Lucas Nakamura
12/2/2025
Economic Data Analysis

Introduction

Artificial Intelligence, or “AI” has exploded in popularity in recent years. Many tech companies have gotten involved in developing these models, each striving to outdo the others. As each model’s “complexity” is largely determined by how many parameters and data points are involved, and computational power requirements scale alongside. Different model types have varying use cases and we would expect different model types or “domains”, to scale differently and the computational power in petaFLOPs required by some models would be greater than that of other domains.

Data Methods

To test this hypothesis, I pulled two separate datasets pertaining to AI models from Our World In Data, with one pertaining to petaFLOPs used, and the other to parameters used. Both sets had model names and domains, and I ended up consolidating all of the data onto one sheet, and then filtering out any models that didn’t have both petaFLOP and parameter values in those respective columns.

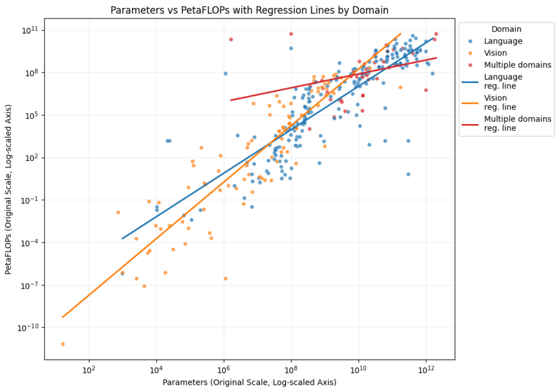
Our World In Data originally pulled the data from Epoch AI, a non-profit research institute for AI that has a database on models published with dates and names. The data is sourced either by the publications directly, or Epoch AI’s estimation, taking into account factors such as model architecture, training data, and training hardware.

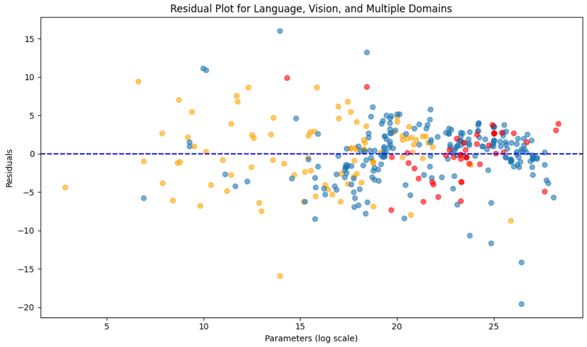
Statistical Methods

Since both of the variables I looked at differed on an exponential scale, I applied a log transform to both, and then plotted and built my model using these transformed values. I used a regression model to determine how parameter count affects the petaFLOPs required.

PetaFLOPs = B_0 + B_1*Parameters + B_2*Domain + B_3*Parameters*Domain + e

Figure 1 shows the relationship between the two variables, sorted by domain.

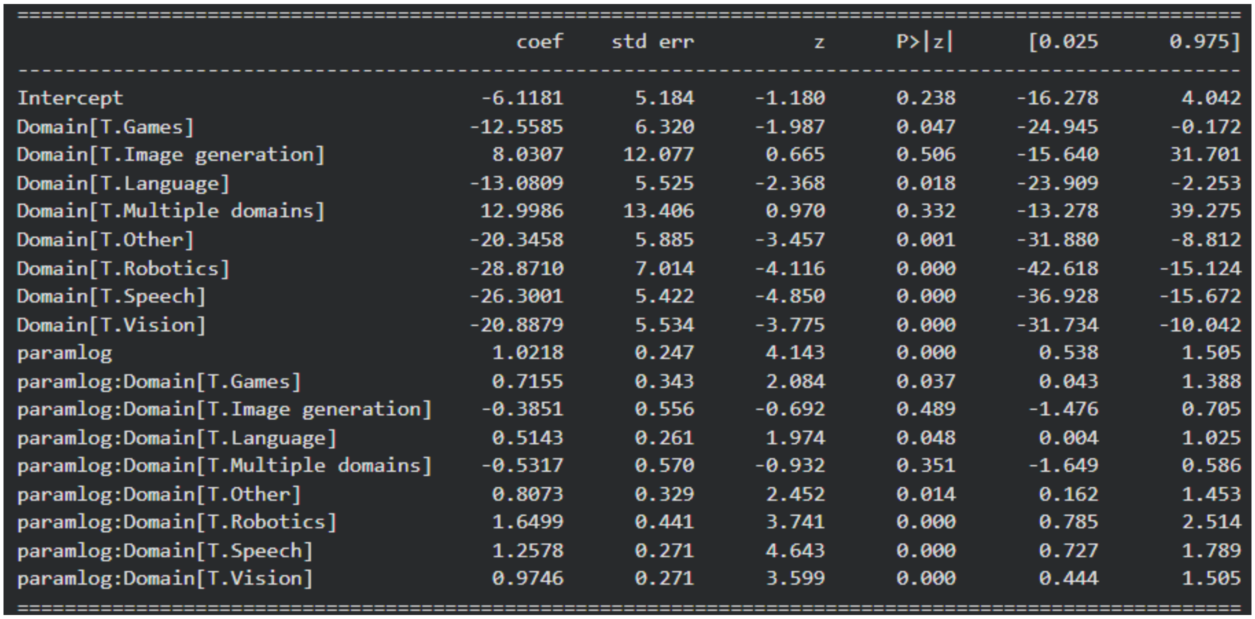




The residual plot shows the possibility of a non-linear relationship, however when separated into domains, the trend only seems to appear in language models, as compared to vision and multiple domains.

Results

The linear regression produced the following output:



```
==================================================================================================
                                          coef    std err          z      P>|z|      [0.025      0.975]
--------------------------------------------------------------------------------------------------
Intercept                              -6.1181      5.184     -1.180      0.238     -16.278       4.042
Domain[T.Games]                       -12.5585      6.320     -1.987      0.047     -24.945      -0.172
Domain[T.Image generation]              8.0307     12.077      0.665      0.506     -15.640      31.701
Domain[T.Language]                    -13.0809      5.525     -2.368      0.018     -23.909      -2.253
Domain[T.Multiple domains]             12.9986     13.406      0.970      0.332     -13.278      39.275
Domain[T.Other]                       -20.3458      5.885     -3.457      0.001     -31.880      -8.812
Domain[T.Robotics]                    -28.8710      7.014     -4.116      0.000     -42.618     -15.124
Domain[T.Speech]                      -26.3001      5.422     -4.850      0.000     -36.928     -15.672
Domain[T.Vision]                      -20.8879      5.534     -3.775      0.000     -31.734     -10.042
paramlog                                1.0218      0.247      4.143      0.000       0.538       1.505
paramlog:Domain[T.Games]                0.7155      0.343      2.084      0.037       0.043       1.388
paramlog:Domain[T.Image generation]    -0.3851      0.556     -0.692      0.489      -1.476       0.705
paramlog:Domain[T.Language]             0.5143      0.261      1.974      0.048       0.004       1.025
paramlog:Domain[T.Multiple domains]    -0.5317      0.570     -0.932      0.351      -1.649       0.586
paramlog:Domain[T.Other]                0.8073      0.329      2.452      0.014       0.162       1.453
paramlog:Domain[T.Robotics]             1.6499      0.441      3.741      0.000       0.785       2.514
paramlog:Domain[T.Speech]               1.2578      0.271      4.643      0.000       0.727       1.789
paramlog:Domain[T.Vision]               0.9746      0.271      3.599      0.000       0.444       1.505
==================================================================================================
```

The model shows that for every 1 percent increase in the amount of parameters, there is an associated increase of 1.0218 percent in the petaFLOPs required. This happens less than 1 percent of the time under the default null hypothesis of there being no difference between the models. Overall, models that fall under the “vision” domain tend to consume the most petaflops when scaling, as a 1% increase in parameter count results in the petaFLOPs required going up by .9746%, holding all else constant.

Conclusion

We can conclude that different model domains will require differing levels of computational increase per additional parameters. Models that fall under the vision domain see the sharpest rise in requirement per 1% increase of parameters, at about .9746% percent, holding all else constant.

References

- https://epoch.ai/data/ai-models#explore-the-data
- https://ourworldindata.org/artificial-intelligence