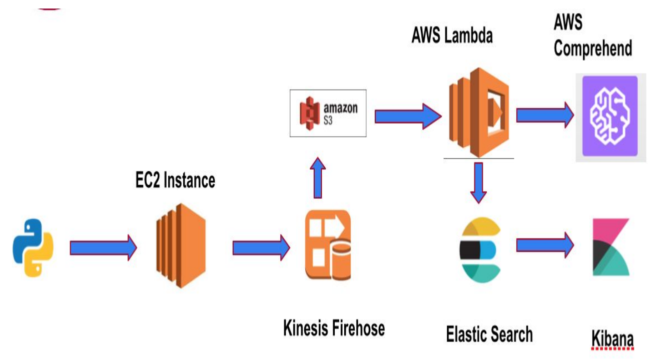
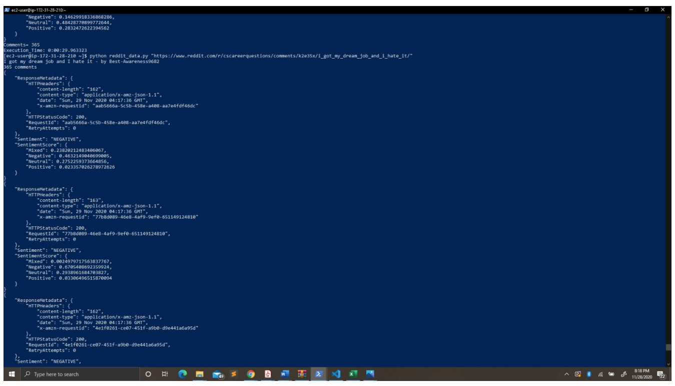
AWS Setup Steps:



System Configuration

- Elastic Compute Cloud (EC2): Amazon Linux AMI 2018.03.0 (HVM) 64 bit
- S3 bucket for storing the comments
- Kinesis Firehose delivery stream: Source as S3 bucket. Retry duration to ES is 60 seconds
- AWS Lambda: trigger added to the function are Amazon Cloud Watch logs, Amazon Comprehend, Amazon S3, Identity and Access Management
- Elastic Search: Version 6.5, instance type t2.small.elasticsearch, EBS Storage of 10 GB.

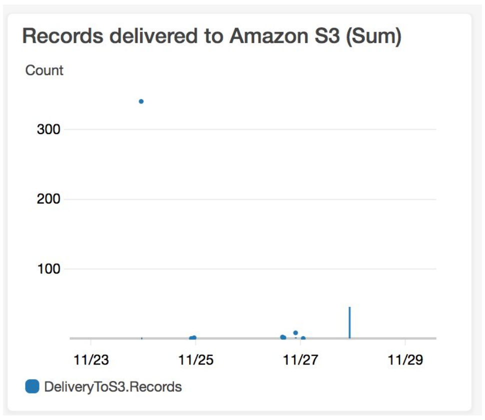
Step 1. Data Collection and the Instance: A VM instance is created on the AWS platform and a python script is executed to retrieve replies on a Reddit post passed as an URL to the script, by accessing a Python Reddit API Wrapper.

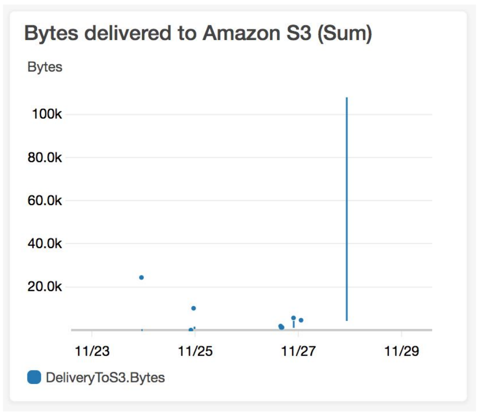


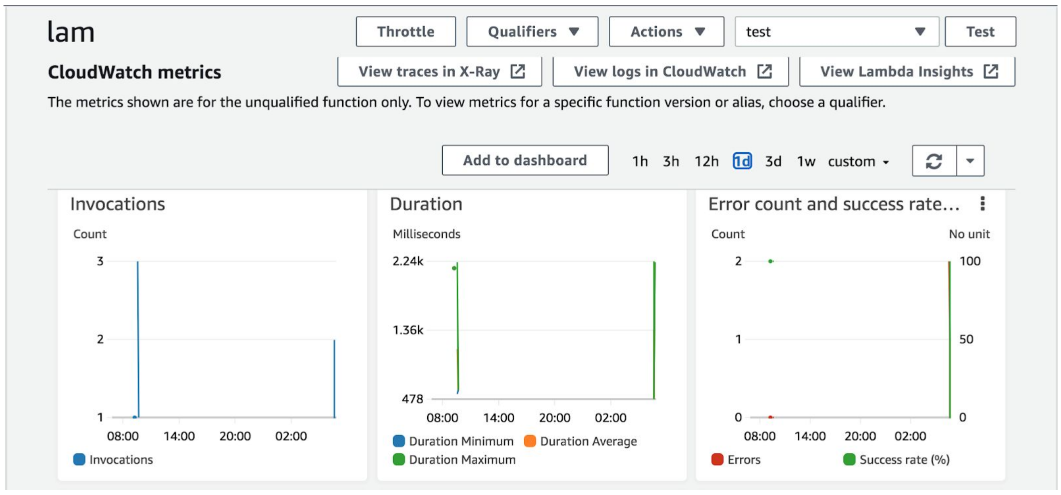
```
        "Negative": 0.14629918336868286,
        "Neutral": 0.48428770899772644,
        "Positive": 0.2832472622394562
    }
}
Comments= 365
Execution_Time: 0:00:29.963323
[ec2-user@ip-172-31-28-210 ~]$ python reddit_data.py "https://www.reddit.com/r/cscareerquestions/comments/k2e35x/i_got_my_dream_job_and_i_hate_it/"
I got my dream job and I hate it - by Best-Awareness9682
365 comments
{
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "162",
            "content-type": "application/x-amz-json-1.1",
            "date": "Sun, 29 Nov 2020 04:17:36 GMT",
            "x-amzn-requestid": "aab5666a-5c5b-458e-a408-aa7e4fdf46dc"
        },
        "HTTPStatusCode": 200,
        "RequestId": "aab5666a-5c5b-458e-a408-aa7e4fdf46dc",
        "RetryAttempts": 0
    },
    "Sentiment": "NEGATIVE",
    "SentimentScore": {
        "Mixed": 0.23820212483406067,
        "Negative": 0.4632149040699005,
        "Neutral": 0.2752259373664856,
        "Positive": 0.023357026278972626
    }
}
{
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "163",
            "content-type": "application/x-amz-json-1.1",
            "date": "Sun, 29 Nov 2020 04:17:36 GMT",
            "x-amzn-requestid": "77b8d089-46e8-4af9-9ef0-651149124810"
        },
        "HTTPStatusCode": 200,
        "RequestId": "77b8d089-46e8-4af9-9ef0-651149124810",
        "RetryAttempts": 0
    },
    "Sentiment": "NEGATIVE",
    "SentimentScore": {
        "Mixed": 0.0024979717563837767,
        "Negative": 0.6705408692359924,
        "Neutral": 0.2938961684703827,
        "Positive": 0.03306496515870094
    }
}
{
    "ResponseMetadata": {
        "HTTPHeaders": {
            "content-length": "162",
            "content-type": "application/x-amz-json-1.1",
            "date": "Sun, 29 Nov 2020 04:17:36 GMT",
            "x-amzn-requestid": "4e1f0261-ce07-451f-a9b0-d9e441a6a95d"
        },
        "HTTPStatusCode": 200,
        "RequestId": "4e1f0261-ce07-451f-a9b0-d9e441a6a95d",
        "RetryAttempts": 0
    },
    "Sentiment": "NEGATIVE",
```

Step 2: Data ingestion using Kinesis Firehose

Below are the graphs which show how much data and when the data is delivered to S3 using AWS firehose

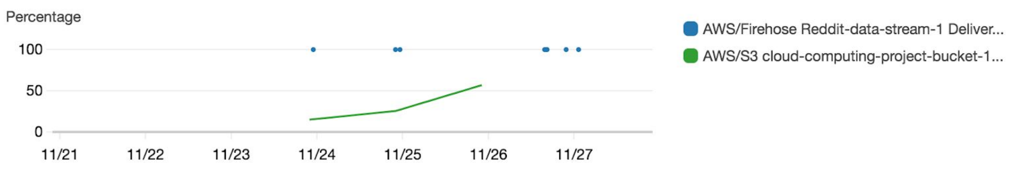




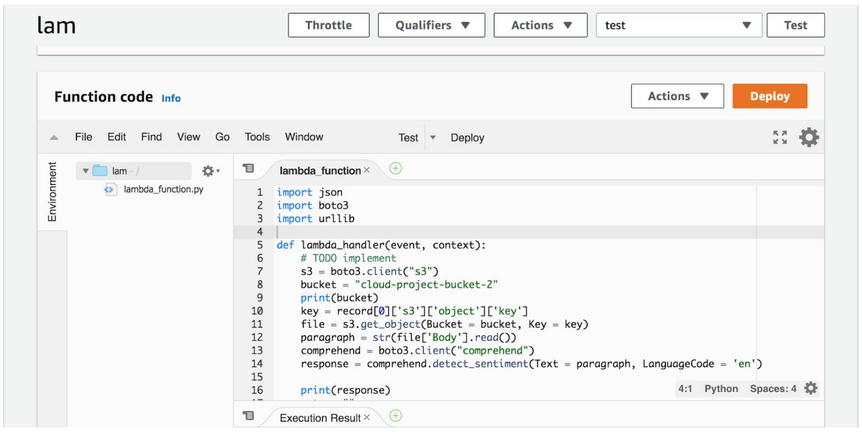


CloudWatch metrics

The metrics shown are for the unqualified function only. To view metrics for a specific function version or alias, choose a qualifier.



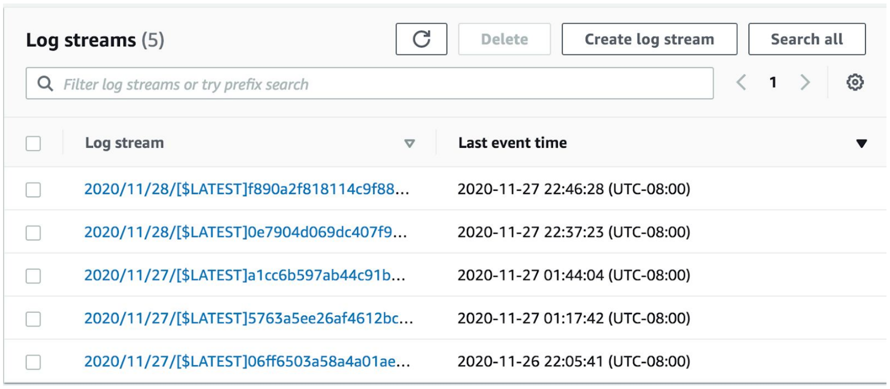
Step 3: Invocation of lambda function and a snapshot of logs



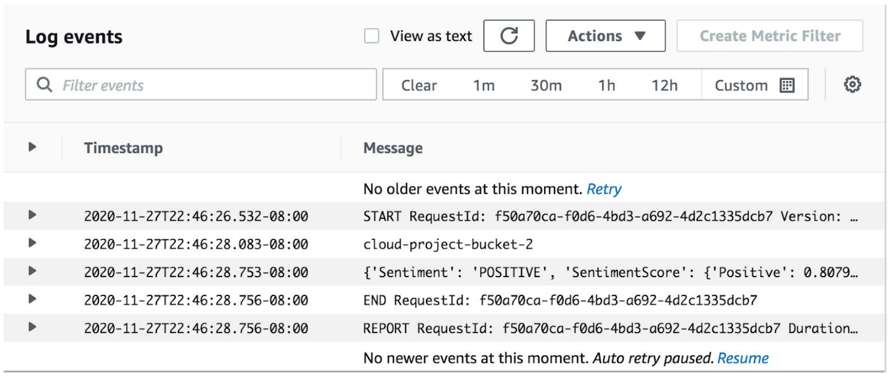
```
import json
import boto3
import urllib

def lambda_handler(event, context):
    # TODO implement
    s3 = boto3.client("s3")
    bucket = "cloud-project-bucket-2"
    print(bucket)
    key = record[0]['s3']['object']['key']
    file = s3.get_object(Bucket = bucket, Key = key)
    paragraph = str(file['Body'].read())
    comprehend = boto3.client("comprehend")
    response = comprehend.detect_sentiment(Text = paragraph, LanguageCode = 'en')

    print(response)
```

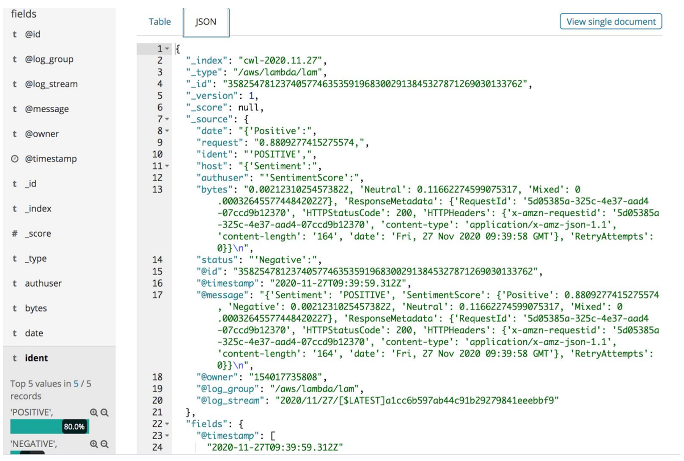


Log stream	Last event time
2020/11/28/[$LATEST]f890a2f818114c9f88...	2020-11-27 22:46:28 (UTC-08:00)
2020/11/28/[$LATEST]0e7904d069dc407f9...	2020-11-27 22:37:23 (UTC-08:00)
2020/11/27/[$LATEST]a1cc6b597ab44c91b...	2020-11-27 01:44:04 (UTC-08:00)
2020/11/27/[$LATEST]5763a5ee26af4612bc...	2020-11-27 01:17:42 (UTC-08:00)
2020/11/27/[$LATEST]06ff6503a58a4a01ae...	2020-11-26 22:05:41 (UTC-08:00)



```
Timestamp                       Message
                                No older events at this moment. Retry
2020-11-27T22:46:26.532-08:00   START RequestId: f50a70ca-f0d6-4bd3-a692-4d2c1335dcb7 Version: …
2020-11-27T22:46:28.083-08:00   cloud-project-bucket-2
2020-11-27T22:46:28.753-08:00   {'Sentiment': 'POSITIVE', 'SentimentScore': {'Positive': 0.8079…
2020-11-27T22:46:28.756-08:00   END RequestId: f50a70ca-f0d6-4bd3-a692-4d2c1335dcb7
2020-11-27T22:46:28.756-08:00   REPORT RequestId: f50a70ca-f0d6-4bd3-a692-4d2c1335dcb7 Duration…
                                No newer events at this moment. Auto retry paused. Resume
```

Step 4: Snapshot of the data from Elastic search



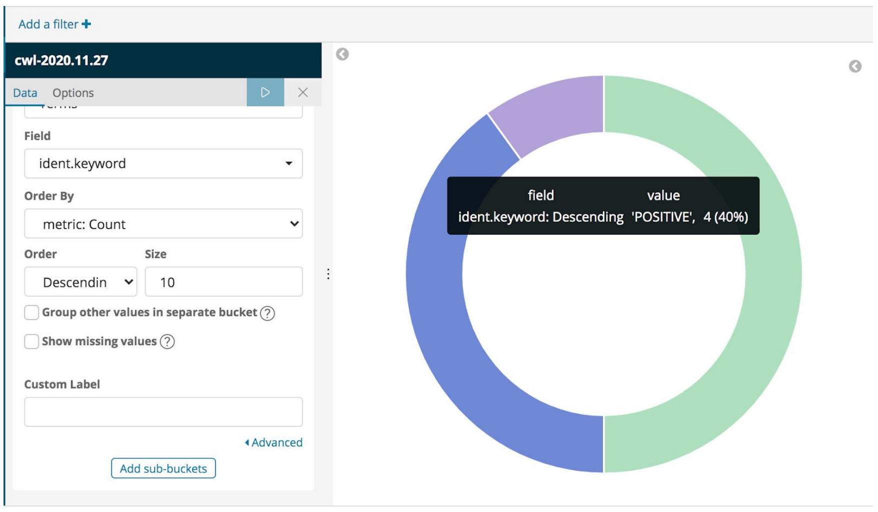
Top 5 values in 5 / 5 records

'POSITIVE', 80.0%

'NEGATIVE',

```
{
  "_index": "cwl-2020.11.27",
  "_type": "/aws/lambda/lam",
  "_id": "35825478123740577463535919683002913845327871269030133762",
  "_version": 1,
  "_score": null,
  "_source": {
    "date": "{'Positive':",
    "request": "0.8809277415275574,",
    "ident": "'POSITIVE',",
    "host": "{'Sentiment':",
    "authuser": "'SentimentScore':",
    "bytes": "0.00212310254573822, 'Neutral': 0.11662274599075317, 'Mixed': 0
        .00032645577448420227}, 'ResponseMetadata': {'RequestId': '5d05385a-325c-4e37-aad4
        -07ccd9b12370', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '5d05385a
        -325c-4e37-aad4-07ccd9b12370', 'content-type': 'application/x-amz-json-1.1',
        'content-length': '164', 'date': 'Fri, 27 Nov 2020 09:39:58 GMT'}, 'RetryAttempts':
        0}}\n",
    "status": "'Negative':",
    "@id": "35825478123740577463535919683002913845327871269030133762",
    "@timestamp": "2020-11-27T09:39:59.312Z",
    "@message": "{'Sentiment': 'POSITIVE', 'SentimentScore': {'Positive': 0.8809277415275574
        , 'Negative': 0.00212310254573822, 'Neutral': 0.11662274599075317, 'Mixed': 0
        .00032645577448420227}, 'ResponseMetadata': {'RequestId': '5d05385a-325c-4e37-aad4
        -07ccd9b12370', 'HTTPStatusCode': 200, 'HTTPHeaders': {'x-amzn-requestid': '5d05385a
        -325c-4e37-aad4-07ccd9b12370', 'content-type': 'application/x-amz-json-1.1',
        'content-length': '164', 'date': 'Fri, 27 Nov 2020 09:39:58 GMT'}, 'RetryAttempts':
        0}}\n",
    "@owner": "154017735808",
    "@log_group": "/aws/lambda/lam",
    "@log_stream": "2020/11/27/[$LATEST]a1cc6b597ab44c91b29279841eeebbf9"
  },
  "fields": {
    "@timestamp": [
      "2020-11-27T09:39:59.312Z"
```

Step 5: Data Visualization using Kibana



Field: ident.keyword

Order By: metric: Count

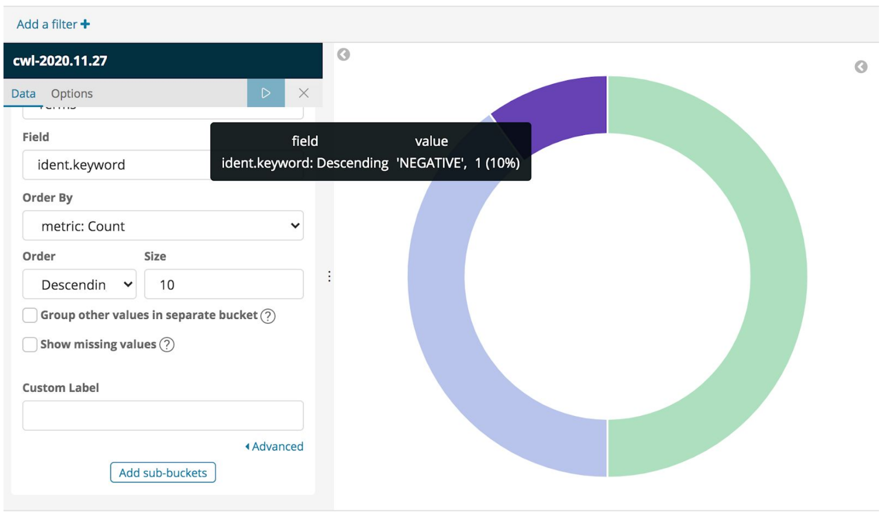
Order: Descendin   Size: 10

Group other values in separate bucket

Show missing values

Custom Label

field	value
ident.keyword: Descending	'POSITIVE', 4 (40%)



Field: ident.keyword

Order By: metric: Count

Order: Descendin   Size: 10

Group other values in separate bucket

Show missing values

Custom Label

field	value
ident.keyword: Descending	'NEGATIVE', 1 (10%)