# NN Briefing - 24/08/2023

Attendees: Anza Kutama, Alexander Venizelos

Event time: August 24, 2023 8:00 PM → 9:00 PM

CHECK OUT MY GITHUB FOR NEW UPDATES ON 25/08/2023: https://github.com/TheRealVeni/Neural_Network_with_Datasets **BACKGROUND** (by Alexander Venizelos)

Alright so as I had constructed a premature NN that kind of went through random websites do access some data points, analysing this with Anza we found that an issue is that a lot of universities don't seem to have their data made accessible on their own website. What does this mean? This means that:

1. As I suggested, we should integrate an AI Bot tailored to our needs (Web scraping) at the initial layer of our NN. We should do this because:

a) We want to get accurate data with as few iterations as possible b) We want a correct CSV file for the NN to process c) We don't want to search for all of the data on our own.

So essentially what does this mean: We will make an AI Bot that has the capabilities of a LANGUAGE MODEL, but is tailored to our needs of web scraping. Simpler: A small Neural Network (NN1) inside the big one (NN2).

1. This also means that a .edu TLD requirement would not work 100% of the time, as some unis choose to publish their data elsewhere or simply on Wikipedia.
2. And this is a point of referral to bullet No1. Some universities decide to publish their data in PDF format, e.g. for their school's pamphlet or something like this. This means that we would also need the NN1 to be able to read through PDFs.

The upsides:

The data we will be getting will be provided with speed, reliability and accuracy of about 95%. We will not need to enter any kind of data into an AI Bot again.

Exceptions: Only when/if we find an extremely reliable AI Bot should we stop the development of NN1.

Negatives: It will take a little more time to code it, but that time saves at the end as the data is automatically transferred into the Main part of the Neural Network.

**MEETING RECAP (**by @Alexander Venizelos)

At first, Anza and I discussed some .ipymb and .py files I uploaded on Jira and Notion. You guys are also all welcome to take a look at my GitHub to see what exactly is behind this NN. We went on to discuss some of the filters as well as some of the problems that may arise with the use of the neural network.

Filters

- We want a 2year filter on the data we retrieve
- We don't want the NN to return empty handed so well have a function in place that simply tells it to rerun some of the scraping in other web locations

Problems

Accuracy, how the data will be properly accessed by the NN

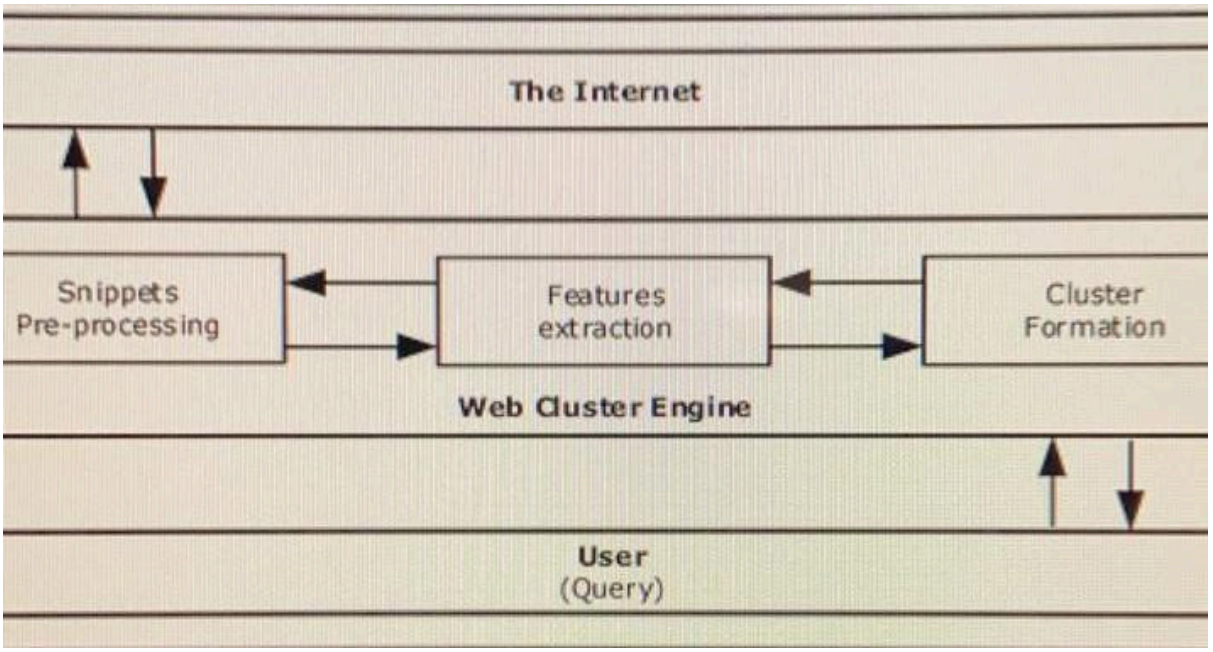
$$P(c|x,y) = (f_c * g)(x,y) = \sum_{m=-M}^{M} \sum_{n=-M}^{M} f_c(x-m, y-n) \cdot g(m,n)$$

$$P\left(c\middle|elem_{position}\right) = P\left(c\middle|(l,r,b,t)\right) = \frac{1}{(r-e)\cdot(b-t)} \sum_{l \le x \le r} \sum_{t \le y \le b} r(c|x,y)$$

What you're seeing in the picture is the 2 equations called "Spatial Probability Distribution equations" These are the theoretical mainframe/definition of how the Neural Network is going to be retrieving the data. Essentially the way I formed these equations is so that the NN doesn't just extract anything, it collects:

- A lot of the data it seems useful
- Less and less of some data it may think are not pertaining to its search objectives

You can imagine this as a 2D Gaussian Distribution.




[Web scraping viz]

Web scraping viz

Afterwards we came up with the flowchart I sent in the 2nd picture which depicts our visualisation of how we're going to proceed with the web scraping.

In case you're worrying this is a little too much out in the open, don't worry; there are a lot of scientific articles on exactly what we are discussing right now.

After all this and my idea to integrate a simpler version of an AI language model that has as a sole function web crawling and providing us with the data that it finds, we came to the conclusion that we need to monitor the accuracy of our NN (what we just discussed with Amjad) using an "analytics feature", as Anza mentioned.

After this Anza requested the PDF scraping part which we can bypass using Amjad's creative method.

Anyway, that's all. Take a look at the Jira source code and PDF I sent yesterday as well as my GitHub, where more updates will follow during the day.