Assignment 1

Author - Ramya Tekumalla
Email - tramya457@gmail.com
Date of Submission - 08/06/2020
Github Link -

1. Summary

The first chapter of the book "Dive into Deep Learning" was a high level explanation of the different components and fragments of Machine Learning. It covered topics right from methodologies (Unsupervised, Supervised, etc) to important terms like loss, cross entropy. I really enjoyed the examples for each Machine Learning concept / term. They made me understand the method used in a high level scope. Usually, when solving problems we only look at deep problems which lead us to lose sight of the higher impact on why machine learning was used in the first place and these examples which were taken from day to day life really gave an amazing insight of how machine learning is everywhere. My favorite example was Training and Test error examples. Understanding the roots of Machine Learning was an eye opener. The "re-discovering" statement of Machine Learning algorithms like "CNN, "LSTM" was fascinating because I did not know that these algorithms were used in the past. I thought they were written after Machine Learning started taking off.

The whole introduction to Machine Learning chapter can be summarized in 4 parts,

1) Problem
2) Data
3) Model / Methodology
4) Computation Power

Identifying the problem is the first key task. Not every problem requires Machine Learning to provide a solution. For example, correcting a spelling error in general English text does not require Machine Learning. While translation from other languages or correcting spellings to a particular field like biology or pharmacovigilance might require Machine Learning (Tekumalla & Banda, 2020). The data retrieval, cleaning and feature extraction is as important as the model or methodology used to solve the problem; otherwise as illustrated in the book, it would be Garbage In and Garbage Out (GIGO). There are several methods / algorithms for solving one type of problem (Classification, Clustering, Regression) and identifying the best model always depends on the data. The key is to solve the problem using a best model and not to improve the performance of an "unused" model. The computation power plays a key role in training a model and obtaining improved performance. With GPUs, the speed of the training phase improves drastically, resulting in improving the model or altering it to better the performance and solve the problem. Finally, Machine Learning must be utilized to automate a tedious human task and try to perform "better" or at least as good as the human mind can.

2. Exercises

1) Which parts of code that you are currently writing could be “learned”, i.e., improved by learning and automatically determining design choices that are made in your code? Does your code include heuristic design choices?

   My current research involves mining tons of Twitter data. For mining the twitter data, we use the Twitter API and retrieve the Json files and on the retrieved files, we work on the Tweet text. For several domain specific research, we would use “keywords” to retrieve tweets relevant to the domain (Banda et al., 2020). Through this approach, we retrieve tons of data, however, not all the retrieved data is relevant. It would be great if we can incorporate a machine learning algorithm within the twitter API so we can only retrieve relevant tweets instead of obtaining the tweets and then filtering the tweets for relevance. This approach would save tons of computational costs and memory.

2) Which problems that you encounter have many examples for how to solve them, yet no specific way to automate them? These may be prime candidates for using deep learning.

   One constant problem that I observe when I deal with Deep Learning is, there are several papers with promising algorithms, but the code doesn't work or the code is not maintained. Once the paper is published, the code repositories are obsolete. One great solution I could think of was having a Deep Learning algorithm library for all the Deep Learning models. They can be arranged by the task (Classification, NER, translation etc). Sci-kit Learn (Pedregosa et al., 2011) and (Rajapakse, n.d.) are amazing examples on how libraries can be arranged. This way many people can easily access and use the models without having to run into environment or configuration issues (which are often a problem with Deep Learning because of several versions of TensorFlow/ Keras / Pytorch).

3) Viewing the development of artificial intelligence as a new industrial revolution, what is the relationship between algorithms and data? Is it similar to steam engines and coal (what is the fundamental difference)?

   The fundamental concept of the Industrial Revolution was to move from hand produced methods or goods to usage of machines (automation) which still applies to Artificial Intelligence. For example consider a factory that manufactures Chocolate. With the industrial revolution, we could use machines to cut the chocolate in the same shape and have 100 pieces of chocolate in a minute, while it might have taken a person to cut 1 piece of chocolate a minute. The most important thing to note here is, there is a difference between cutting a chocolate in a minute and “learning” how to cut a chocolate according to

different tastes, shapes or sizes (given a context). The learning part is where Machine Learning comes to place. We build an algorithm by learning several contexts, shapes and sizes and test them and finally roll out when the "machine" is ready. I hope I could make some sense with this analogy.

4) Where else can you apply the end-to-end training approach? Physics? Engineering? Econometrics?

   All fields. Any field where there is a need to automate tedious manual work and employ a machine / algorithm to decrease either labor / give some perspective. Few other fields, other than technology where Machine Learning has boomed are
   a) Automotive Industry - Example- Tesla
   b) Mechanical / Metallurgical / Biology / Astronomy / Engineering - In fact all Engineering fields; Microscopic / Solar images can be analyzed. Events can be predicted and analyzed.
   c) Administration
   d) Real Estate

REFERENCES

Banda, J. M., Tekumalla, R., Wang, G., Yu, J., Liu, T., Ding, Y., Artemova, K., Tutubalina, E., & Chowell, G. (2020). A large-scale COVID-19 Twitter chatter dataset for open scientific research - an international collaboration [Data set]. In *A large-scale COVID-19 Twitter chatter dataset for open scientific research - an international collaboration*. https://doi.org/10.5281/zenodo.3941294

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research: JMLR*, *12*(Oct), 2825–2830. http://www.jmlr.org/papers/v12/pedregosa11a

Rajapakse, T. (n.d.). *simpletransformers*. Github. Retrieved August 7, 2020, from

https://github.com/ThilinaRajapakse/simpletransformers

Tekumalla, R., & Banda, J. M. (2020). *Characterization of Potential Drug Treatments for COVID-19 using Social Media Data and Machine Learning*. arXiv. https://arxiv.org/abs/2007.10276