**A Case for Heuristics: Why Simple Solutions Often Win in Data Science**

**In this defence of heuristics, I examine how simple solutions can often be the best port of call when looking to ship data science products**

Photo by [Mickey O'neil](https://unsplash.com/@mickeyoneil?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com?utm_source=medium&utm_medium=referral)

In 2016, Lisa Bodell — the CEO of futurethink and a top speaker at Google events — proposed that [“simplicity is fast becoming the advantage of our time”](https://www.routledge.com/Why-Simple-Wins-Escape-the-Complexity-Trap-and-Get-to-Work-That-Matters/Bodell/p/book/9781629561295). However, within the field of data science and machine learning, we can often prefer more complex solutions, that while can typically lead to incredible results, can also lead to frustration, failures and long lead times.

While this article isn’t a rallying cry to abandon Keras and revert back to Excel, *it is* a gentle reminder to consider utilising simple heuristics to baseline your solution, and even for you to consider shipping them to start with *then* building something more advanced.

In this article, I’ll draw from learnings from a recent research project, and how I used these findings from to inform my day-to-day approach when developing data science solutions. I’ll dig into a definition of heuristics, learning from [Martin Zinkevich’s *Rules of Machine Learning*](https://developers.google.com/machine-learning/guides/rules-of-ml), deep dive into a recent project which looked to identify dangerous photosensitive epilepsy sequences in gifs, and finally summarise the key learnings on utilising heuristics.

**A Gentle Introduction to Heuristics**

In *Rules of Machine Learning*, Zinkevich proposes that heuristics are “a simple and quickly implemented solution(s) to a problem”. Heuristics is often used as a catch-all term for rules-based algorithms or metrics that allow you to categorise or infer a class, or decision about data. Heuristics can be applied to solve a variety of problems including:

* Categorising if an email is spam (such as using a rules-based approach to detect certain words)
* Showing relevant results to a user (such as the most popular results in their country or overall)
* Identifying the highest-performing users in an app or game (by ranking actions or engagement)

While heuristics are considered simple and quick, they might not always be a data scientist’s first choice to solve a problem. In my experience, heuristics can be neglected in favour of more complex solutions up front, and then simplicity takes over when the whizzy, more advanced solution fails. Within my own academic and professional career, this is a situation I have experienced first-hand. In this post, I wanted to share my findings of comparing heuristics with a deep learning solution, and why simple heuristics should often be your first port of call.

**The Challenge**

Recently, I had the opportunity to conduct research into understanding if it was possible to develop solutions to detect photosensitive epilepsy triggers in videos and gifs. It’s worth highlighting, **this work was never destined for production**, and that healthcare for AI solutions that do look to impact people and make decisions should [follow principles developed by researchers at the World Health Organisation](https://www.who.int/news/item/28-06-2021-who-issues-first-global-report-on-ai-in-health-and-six-guiding-principles-for-its-design-and-use) and other leading organisations. However, my challenge was identifying what was possible within the field and establishing a first-pass solution with deep learning.

**Understanding Photosensitive Epilepsy**

Photosensitive epilepsy falls under the broader category of reflex epilepsy, which is when epileptic seizures can be caused by known and “objective specific” triggers, as noted by Okudan and Özkara, two key researchers in this field. In recent years, unfortunately, bad actors have maliciously targeted the photosensitive epilepsy community online. For example, Liana Ruppert, a journalist, was targeted with photosensitive epilepsy-triggering videos after writing about dangerous content within the [video game, Cyberpunk 2077](https://www.washingtonpost.com/video-games/2020/12/31/cyberpunk-2077-seizure/).



Cyberpunk 2077, a video game which featured dangerous sequences at launch. Photo by Stefans02: [Source](https://www.flickr.com/photos/stefans02/51233501047)

South, Saffo and Borkin, in their paper *Detecting and Defending Against Seizure-Inducing GIFs in* Social *Media,* developed a consumer-driven approach to detecting dangerous gifs, which could be used to combat this rise in online targeting. To measure the performance of their tool, they also developed a dataset of dangerous and safe gifs. The gifs were classified as follows:

* Safe: Contains no photosensitive epilepsy triggers
* Flashes: Contains flashing sequences
* Red: Contains transitions to and from saturated red
* Patterns: Contains repeating patterns
* Dangerous: Contains red, patterns or flashes

The development of this dataset opens opportunities to develop machine learning approaches to the challenge of identifying photosensitive epilepsy triggers in gifs, which South, Saffo and Borkin also identify as a future research direction. I aimed to develop a relatively trivial deep learning solution, using a 2D CNN architecture and transfer learning, then measure the performance of this approach on the different dangerous categories identified by South, Saffo and Borkin.

**Deep Learning Approach**

For this project, I had complete flexibility regarding my approach, a long deadline and an inclination to understand*; is this even possible?* From my own experience, the combination of these three factors can often lead data scientists to choose more complex solutions over simpler ones; the latter of which can be guaranteed to deliver by a certain deadline.

I developed a convolutional neural network (using the Xception architecture), leveraging transfer learning to take an input of gif sequences broken down into four images, then coalesced together.

**Why Choose 2D Convolutional Neural Network?**

As part of my research, I identified the advantages and disadvantages of a variety of gif and video classification architectures, including multistream and 3D CNN approaches. For a first pass, I settled on converting the dangerous gif sequences into an image input, with four frames of the gif coalesced together into one single image. An example of the input data for the CNN is shown below:

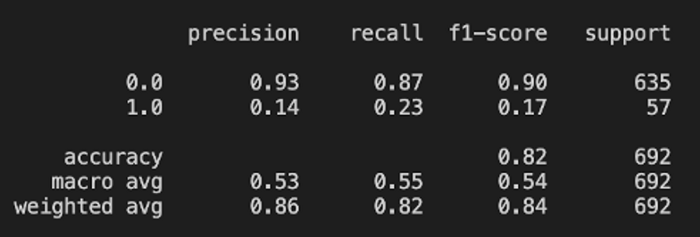


*Example input for the neural network trained on detecting dangerous sequences. Gif from:* South, L., Borkin, M., & Saffo, D. (2022, May 9). [Detecting and Defending Against Seizure-Inducing GIFs in Social Media](http://osf.io/5a3dy). Image created as part of the project.

While the back and forth of choosing the appropriate architecture for this project is outside the scope of this post, I highly recommend [Rehman and Belhaouari’s 2021 review of deep learning for video classification.](https://www.techrxiv.org/articles/preprint/Deep_Learning_for_Video_Classification_A_Review/15172920)

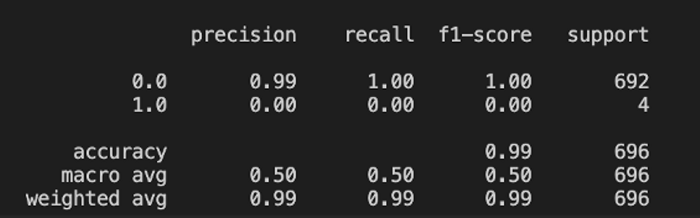
Four models were trained, aiming to detect the photosensitive epilepsy triggers (red, patterns, flashes and all of these, grouped under dangerous). While the dangerous model performed best, almost all models fell short of being viable to detect dangerous gifs. Below, the classification report is provided for the models trained to detect dangerous and saturated red transitions:

***Dangerous Model Performance***



Performance results for ‘Dangerous’ model. Image created by author.

***Red Model Performance***

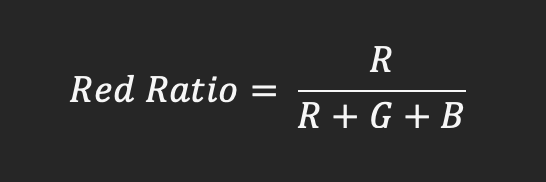


Performance results for ‘Red’ model. Image created by author.

While with a larger dataset, use of sampling techniques and different architecture choices, this performance could undoubtedly be improved, and as is — overfitting as well as optimising for the majority class has happened — my goal at this point was to simply establish what was viable with deep learning. In future, I would look to develop different solutions with this trivial approach as a baseline (alongside heuristics).

**Heuristic Approach**

As part of their research, South, Saffo and Borkin developed three rules-based algorithms for detecting flashes, patterns and red saturation. Each algorithm (or heuristic) is rules-based, making it both explainable and relatively simple to implement. For example, to detect dangerous sequences containing red transitions, the following equations are utilised:



Calculating the red ratio of frames, equation from South, Saffo and Borkin (2021). [Source](https://dl.acm.org/doi/pdf/10.1145/3411764.3445510)

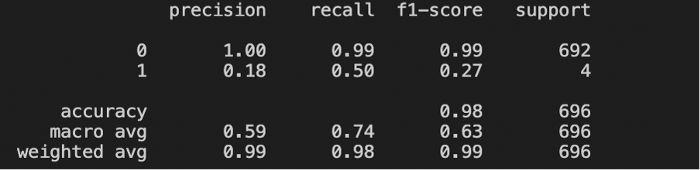
https://miro.medium.com/max/700/1*E-KTAX4xCEs_QusXKZY0bg.png

Calculating Pure Red for frames, equation from South, Saffo and Borkin (2021). [Source](https://dl.acm.org/doi/pdf/10.1145/3411764.3445510)

With their heuristics, South, Saffo and Borkin report strong results, with perfect accuracy, recall and precision for red saturation detection, with **accuracy at 100%, recall at 100% and precision at 67%.**

Once my deep learning models were developed, I wanted to see if even the most rudimentary heuristic could outperform the deep learning models developed. This is quite a biased exercise, given the poor performance of the red saturation model, however, it was still interesting to develop a trivial solution and examine its performance.

For my heuristic, I utilised Numpy and Matplotlib to read the images created for the CNN and then used South, Saffo and Borkin’s RedRatio equation to calculate a score for the image. With the data split between training and test, I developed a cut-off from training data for dangerous red images and applied this to the very small test set developed for the neural network approach. The results of this are shown below:



Performance results from using just the Red Ratio rule on pre-processed gifs. Image created by author.

Even on this very small dataset, the simple heuristic outperformed the deep learning approach. Furthermore, South, Saffo and Borkin’s more advanced rules-based heuristic also performed incredibly well at identifying dangerous content.

**Evaluating Heuristics**

With these findings in mind, some clear reflections and conclusions began to develop on the power of heuristics, and when to move beyond them. My three key learnings in this space are provided below.

**Benchmark with Simple Solutions First**

While South, Saffo and Borkin identify that future research methods could incorporate machine learning, there is an interesting inflection point here. In *Rules of Machine Learning*, Zinkevich proposes:

“ If you think that machine learning will give you a 100% boost, then a heuristic will get you 50% of the way there.”

However, with the dataset developed by South, Saffo and Borkin, they were already able to get 100% of “the way there” with rules-based algorithms. This is an important learning when developing machine learning solutions: **Learn to benchmark with a simple heuristic first and consider if that is enough for production.**

**Closely examine if you have enough data to develop a machine-learning solution**

As Zinkevich highlights out in *Rules of Machine Learning*:

“Machine learning is cool, but it requires data. Theoretically, you can take data from a different problem and then tweak the model for a new product, but this will likely underperform basic [**heuristics**](https://developers.google.com/machine-learning/glossary#heuristic).”

From the red model performance results, it’s possible to see this challenge in action. There is barely any data within the ‘red’ test dataset, meaning the model is incredibly hindered. Of course, it’s viable to try over-sampling techniques (and as part of this project I also implemented Tensorflow’s [class imbalance guidance](https://www.tensorflow.org/tutorials/structured_data/imbalanced_data)). However, it simply was not enough to get over the challenge of a very small minority class.

A simple win going forward here, especially when developing multiple machine learning models at once, is to **assess the differing class sizes, and don’t be afraid to ship heuristics for some problems, utilising machine and deep learning approaches where and when you have more data.**

**Develop your heuristic, then go bigger**

Reviewing the different sizes of the safe and dangerous (dangerous, red, flashes and patterns) classes in [South, Saffo and Borkin’s dataset](https://vis.khoury.northeastern.edu/pubs/South2021DetectingDefendingSeizure/), some classes are more highly populated than others. If you have a ‘good amount’ of data (and this can be quite a nebulous concept depending on your problem), then developing beyond simple heuristics can give you a performance boost in the outcome you’re trying to predict.

While my project was focused on ‘*is this possible*’, with complete freedom and a long deadline, there is a strength to leveraging heuristics over more complex solutions:

Firstly, heuristics are inherently more explainable. Consider South, Saffo and Borkin’s equations for identifying red transition sequences. Explainability has clear benefits in the field of healthcare AI, and while the models developed here were never meant for production or checking if content is safe or not, explainability should be a critical focus if you are developing healthcare AI solutions.

Secondly, heuristics can be implemented quickly. This means that if you can identify new rule additions or changes that can improve performance, you can do so relatively quickly. This has clear benefits when you’re approaching a strict deadline and want to develop a data science solution at pace.

Finally, as Zinkevich notes if you have very little, or no data, heuristics can be developed from previous experiences such as data from different subject areas, user research and even gut feel (although that last one is worthy of an article on its own, as it can come with some major risks).

Before we conclude, it’s worth calling out that my models were predicated on exploring the problem space, without an over-focus on optimisation or eking out performance. If they had been, there is **much more** I could have done in this space, including collecting more data, employing sampling techniques and even reviewing if the architecture chosen (Xception) was the appropriate one, as well as of course handling the overfitting and focus on a majority class. However, as a stake in the ground for deep learning viability for this problem space, this approach achieved its purpose.

**Go Forth and Heuristic?**

In this post, I’ve been a clear proponent of heuristics — and of course Zinkevich’s *Rules of Machine Learning*. However, the technologist in me will always love bold and new solutions, and trialling these. Zinkevich also posits that if you’re developing complex heuristics, then they can become difficult to maintain, and this is where machine learning can step in and perform.

When considering how to implement heuristics into your next project, I highly recommend using them to establish a baseline, and if deadlines are tight, it can even become the solution that you ship. However, there is still obvious value in machine learning and deep learning solutions. Using your heuristic to establish a **baseline to beat**, can be a good grounding exercise for your more complex solutions.

**Selected Bibliography**

Okudan, Z. and Ozkara, C. (2018) “Reflex epilepsy: triggers and management strategies”, Neuropsychiatric Disease and Treatment, Volume 14, pp. 327–337. doi: 10.2147/ndt.s107669.

South, L., Saffo, D. Borkin, D. (2021) Detecting and Defending Against Seizure-Inducing GIFs in Social Media. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. Association for Computing Machinery, New York, NY, USA, Article 273, 1–17. DOI:https://doi.org/10.1145/3411764.3445510

Why Simple Wins — Books by Lisa Bodell — FutureThink (2022). Available at: <https://www.futurethink.com/why-simple-wins>

Zinkevich, M. (2022) Rules of Machine Learning: | Google Developers. Available at: <https://developers.google.com/machine-learning/guides/rules-of-ml>.