

Data processed by spark needs to be visualized. Kibana is an amazing dashboard that can be used to visualize such data. In order to pass data from Spark to Kibana, processed data has to be passed into elasticsearch over an index. Kibana uses this index to get the data.

Now, Kibana dashboard can do many types of analysis in itself and it requires data in particular format. This can be done by adding a mapping to our index in elasticsearch. Mapping used in our project:

```
PUT /myproject
```

```
{  
  
  "mappings": {  
  
    "america": {  
  
      "properties": {  
  
        "createAt": {"type": "date"},  
  
        "device": {"type": "text",  
          "fields": {  
            "keyword": {  
              "type": "keyword",  
              "ignore_above": 256  
            }  
          }  
        },  
  
        "hashtags": {"type": "text",  
          "fields": {  
            "keyword": {  
              "type": "keyword",  
              "ignore_above": 256  
            }  
          }  
        }  
      },  
  
      "location": {"type": "geo_point"  
  
    },  
  
    "sentiment": {"type": "text",  
      "fields": {  
        "keyword": {  
          "type": "keyword",
```

```
        "ignore_above": 256
      }
    },
    "text": {"type": "text",
      "fields": {
        "keyword": {
          "type": "keyword",
          "ignore_above": 256
        }
      }
    },
    "user_lang": {"type": "text",
      "fields": {
        "keyword": {
          "type": "keyword",
          "ignore_above": 256
        }
      }
    },
    "user_name": {"type": "text",
      "fields": {
        "keyword": {
          "type": "keyword",
          "ignore_above": 256
        }
      }
    },
    "place": {"type": "text",
      "fields": {
        "keyword": {
          "type": "keyword",
          "ignore_above": 256
        }
      }
    },
    "followers": {"type": "integer"},
    "friends": {"type": "integer"},
    "statusCount": {"type": "integer"},
    "textSize": {"type": "integer"}
  }
}
```

```

    }
  }
}

```



















In the above mapping, it is clearly visible that we need to use map the incoming data with correct data types. One of the most important thing to take care of is, if one needs to perform aggregations on text data then following part needs to be added to that field. By making its as keyword will make it aggregatable in Kibana

```

"fields": {
  "keyword": {
    "type": "keyword",
    "ignore_above": 256
  }
}

```

Fields for this mapping in Kibana with its datatypes:

name ⇅	type ⇅	format ⇅	searchable ⓘ ⇅	aggregatable ⓘ ⇅	analyzed ⓘ ⇅	excluded ⓘ ⇅	controls
sentiment	string		✓		✓		
device.keyword	string		✓	✓			
hashtags	string		✓		✓		
user_name	string		✓		✓		
createAt	date		✓	✓			
sentiment.keyword	string		✓	✓			
place	string		✓		✓		
user_lang	string		✓		✓		
text	string		✓		✓		
textSize	number		✓	✓			
user_lang.keyword	string		✓	✓			
statusCount	number		✓	✓			
friends	number		✓	✓			
user_name.keyword	string		✓	✓			
followers	number		✓	✓			
_source	_source						
location	geo_point		✓	✓			
text.keyword	string		✓	✓			

place.keyword	string	✓	✓		
device	string	✓		✓	
hashtags.keyword	string	✓	✓		
_id	string				
_type	string	✓	✓		
_index	string				
_score	number				