Has llegado a la página guardada en caché de http://blogs.technet.com/b/machinelearning/archive/2015/04/29/fun-with-ml-stream-analytics-and-powerbi-observing-virality-in-real-time.aspx

A continuación aparece una instantánea de la página web tal y como aparecía en **29/04/2015** (la última vez que nuestro rastreador la visitó). Esta es la versión de la página que se usó para la clasificación de los resultados de búsqueda. Puede que la página haya cambiado desde la última vez que la guardamos en caché. Para ver lo que puede haber cambiado (sin la información destacada), ve a la página actual.

Has buscado: **Fun** with **ML**, **Stream Analytics** and PowerBI - Observing Virality in **Real Time** Hemos resaltado las palabras coincidentes que aparecen en la página que está a continuación.

Bing no se hace responsable del contenido de esta página.

| Data Development | Data Quality Services | OLTP | Integration Services | Data Security & Storage | Data in the Cloud | Machine Learning |

Machine Learning Blog

Fun with **ML**, **Stream Analytics** and PowerBI – Observing Virality in **Real Time**

ML Blog Team   29 Apr 2015 3:13 PM   0

This post is authored by Corom Thompson and Santosh Balasubramanian, Engineers in Information Management and Machine Learning at Microsoft

This is a **fun** story of how we were expecting perhaps 50 users for a test but – in the end – got over 35,000 users and saw the whole thing unfold in **real time**.

We were playing with Microsoft's newly released Face detection API's through a webpage called http://how-old.net. This page lets users upload a picture and have the API predict the age and gender of any faces recognized in that picture. We sent email to a group of several hundred people asking them to try the page for a few minutes and give us feedback – optimistically hoping that at least 50 people would give it a shot. We monitored our **real time analytics** dashboard to track usage and, within a few minutes, the number of people using the site vastly exceeded the number of people we had sent our email to. We watched the usage quickly spread across continents, and, 3 hours later, here is what our dashboard showed:
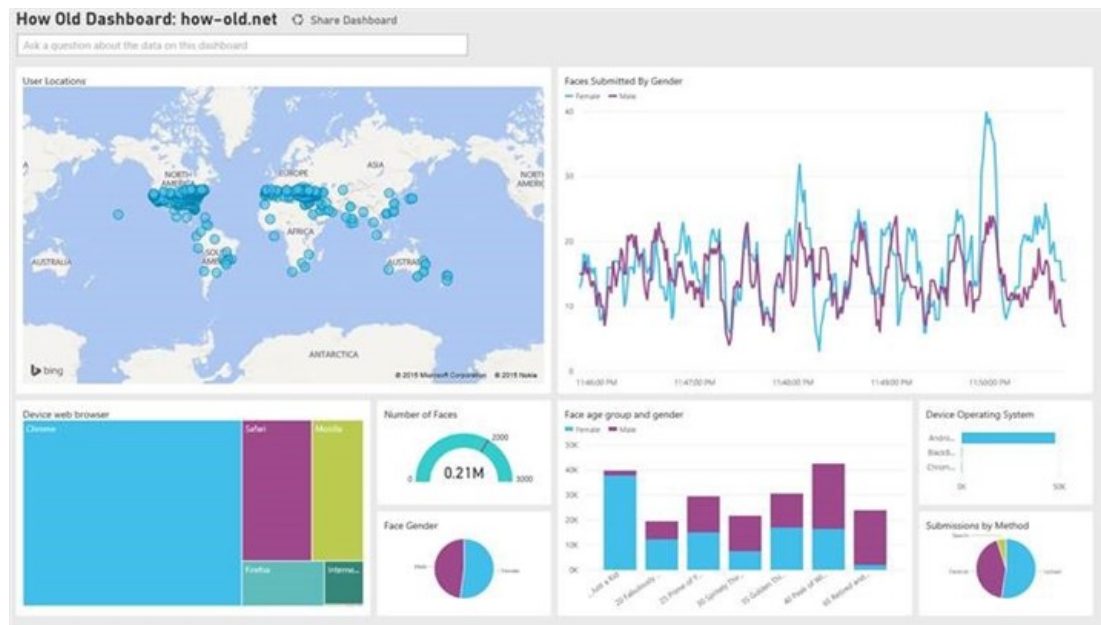
We were shocked. Within a few hours, over 35,000 users had hit the page from all over the world (about 29k of them from Turkey, as it turned out – apparently there were a bunch of tweets from Turkey mentioning this page). What a great example of people having **fun** thanks to the power of **ML**!

Unique count of users by country or region

COUNTRY OR REGION	UNIQUE COUNT
Turkey	29 K
United States	5.6 K
United Kingdom	885
Germany	555
Canada	442

We also got **real time** insights to learn more about how people were using this tool. For instance, we assumed that folks would not want to upload their own pictures but would prefer to select from pre-canned images such as what they found online. But we what we found out was that over half the pictures analyzed were of people who had uploaded their own images. We used this insight to improve the user experience and did some additional testing around image uploads from mobile devices. Given the power of such **real time** insights, we could have also tested additional ideas around things such as page layout and get instant feedback around the same.

So What's The Magic Behind All This?

This may be hard to believe but it took a couple of developers just a day to put this whole solution together, starting with the pipeline from the web page to the Machine Learning APIs to the **real time streaming analytics** and **real time** BI. This turned out to be a great example of the power of Azure services. The key components of this solution are:

1. Extracting the gender and age of the people in these pictures.
2. Obtaining **real time** insights on the data extracted above.
3. Creating **real time** dashboards to view the above results.

Extracting Gender and Age

We wanted to create an experience that was intelligent and **fun** could capture the attention of people globally, so we looked at the APIs available in the Azure Machine Learning Gallery. The gallery contains many finished intelligent services such as Face, Speech, and Vision which are part of a new suite called Project Oxford from Bing and Microsoft Research. The Face API has a demo page that uses the API to detect and extract information about faces in a photograph. We found the ability of the face API to estimate age and gender to be particularly interesting and chose this aspect of it for our project. To make the experience more **fun** we used the face API alongside the Bing Search API from the Azure marketplace to create http://how-old.net.

In addition to age and gender, we also got additional information such the User Agent string of the users of the web site, the latitude and longitude of location from where the picture was uploaded and more. This is represented in following JSON document:

```
[ {     "event_datetime": "2015-04-27T01:48:41.5852923Z",
"user_id": "91539922310b4f468c3f76de08b15416", "session_id": "fbb8b522-6a2b-457b-bc86-62e286045452",
"submission_method": "Search",
"face": { "age": 23.0, "gender": "Female" },
"location_city": { "latitude": 47.6, "longitude": -122.3 },
"is_mobile_device": true, "browser_type": "Safari", "platform": "iOS", "mobile_device_model": "IPhone"
} ]
```

Real Time Insights

Once we have extracted the information we want from the uploaded pictures and web logs, we collect and analyze the data obtained from thousands of users uploading pictures to this site in **real time**. We use a set of Microsoft Azure **streaming** services for this purpose.

We collect all the data using Azure Event Hubs, a highly scalable publish-subscribe ingestor that can intake millions of events per second. We use the Event Hubs API to **stream** the JSON document from the web page when the user uploads a picture. Note that the picture is not saved, just the metadata extracted in the JSON file gets streamed to Event Hubs.

Next we need a **stream** processing service to aggregate and process the information from thousands of users uploading pictures in **real time**. For this we use Azure **Stream Analytics** (ASA), a fully managed low latency high throughput **stream** processing solution. ASA lets you write your **stream** processing logic in a very simple SQL -like language.

An example of using ASA, if you want to get the count of "gender" in a 10 second window with a result written every second, all you need is a very simple query to aggregate this information:

```
SELECT
System.Timestamp AS OutTime,
    Face.gender AS Gender,
    Count(*) AS Count
FROM
    StreamInput
GROUP BY HoppingWindow(second,10,1),
        Face.gender
```
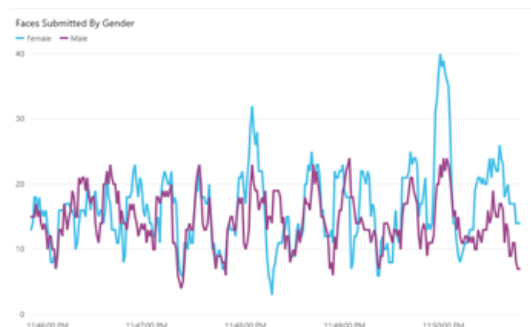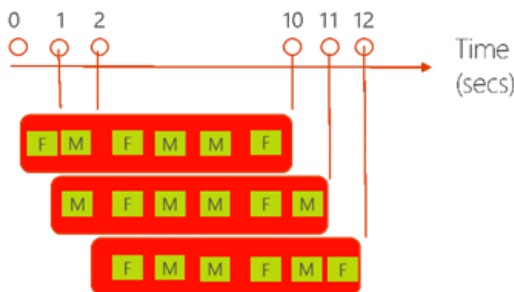

In the above query, we are selecting the **Time** when the result is written (OutTime), Gender, and count of gender. StreamInput is the alias of the Event Hub to which the **streaming** log data is sent. This is done in a hopping window of 10 seconds, with a hop of 1 second. This query gives the aggregate count of Female and Male faces in the uploaded pictures and this information can be displayed in a dashboard. You can have multiple **stream** processing queries on data coming from the same Event Hub.

Real Time Dashboards

We use PowerBI to display the results in a **real time** dashboard. All we did was to choose PowerBI as the output of our **stream analytics** job (click here to learn how). Then we went to http://app.powerbi.com, and selected the dataset and table created by ASA. There is no additional coding needed to create your **real time** dashboards.
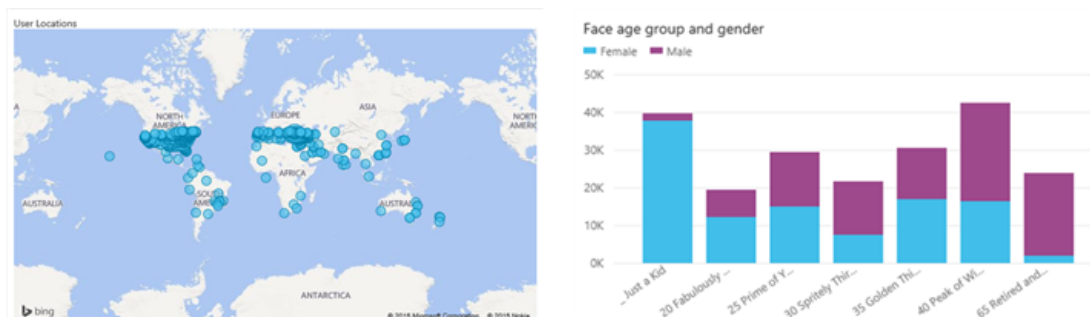
In this example we have a couple of **stream analytics** queries. One aggregates age into an age range and passes in other fields such as location to PowerBI, and the other is the query mentioned above. PowerBI lets you easily create a variety of visualizations including maps, line charts, tree view charts and more. Charts get updated in **real time** as data is generated

by users uploading pictures at http://how-old.net. Additionally you can ask natural language questions too (e.g. "What's the count of people using IOS by gender by age group") and the charts that are displayed as a result of such a question can be pinned to the **real time** dashboard.

Go try out http://how-old.net for yourself (#HowOldRobot) – we hope you have **fun** with it and are inspired to create your own solutions using Azure services and the APIs available in the **ML** Gallery.

Corom and Santosh

Tweet 0   Like 0   Share   ◼ Save this on Delicious

Loading ...