08    | fake   | 438   |
| 2016-09    | fake   | 486   |
| 2016-10    | fake   | 519   |
| 2016-11    | fake   | 513   |
| 2016-12    | fake   | 496   |
| 2017-01    | fake   | 580   |
| 2017-02    | fake   | 467   |
| 2017-03    | fake   | 541   |
| 2017-04    | fake   | 362   |
| 2017-05    | fake   | 827   |
| 2017-06       | fake     | 399     |
| 2017-07       | fake     | 312     |
| 2017-08       | fake     | 313     |
| 2017-09       | fake     | 227     |
| 2017-10       | fake     | 199     |
| 2017-11       | fake     | 142     |
| 2017-12       | fake     | 84      |
```

```
| day_of_week | source | count |
|-------------|--------|-------|
| Monday        | fake     | 1620   |
| Tuesday       | fake     | 1764   |
| Wednesday     | fake     | 1829   |
| Thursday      | fake     | 1860   |
| Friday        | fake     | 1777   |
| Saturday      | fake     | 1460   |
| Sunday        | fake     | 1558   |
```

3.Count the number of true and false news by subject:

```
SELECT
    subject,
    source,
    COUNT(*) AS count
FROM
    news_data
WHERE
    subject IS NOT NULL AND subject != ''
GROUP BY
    subject, source
ORDER BY
```

```
    source ASC, count DESC;
```

Outcome:

```
|      subject           |     source     |     count     |
|------------------|------------|-----------|
| politics                 | fake            | 6838           |
| News                         | fake            | 9050           |
| left-news                | fake            | 4459           |
| Government News    | fake            | 1570           |
| US_News                  | fake            | 783             |
| Middle-east            | fake            | 778             |
| politicsNews          | true            | 11220         |
| worldnews              | true            | 9991           |
```