**Writing a fanfiction writer’s guide using Linear Regression**

By Sean Flanagan

**I. Abstract** – The goal of this project was to identify key items of metadata from fanfiction posts that could reasonably predict the number of views those posts would get in the first week after being posted. This was primarily motivated by the online fanfiction site Archive of Our Own and its content moderation policies, which are practically nonexistent. AO3, as its known, has quite a robust technical FAQ and helpful explanations of its policies. But it has adopted a laissez-faire attitude towards what kinds of work gets posted on its site, and places almost no restrictions on user-generated content. With that in mind, I was interested in identifying some best practices for posting work on the site, with new authors/posters trying to build an audience at the front of my mind. This might be a way for AO3 to do at least a little quality assurance without actually moderating content. Assuming that the most authors posting to the site would consider their work successful if more people read it, my project can be summarized as: **What characteristics best predict a high pageview count?**

**II. Design** – The project took two parts – a data collection phase using webscraping tools, and a linear regression phase, in which I used the data had scraped to try to accurately predict pageviews for recent works posted to the site. All the data was scraped a week after it was posted so that I was fairly comparing works in the same timeframe.

**III. Data** – The data consists of ~1,000 posts to the site, along with the metadata associated with each post. This includes word count, “hits” (our target variable), chapter count, tag counts, content warnings, and ratings. Additionally, I calculated sentiment scores based on the actual text of the posts using a Python package.

**IV. Algorithms**

**-** Webscraping the data through iteration, including error handling

- Data cleaning by ensuring the data collected was of the proper type (i.e. integer data was not mistakenly converted to strings)

- Dummy variable creation for additional variables

**V. Tools**

**-** Python BeautifulSoup for webscraping and HTML parsing

- Python Pandas and NumPy for data cleaning and manipulation

- Matplotlib/Seaborn/Yellowbrick for plotting

**VI. Communication**

In addition to the visuals and slides, this project will expand on my [personal GitHub page.](https://github.com/SeanMartinFlanagan/EDA-Project)