

**User:**

Hi, nice to meet you.

**Assistant:**

Hello! It's nice to meet you too. Thank you for taking the time to speak with me today about the Entry-Level Machine Learning Engineer position at Mozilla.ai. To start off, could you tell me about your experience with machine learning and any projects you've worked on in this area?

**User:**

Yeah, sure, I'd be happy to. To start with, I've graduated from the University of Kent with a distinction in artificial intelligence, and I have a couple of years of experience as well in machine learning in general. At my current job at Vascon.ai, the company is building a patent analysis system to identify and detect potential patent infringement, and I've played a big part of this by building a retrieval augmented generation powered web scraping pipeline, which allows a large language model to take an input of a patent. It allows it to extract the key claims within the patent. And then it gets another large language model to generate a large amount of Google search queries to massively web scrape the internet to identify as many products as possible that may potentially infringe on these patents. It takes all these web pages and vectorizes them and places them into a vectorized database. And then we use an embeddings model to compare similarity between the patent claims that we are provided. And then we use an embeddings model to compare similarity between the patent claims that we are provided. And then we use an embeddings model to compare similarity between the patent claims that we are provided. From the customer's patent and the created vectorized database. And then for any matches that are found, we then take the entire web page, we provide it to another large language model, and then we get this large language model to generate a single large report for each provided patent to us, which we then provide to our customer. Beyond this, I've also got some other experience. And several other key fields of machine learning, namely computer vision and deep learning. So if you'd like to hear about those, I'd be happy to talk about those.

**Assistant:**

Thank you for sharing that. Your experience with RAG and LLMs for patent analysis is quite relevant. Could you elaborate on any specific challenges you faced in implementing the web scraping pipeline and how you overcame them?

**User:**

Yeah of course. One of the key challenges that I faced in building this pipeline was the large-time model that was in charge of generating search queries on Google to identify products that might be infringing on our patents. Initially this was not actually generating very helpful search queries, it was just generally searching, it was doing very broad searches such as for example product infringing on XXX feature right and that's not very useful so it took a good amount of prompt engineering to get the the search generation algorithm to actually generate searches that were a bit more abstract that were able to more broadly cover the entire field of existing products to which we could then use to compare and identify any potential infringement.

**Assistant:**

That's an interesting challenge. How did you approach evaluating the performance of your model, particularly in terms of its accuracy in identifying potential patent infringements?

**User:**

Yeah, sure. There is a very big challenge when it comes to evaluating the systems that involve using large language models, mainly because the output of large language models always has a bit of variance in them. So it's very hard to get deterministic outputs. So instead, what we did is we mass tested our system. So we conducted experiments on 100 patents. And for each one of these, we went through and compiled essentially a score for each patent on how many false flags were identified in terms of potentially infringing products. So that's it. And we defined a false flag as any potential infringing product that does not actually infringe on the patent or one that is already owned by our client's patent. So, for example, if there is a product that is built off of a given patent and the system identifies that product, that should not be identified. And so we also counted that as a fail. Then, once we conducted the experiment, we identified some very interesting patterns in our system in that for very popular patents that are very broad, such as a couple of patents that we ran involving 5G communications, it had a very large amount of red flags, basically, of failures, where it identified products that it thought infringed on the patent, but that did not. However, ironically, for the more obscure patents that are very abstract and are not very well publicized, it did a very good job of identifying products that might be infringing upon these.

**Assistant:**

Thank you for that detailed explanation. You've mentioned experience with computer vision and deep learning. Could you briefly describe a project where you applied these techniques?

**User:**

Yeah, sure. I'd be happy to. I did a project last year where I built an image recognition based resistor decoder, which was mainly used to help the people with poor eyesight or that are poor of vision. And as the first step in this process, I had to get a image recognition system which was able to identify any resistors within a picture that was captured. So to do this, I made use of TensorFlow to build a convolutional neural network and then train this convolutional neural network on images of resistors that I had manually acquired from manually taking pictures of resistors. So to start, I built my model with, first of all, several convolutional layers and then after that, several normal dense layers. And I made use of a bunch of techniques such as dropout to make certain nodes no longer connect to each other temporarily for certain batches in order to be able to avoid, model to overfit. So then once the resistors were identified, I then built a little pipeline to identify bounding boxes around the resistors in the image. After this was done, I then built a K nearest neighbor classifier. Sorry, a K. Yeah, a K nearest neighbor classifier, which was then able to identify the colors. So I was able to put the colors in each band of the resistor and then was able to calculate the resistance of the resistor and speak out loud to the visually impaired engineers.

**Assistant:**

Thank you for sharing that project. It's an interesting application of computer vision. Last question: How do you stay updated with the latest developments in machine learning, particularly in areas relevant to Mozilla.ai's work?

**User:**

Sure. I have two primary ways in which I stay up to date on new developments in machine learning. The first one is I have a feed set up in Google Feeds. So, sorry, more specifically, what I mean by that is on your home screen on Android, if you swipe right, it takes you to a curated Google feed. I've set up a Google account, which has only got interest in machine learning and artificial intelligence. And so on my phone now, when I swipe right, I get new curated up-to-date information on machine learning, be it new research papers, new finding, new open source models that are published on Hugging Face, all of the above. The other way that I stay up to date is through YouTube. YouTube has a large amount of great educational content, particularly on machine learning. And there's one channel that stands out to me called Two Minute Papers that reviews most of the key, like large developments in machine learning. And specifically goes through those research papers and elaborates on them and explains them, which allows me to obtain information from bleeding edge technology that is published in research papers. In a very simple way. Very good condensed format. Allow me to stay up to date.

**Assistant:**

Thank you for your time. We appreciate you sharing your experiences and insights with us today.