Cameron Stine

https://github.com/cs-code-cs/MachineManifesto/blob/main/MachineManifesto.ipynb

Above is the link to the art project notebook and below is a link to the entire project.

https://github.com/cs-code-cs/MachineManifesto

*Please note that the file lshw.html is my computer information.

Information Society and Its Future

Computers and technology for the longest time have held a place of innovation in the minds of people and they were seen by many as a way to reduce costs through the process of digitization. It is a lot easier to distribute movies on a streaming platform than to pay for the shipping costs to send DVDs to a local store than the person drives to the store and picks it up and drives home. This is just one example of the way computers saved money as well as energy by automating and digitizing processes. They were seen as greener than printing a book for example to have it stored on a server and be accessible to everyone instead of having the book be shipped throughout the whole world. As more and more tasks that humans used to do became automated and digitized the scope became not to automate tasks humans were doing but to automate tasks that humans could not do or even fathom. Thus, we saw the rise of computational and machine learning fields which allowed the computer to interpret and analyze massive amounts of data and automate processes that no amount of human power could compete with. This allowed a few amount of people to consolidate power through machine processes and control almost all of the information flow and a massive amount of wealth in this country and in the world.

I decided on this project because of a combination of reasons and the main one being is the framework that emerged from the book, *The Age of Surveillance Capitalism*, by Shoshana Zuboff, in which corporations are able to extend such extreme surveillance over the users of their products and able to track ridiculous amounts of data about an individuals life. I understand the architecture exists for tracking all of the movements people make on websites so why was there no ability for me to easily track the energy consumption being used by a simple model? Computing technologies are so good at tracking that it made no sense to me that there was no way to track the energy consumption my machine learning models were using. I can scrape the web for millions of data point at the tip of my finger but at what cost? It was a desire to find out the cost of my actions that made me want to explore energy consumption in machine learning.

According to Bloomberg cloud computing uses about 1% of the energy the earth consumes. This might seem like an insignificant amount of energy in the overall scheme of things but through this energy usage cloud computing companies like, Google, AWS, Microsoft Azure are able to pierce other industries and essentially control information flow and economies. The overall development of these things which should have allowed people to communicate remotely and saved energy have heightened capitalistic tendencies that have made us pillage the planet for rare resources and fight wars over energy.

The field of Green AI is often overlooked and while reducing energy costs can save money that is often not the biggest factor for creating a model and it is more about accuracy and automating some sort of task or creating some sort of value. Green AI also focuses on embedded systems which means creating systems that create the most efficiently possible ways which means there would likely be a device that is solely programmed to do this task and they likely wrote the code on a client device in C or C++. Because these programming languages are more energy efficient than a non compiled language

/ a runtime language like Python there is little focus given to improving Python when it is the most common programming language for Data Science. This has sort of led to a rejection of Python in the Green AI field but with the ease of using Python it does not make much sense especially when a consumer or researcher is likely not going to use an embedded system unless they are working for a massive company. That is why I think a project in Python was important.

I picked the domain of natural language processing because that is what I am most knowledgeable in and most trained in. The current problem in the field of natural language processing is the goal set out by instructors, competition leaders, and industry placing a heavy emphasis on the accuracy of models used in computational language models. When I took the natural language class offered in the fall of 2021, I was eager to begin using computational methods to analyze text. Seeing trends and interpreting a massive amount of textual data was interesting to me. We were instructed and it was actually part of the class to integrate Google Colab, a cloud based computing platform for collaboration. In previous machine learning classes they usually tried to refrain from using products within the Google ecosystem and opted for using Jupyter Notebook which runs on the user's system. The cloud computing required to be used made a disconnect between the user's system and the modeling being done. We would submit our assignments not only to the grader but our results would be sent to a Kaggle (also Google owned) competition board to see which team was the most accurate. As well as critiquing papers written about NLP which showed that even half a percentage of more accurate scoring would allow a person to get published and I also saw how some of these massive corporations like Google and Baidu had departments that worked more like academic researchers in the field of NLP and they publishing research papers similar to how professors and students publish them at universities.

Methodology:

I took the three projects we were assigned for the NLP class and recorded the amount of energy each one took to run. This was a harder process than I initially realized and is almost impossible to accurately track. Not did I not have the tracker on when I was writing the code for the assignment and tracking the energy usage at each stage of the project. There were also limitations such as only being able to do this using a client based system which utilizes a built-in feature on Linux machines. This is called RAPL, or the running average power limit and tracks the energy consumption used by devices on the system I was running these on. The RAPL does not support external graphics cards but running this on my Lenovo laptop I was able to track the energy usage of the integrated GPU which there is a call to use the GPU instead of the CPU for one of the projects.

Using the program called pyRAPL, a python package that allows the tracking of the usage data and reporting the energy consumption and time, I was able to estimate how much my own program took to run and I attempted to extrapolate based off of this the amount of times my partner and I ran to check these programs and then using the number of teams on Kaggle I attempted to estimate how much the programming alone in the course consumed in energy.

The draw backs here are that cloud computing is impossible to track and that it was required for the project as well as people having various different machines, mine was being tracked on a relatively light operating system compared to Windows or Mac. There is also no way to tell the amount of time people spent on these projects and the amount of time people took to make their models perfect. For example, we had Kaggle competitions for who had the most accurate model and although my team's model were able to obtain A grades on the assignments we were often in the middle of the road for the Kaggle competition meaning people were going out of their way and using methods not taught in class to get a higher accuracy score which almost guarantees these teams were using more energy and more advanced models and likely were testing their models more than our team. Yet there were half the teams reporting scores that were lower than ours who likely missed something or were not scoring better than randomly selected data.

PyRAPL also only reports the energy consumption of key components of the device and there are other processes and hardware that consume power and without a metered connection it is impossible to tell how much energy is being used. This was based on Marcel Garus' work where he used a metered connection to track the difference between the RAPL statistics and the overall metered connection and thus furthering this project would be obtaining one of these devices and comparing the statistics as well as updating his work for a user based out of the United States.

There is also the sunk cost of using the amount of energy used in the creation of my computer and all the hardware that the classroom used to teach the course. As well as the packages and models that were developed at massive companies with no way to understand the energy used to create the models as all these research projects are done behind curtains. This is on purpose to distance the user from the technology and legacy of emissions they are creating.

Findings:

The first project for the NLP class used about 68446 Joules to run once. Now when my partner and I were writing this program we easily ran it over 100 times. We do not know exactly because no one is tracking these sorts of things except maybe machine at Google. No person sits there and counts how many times I ran my Google Colab notebook. This might not seem like a lot of power but given we ran this over 100 times because it does not take that long to run and rerun as well as the fact that there were 102 teams submitting assignments this project likely amassed 698 mJ just from the computations alone that is not including the total power the laptop consumes. This is almost a GJ which according to Genesis Now, an Australian energy efficiency firm, an efficient home consumes 10-30 GJ in an entire year. This was only one project though and there were other factors not being included and then there was the second project which was more about showing you could implement an algorithm by hand which was way more energy efficient since it was less about the findings. The final one was about Natural Machine Translation which took any English and translated it to Shakespearean English and made sense. This was measured statistically. This process consumed a massive amount of energy and used almost 8,000 Joules just to run.

Project:

This is why I decided to use the idea of machine translation and create a project. In the book, *Humanities Data Analysis*, by Folgert Karsdorp, Mike Kestemont, and Allen Riddell which was used in a text mining course I took, they start the preface of the book talking about how useful computational methods are for analyzing text. They end the book with an epilogue entitled "Good Enough Practices" a topic which we discussed in class and sort of changed my perspective on machine learning, they quote Wilson et al and implore people to adopt "good enough practices in scientific community" and allow all levels of computational expertise to be able to explore these practices. I thought of it not only for allowing more access to these models but also the less abstract and lower level the language model is it is simply easier for the anyone to understand. I understood how something could be more statistically like Shakespeare than modern English but when I would test my model on real sentences in would make no sense and just seemed weird and unartistic when the task of translating is a creative and artistic pursuit that allows for a number of different things to do. I decided to create a tool that would allow for a more nuanced and creative solution that did not test whether or not something was statically more Shakespearean. I also did not want to use a western canonical author like Shakespeare so I decided to create a translation of Dr. Theodore Kaczynski's work, *Industrial Society and Its Future.*

I ran some pretty standard analytics and found the most commons words that aren't stopwords (common words, non informative words). I then took these words and came up with things to translate them to and manually entered them and I easily translated the book and replaced these things words without the need for them and I used the results from my query to find ways to translate the Industrial Society into the Information Society. This was a fun serendipitous part of my exploration that focused on getting a result with machine translation that left in room for creativity. It was still more efficient than going through and analyzing the book myself and rewriting it by hand and the best part if every team in my NLP class ran this 100 times it would only cost 17,2612 Joules. It was also good enough to work as an example of machine translation and ties in creativity and artistry into the coding process as well as the main purpose is not to be as statistically accurate as possible. I did not take papers about information technologies and create a corpus and style codification and use that because the goal was not about in my opinion an arbitrary statistic it was about being energy efficient and working with what is available. In doing so I think I wrote some of the most easily understandable code since my first year and did so in a way that prioritizes ecological impacts. Code is an art why is it judged in such finite ways in class when it is subjective and a place for people to explore and be creative. Using statistics to see if my code worked better is like using an arbitrary metric to see if one book, or painting is better than another. Code is art and the things that people can do with it is art and it should not be a tool to embed mathematical processes which consume more and more power. Code should be good enough and allow for serendipitous exploration and allow people to be creative in the way they use it. If it was up to me and I was teaching the NLP class, I would award the winner not on who was the most accurate but which student used the least amount of energy. I would not assign these technical papers to critique instead I would assign ones about people using NLP for social change to show that there are more important things than improving an already working model by 1%. There is subjectivity in natural language processing and in a field about making machines interpret language as natural as possible it felt so unnatural and not nuanced. To make things more natural wouldn’t mistakes more likely occur wouldn’t people translate works writing out or in certain opinions. Why is a translation of some work from the 1950s a lot different than one from the 2020s when they are both translations to English of the same work? Computer science especially when getting into linguistics is not a hard science as people think it is and usually doing this embeds systems of whiteness and Eurocentricity that occur along with language. Why train on this data that is known to be flawed, why not explore creative ways for people to complete the task differently with different goals in mind? Hopefully, I have demonstrated another approach to machine translation which focuses on the energy consumption and I have shown there are other goals that we can have when programming not the ones ingrained in us at school and being taught to us.

Sources:

https://genesisnow.com.au/

https://pyrapl.readthedocs.io/en/latest/index.html

https://github.com/marcelgarus

https://github.com/jupyter-server/jupyter-resource-usage

Pesce, M. (2021). Cloud Computing’s Coming Energy Crisis: The Cloud’s Electricty Needs Are Growing Unsustainably. IEEE Spectrum https://spectrum.ieee.org/cloud-computings-coming-energy-crisis

Karsdorp, F., Kestemont, M.. Riddell, A., (2021). Humanities Data Abalysis: Case Studies with Python. United States: Princeton University Press

Zuboff, S. (2019). The Age of Surveillance Capitalism: The Fight for a Human Future at the New Frontier of Power. United States: PublicAffairs.