Project: Give Me The Next AAA Title

A Deep Learning Approach

Jose Aries E. De Los Santos

Contents

Introduction

IMDb: Internet Movie Database

What is IMDb?

- IMDb (Internet Movie Database) is an online database of information related to films, television programs, home videos, video games, and streaming content.
- It includes details such as cast, production crew, plot summaries, trivia, ratings, and reviews.
- Founded in 1990 by Col Needham, it is now owned by Amazon.
- IMDb is widely used by audiences and industry professionals for research, entertainment, and decision-making.
- The platform also hosts user-generated content, including ratings and reviews.

MAIN Objective: Build a deep learning model that will help us predict the next AAA title.

Overview of the Datasets

Key IMDb Datasets

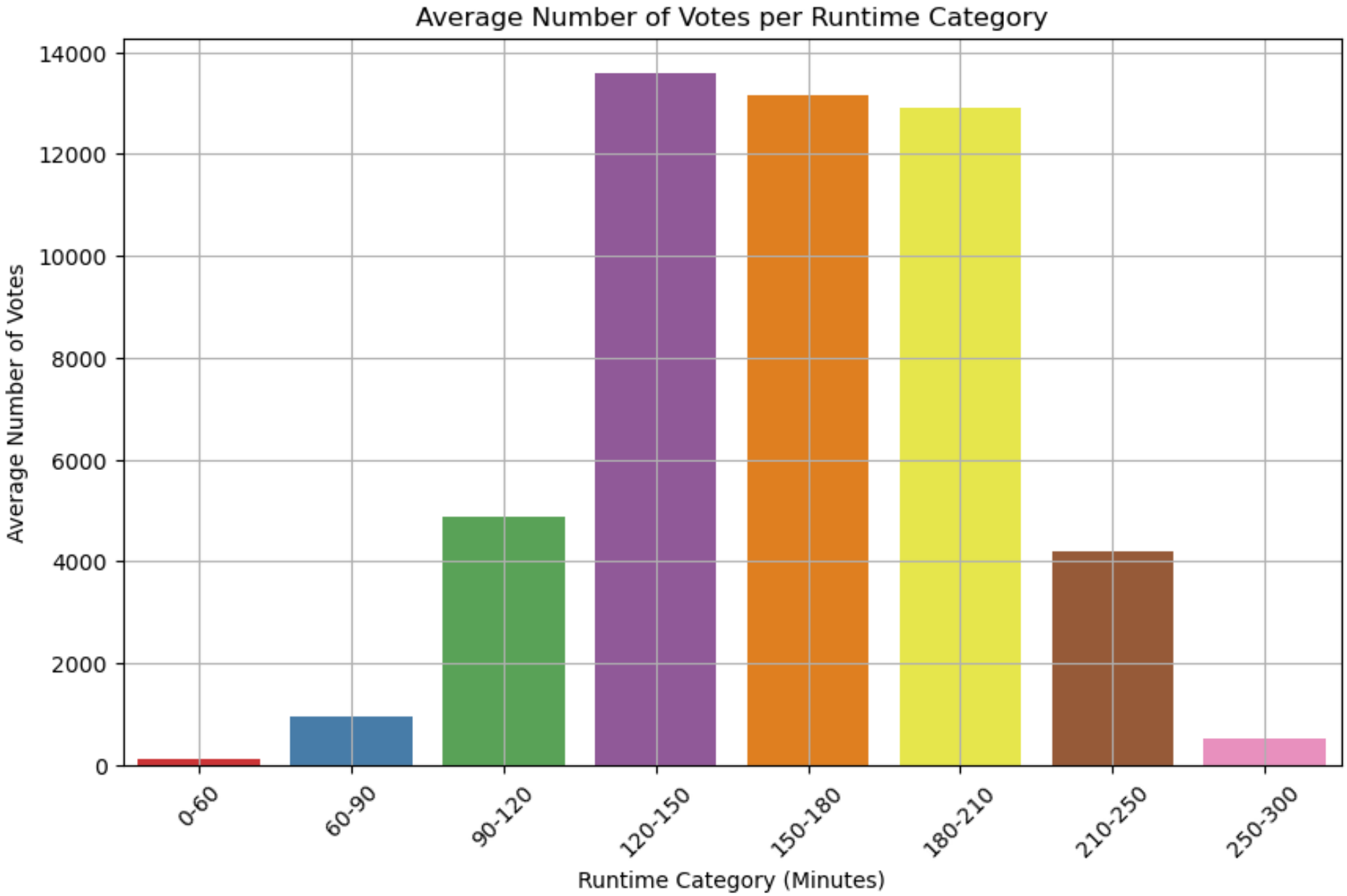
- **title.basics**: Provides basic details about titles, including title name, genres, runtime, and release year.
- **title.ratings**: Contains IMDb rating information, such as average rating and number of votes for each title.
- **title.crew**: Lists directors and writers associated with each title.
- **title.principals**: Includes the main cast members for each title.
- **names.basics**: Stores basic information about individuals, such as their name, birth year, death year, and the titles they are most known for.
- **title.akas**: Provides details about alternate versions, languages, and regional releases of titles.

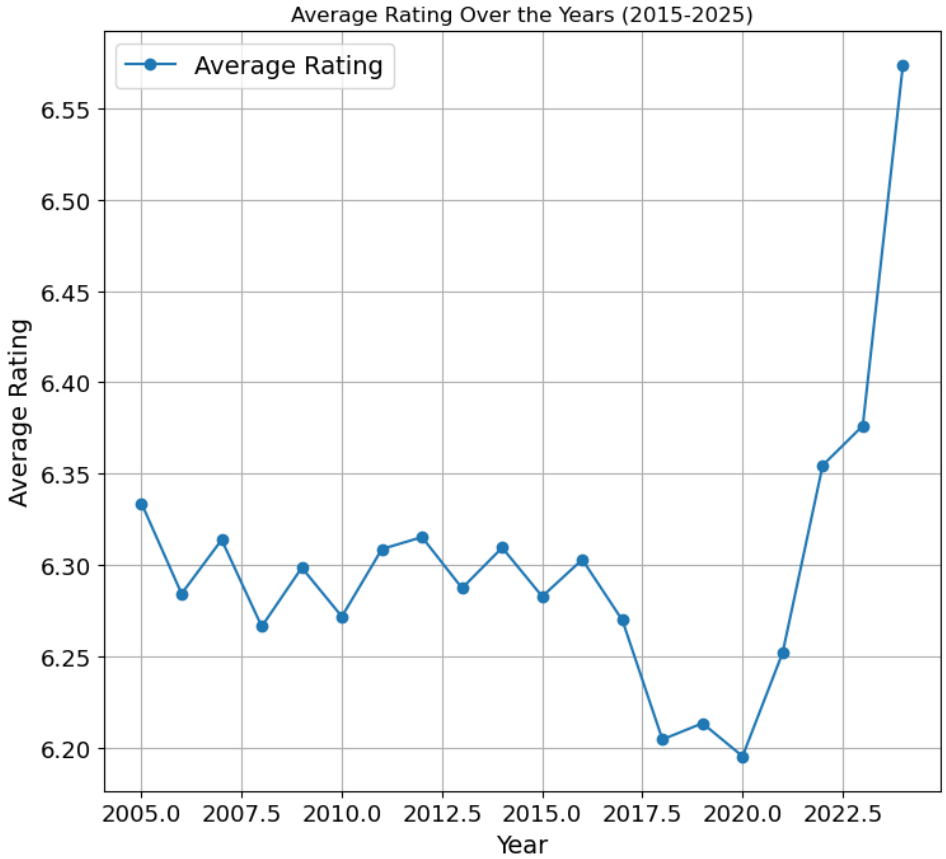
The IMDb datasets are publicly available and can be accessed at:

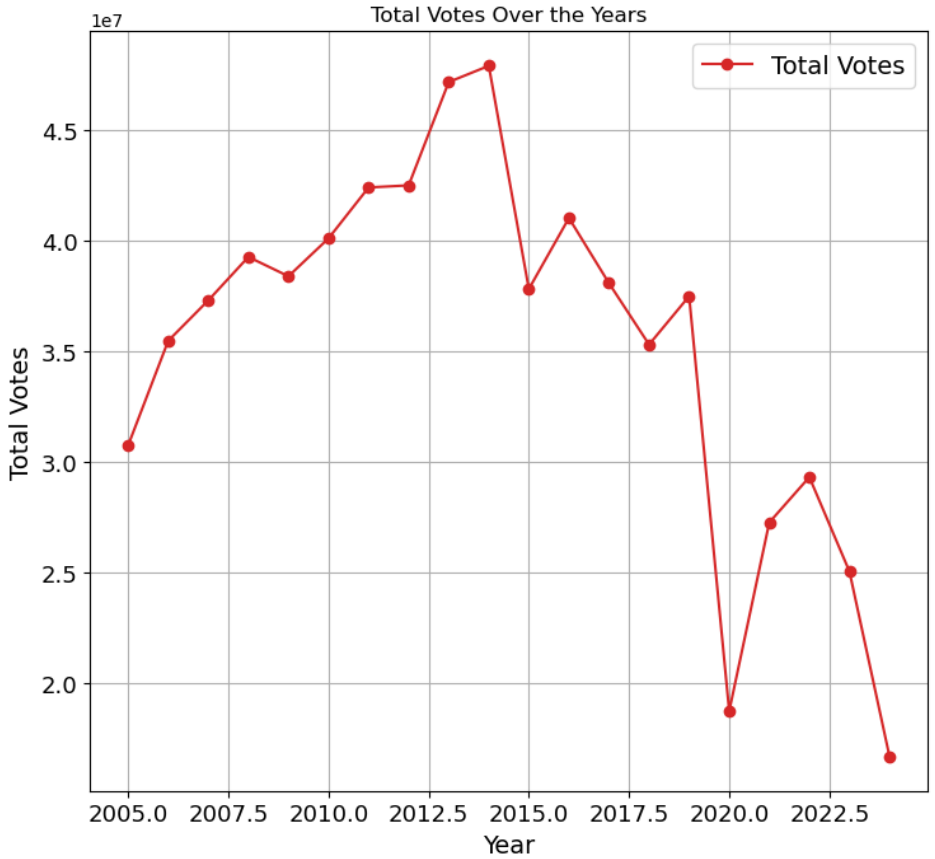
https://datasets.imdbws.com/

Some things the Data Tells Us

Note: *The dataset is filtered to movies only*







Key Takeaways

- Modern movies are rated more favorably on average, but they don't receive as many votes as older movies did.
- This suggests a shift in audience engagement—perhaps niche, highly-rated movies dominate while mass-voted blockbusters decline.

Weighted Rating

The weighted rating formula is given by:

$$\textbf{Weighted Rating} = \frac{v}{v+m}R + \frac{m}{v+m}C \tag{1}$$

where:

$$\begin{aligned} R &= \text{average rating of the movie,} \\ v &= \text{number of votes for the movie,} \\ m &= \text{minimum votes required for consideration,} \\ C &= \text{mean rating across all qualifying movies.} \end{aligned}$$

- To determine the best movie in 2023 we use the preceding formula
- Filter the movies for the year 2023

Intuition

Bayes' Theorem:

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(\mathbb{B}|\mathbb{A}) \cdot \mathbb{P}(\mathbb{A})}{\mathbb{P}(B)}$$

- $\mathbb{P}(A)$ is prior belief about event A
- $\mathbb{P}(B|A)$ is the posterior belief about A after observing B
- $\mathbb{P}(\mathbb{A}|\mathbb{B}$ posterior belief about A after observing B

In the Weighted Rating Formula

- C corresponds to the prior
- R correspond to the likelihood ($\mathbb{P}(B|A)$)
- the Weighted rating corresponds to the posterior

Spider-Man: Across the Spider-Verse

Weighted Rating: 8.3229
Average Rating: 8.5
Number of Votes: 439,720

Most Popular actor 2023

$$\text{Popularity Score} = w_1 \mu_{aveRating} + w_2 m_{aveRating} - w_3 \sigma_{aveRating}$$

where

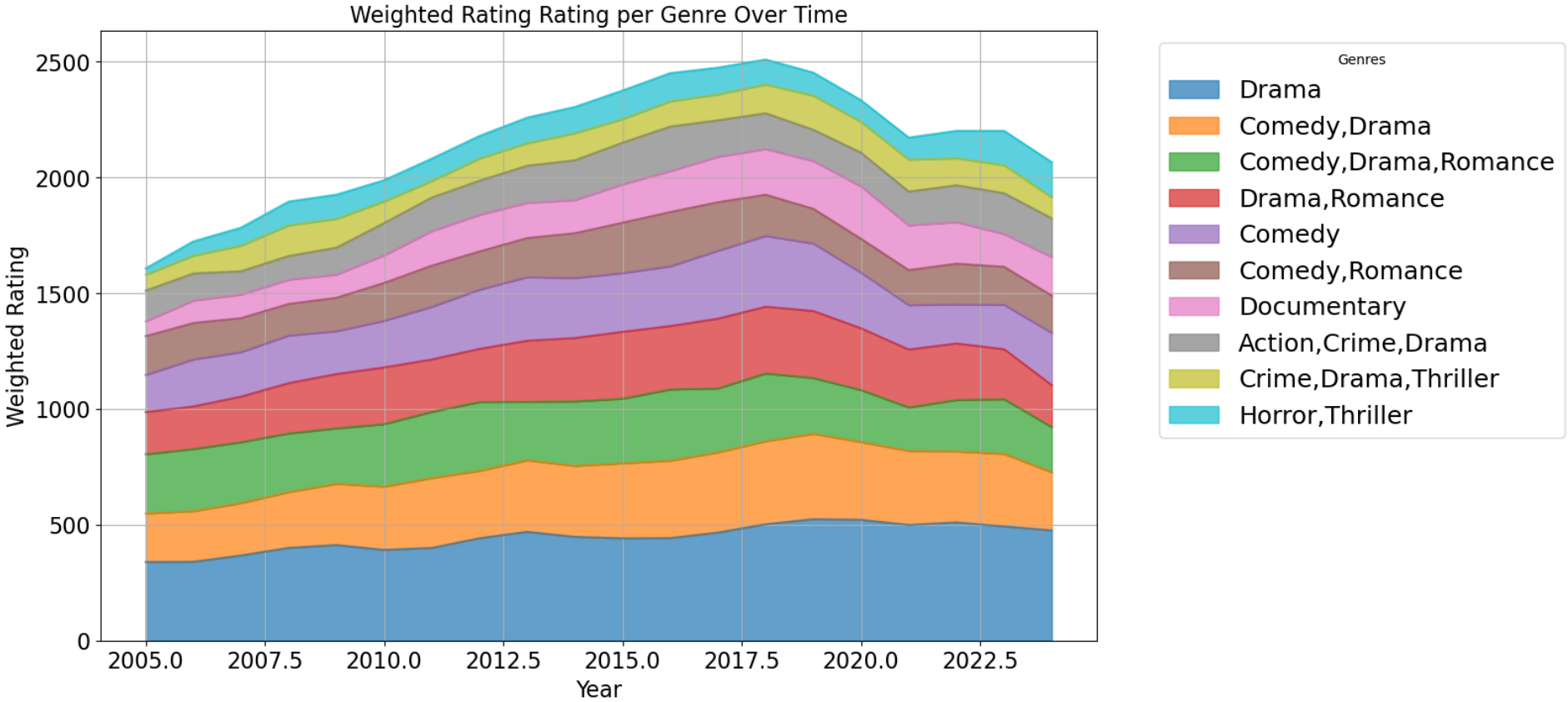
- $w_1, w_2, w3$ are the weights we assigned
- $\mu_{aveRating}$ mean of the list of average ratings of the movies the actor is involved likewise with the median $m_{aveRating}$ and standard deviation $\sigma_{aveRating}$

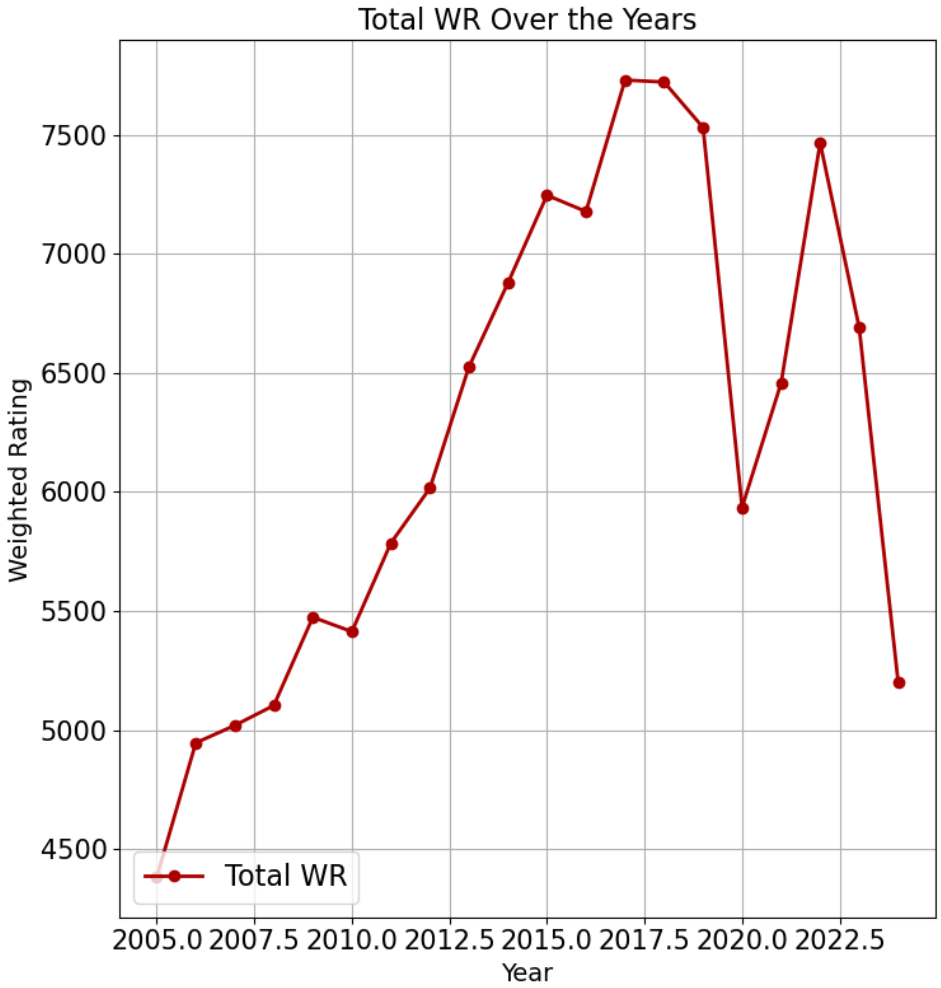
Intuition

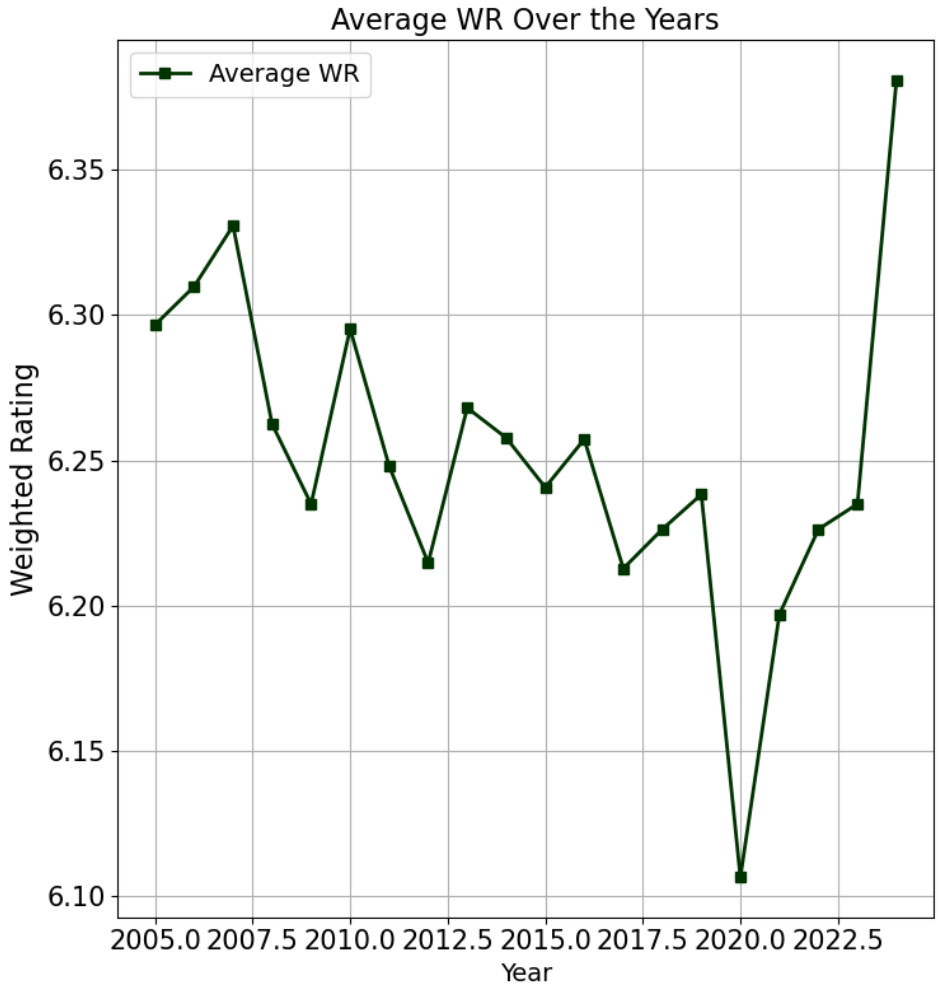
- The **mean** of an actor's ratings provides an idea of their overall popularity.
- The **median** of the ratings captures the general consensus on the actor's performance.
- The **standard deviation** characterizes the variability in the ratings:
  - A **high mean** with a **low standard deviation** indicates consistent excellence in performance.

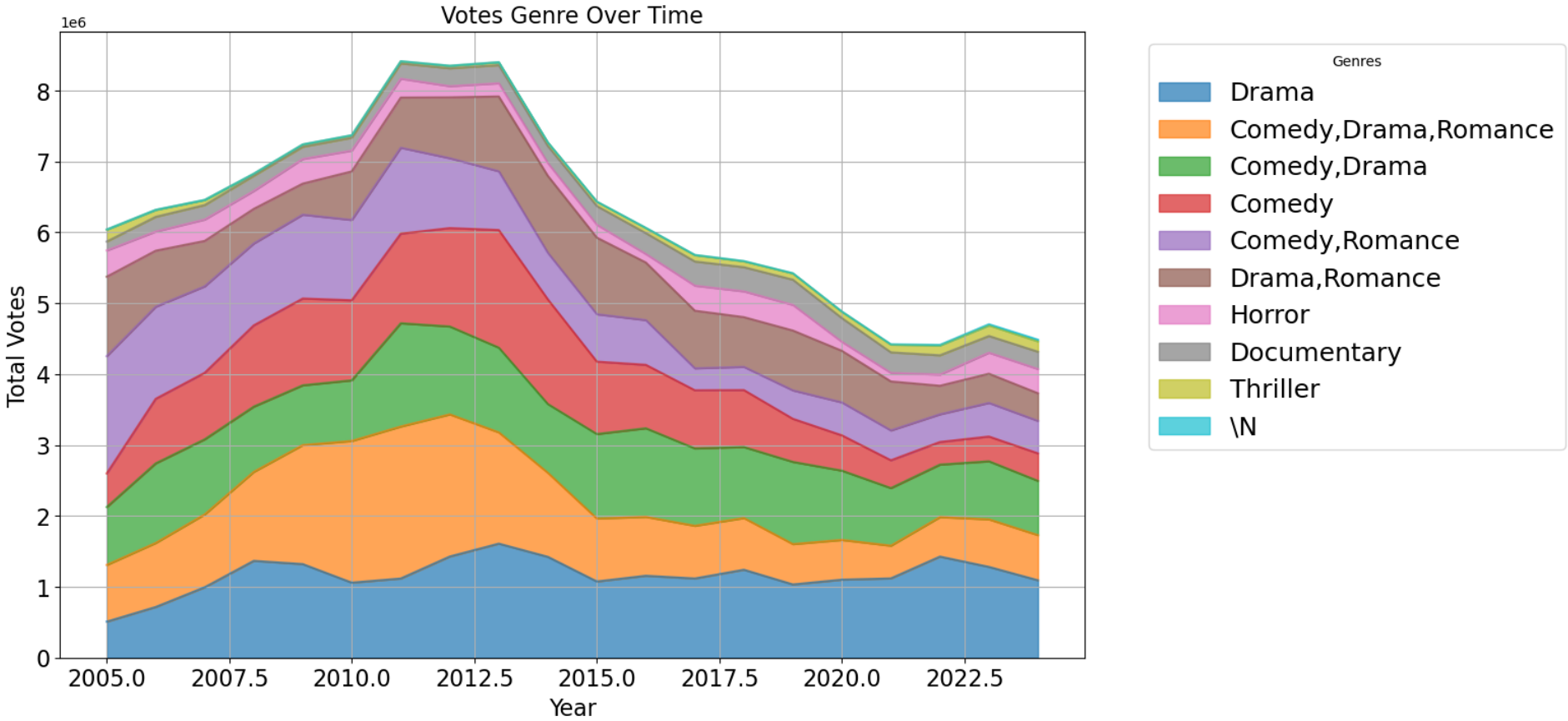
Name: Chris Patt

User Preferences Trend Over the Years




Total WR Over the Years
Weighted Rating
Year
Total WR
7500
7000
6500
6000
5500
5000
4500
2005.0 2007.5 2010.0 2012.5 2015.0 2017.5 2020.0 2022.5


Average WR Over the Years
Weighted Rating
Year
Average WR
6.35
6.30
6.25
6.20
6.15
6.10
2005.0 2007.5 2010.0 2012.5 2015.0 2017.5 2020.0 2022.5


Votes Genre Over Time
1e6
Total Votes
Year
0
1
2
3
4
5
6
7
8
2005.0
2007.5
2010.0
2012.5
2015.0
2017.5
2020.0
2022.5
Genres
Drama
Comedy,Drama,Romance
Comedy,Drama
Comedy
Comedy,Romance
Drama,Romance
Horror
Documentary
Thriller
\N

Key Takeaways

- **Average Rating (Mean) is Increasing** → Viewers tend to give higher ratings over time.
- **Mean Weighted Rating (WR) is Rising** → Movies that accumulate enough votes generally receive higher WR scores.
- **Total WR (Sum) is Declining** → The overall impact of ratings is decreasing over the years.
- **Number of Votes is Decreasing** → Fewer users are actively rating movies, indicating lower engagement.
- **Genre Preferences are Shifting** → Drama and Comedy remain popular, while genres like Thriller and Horror show fluctuations.

Hit Score

$$\text{Hit Score} = \alpha \cdot \text{Average Rating} + \beta \cdot \log(1 + \text{NumVotes}) \cdot e^{-\lambda(endYear - startYear)}$$

where:

- **Average Rating** – Represents the perceived quality of the movie.
- **NumVotes (log-scaled)** – Ensures popular movies receive higher weight without extreme dominance.
- α **and** β – Weighting factors that balance rating and popularity.
- λ – The rate of decay.

Methodology

Universal Approximation Theorem (G. Cybenko, 1989

) Let σ by any continuous sigmoidal function. Then finite sums of the form

$$G(x) = \sum_{j=1}^{N} \alpha_j \sigma(w_j x + \theta_j)$$

are dense in $C(I_n)$. In other words, given any $f \in C(I_n)$ and ε, there is a sum $G(x)$ of the above form, for which

$$|G(x) - f(x)| < \varepsilon \text{ for all } x \in I_n.$$

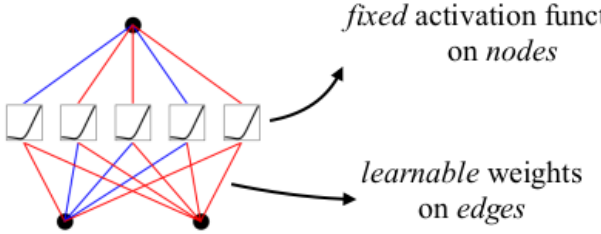
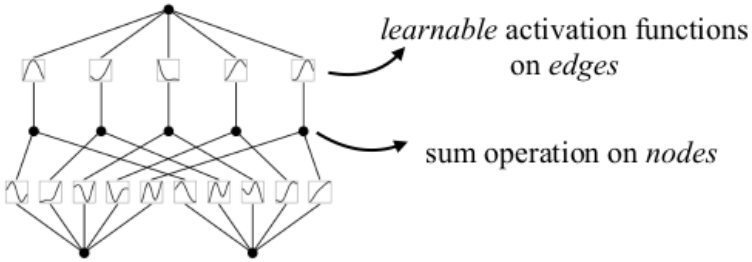
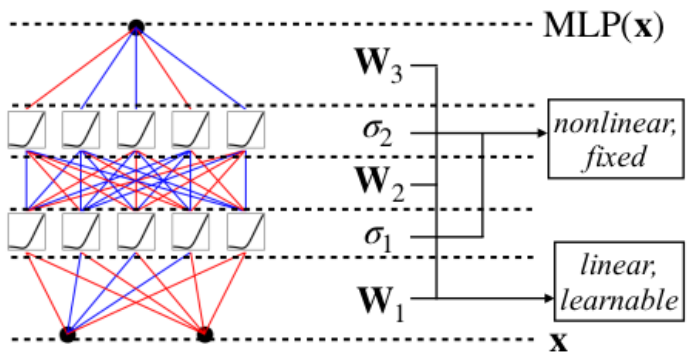
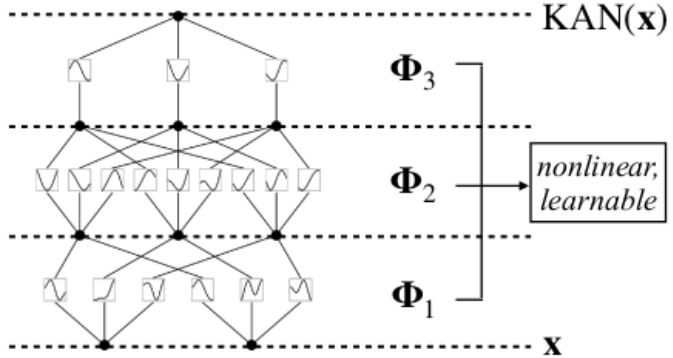
A set of functions $\mathcal{F}$ is **dense** in a function space $C(I_n)$ if, for every function $f \in C(I_n)$ and for every arbitrarily small error $\varepsilon > 0$, there exists a function $G \in \mathcal{F}$ such that $\|G - f\|_\infty = \sup_{x \in I_n} |G(x) - f(x)| < \varepsilon$

Kolgomorov Arnold Representation Theorem

If f is a multivariate continuous function on a bounded domain, then it can be written as a finite composition of continuous functions of a single variable and the binary operation of addition. More specifically, for a smooth $f : [0, 1]^n \to \mathbb{R}$,

$$f(\mathbf{x}) = f(x_1, ..., x_n) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right)$$

where $\phi_{q,p}(x_p) \to \mathbb{R}$ and $\Phi_q : \mathbb{R} \to \mathbb{R}$.

Model	**Multi-Layer Perceptron (MLP)**	**Kolmogorov-Arnold Network (KAN)**
Theorem	**Universal Approximation Theorem**	**Kolmogorov-Arnold Representation Theorem**
Formula (Shallow)	$f(\mathbf{x}) \approx \sum_{i=1}^{N(\epsilon)} a_i \sigma(\mathbf{w}_i \cdot \mathbf{x} + b_i)$	$f(\mathbf{x}) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right)$
Model (Shallow)	**(a)** 	**(b)** 
Formula (Deep)	$\text{MLP}(\mathbf{x}) = (\mathbf{W}_3 \circ \sigma_2 \circ \mathbf{W}_2 \circ \sigma_1 \circ \mathbf{W}_1)(\mathbf{x})$	$\text{KAN}(\mathbf{x}) = (\mathbf{\Phi}_3 \circ \mathbf{\Phi}_2 \circ \mathbf{\Phi}_1)(\mathbf{x})$
Model (Deep)	**(c)** 	**(d)** 

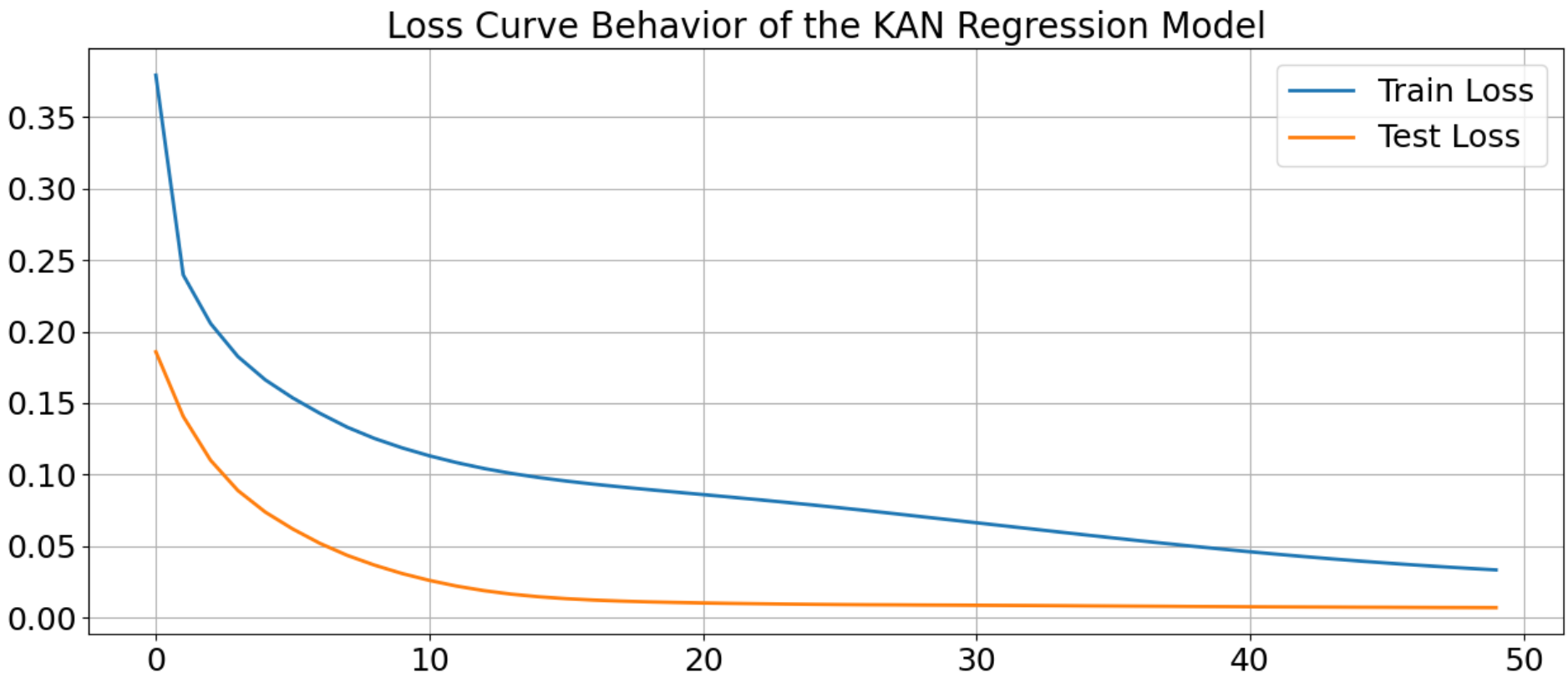
Data Preprocessing for MLP and KAN

Baseline Model: Multilayer Perceptron
Interpretable Model: Kolgomorov Arnold Network

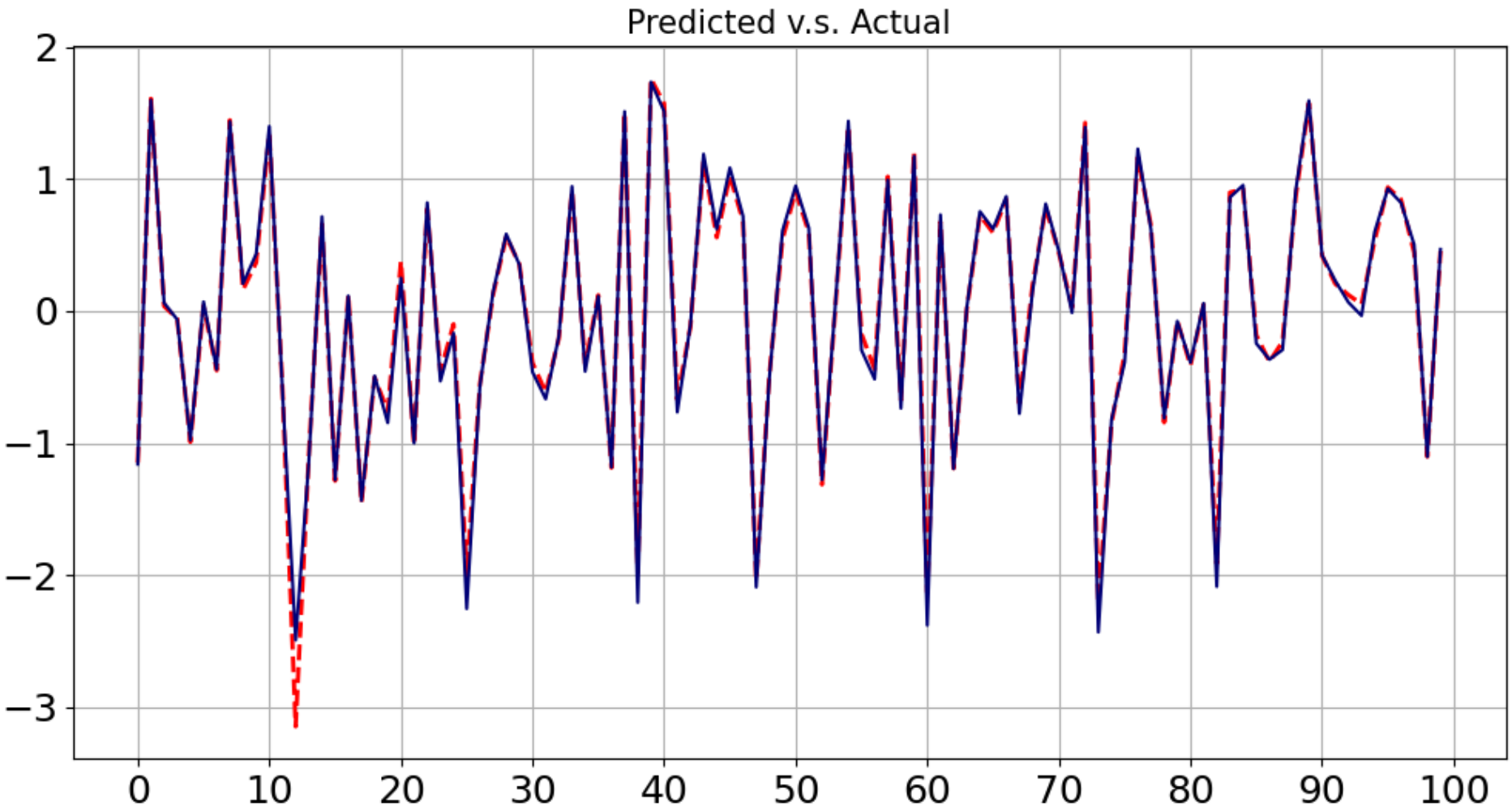
	averageRating	log(1+numVotes)	Genre 1	Genre 2	Genre 3	Genre 4	Genre 5	HitScore
Movie A			1	1	0	0	0	
Movie B			1	0	0	0	1	
Movie C			0	0	1	0	1	

- One hot encoded Genres: Some genres might contribute to success more than others, and one-hot encoding allows models (especially KANs and MLPs) to learn these effects independently.
- GOAL: Use deep learning to identify the relationship of the features: averageRating, numVotes, and genres to the HitScore.
- We expect that the with this approach may be able to predict the hit scores of future movies.

Initial Results: Loss Curve Behavior



Initial Results: Predicted vs Actual on the Test Dataset



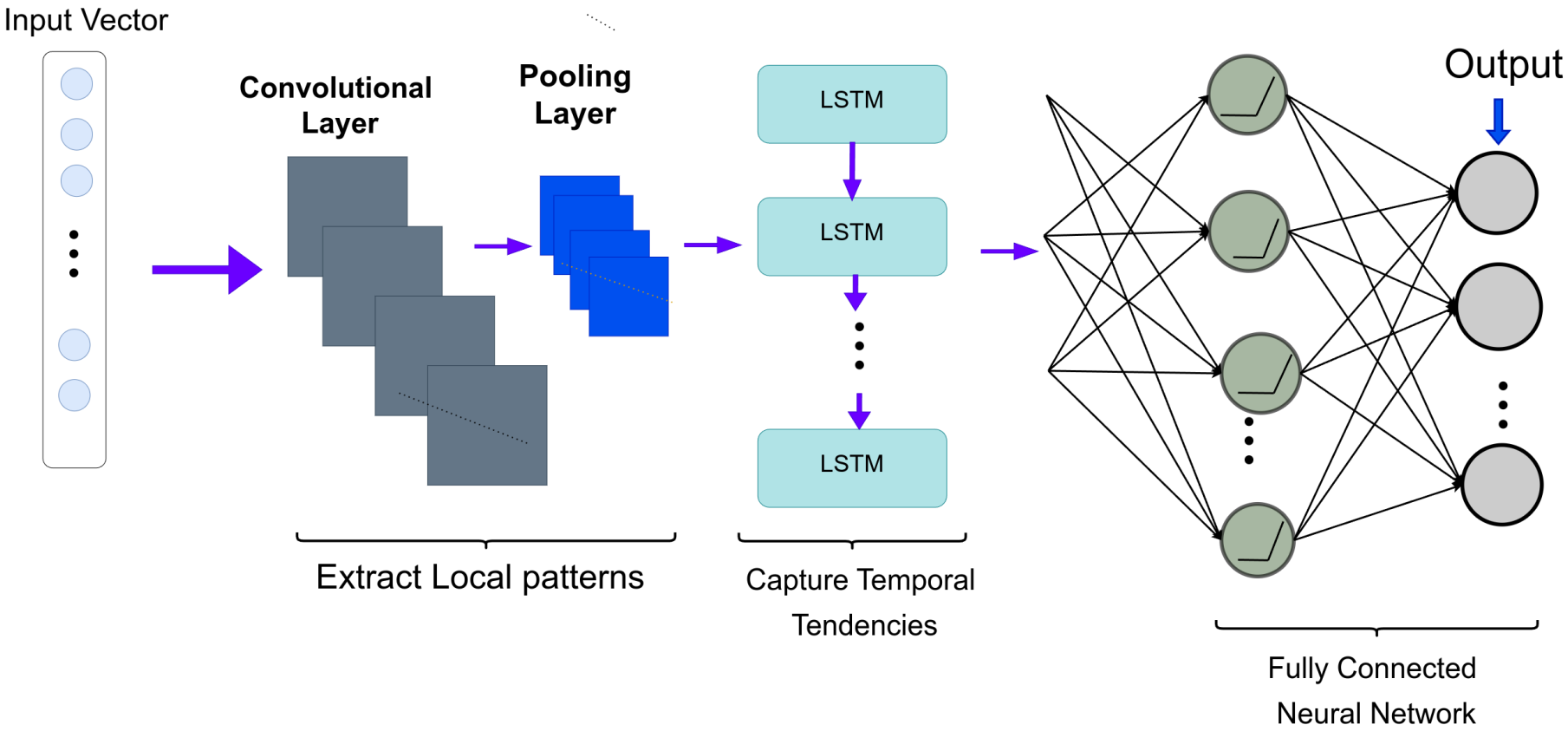
Conclusion

Conclusion and Suggestion

- MLPs are used as a baseline model to identify initial parameters for manual tuning
- KANs was able to generalize on the unseen data at a certain accuracy
- We can further engineer the KAN model, to identify the what functions it has learned. This allows us to understand further the relationship between the chosen features and the target in the dataset.

Future Suggestion

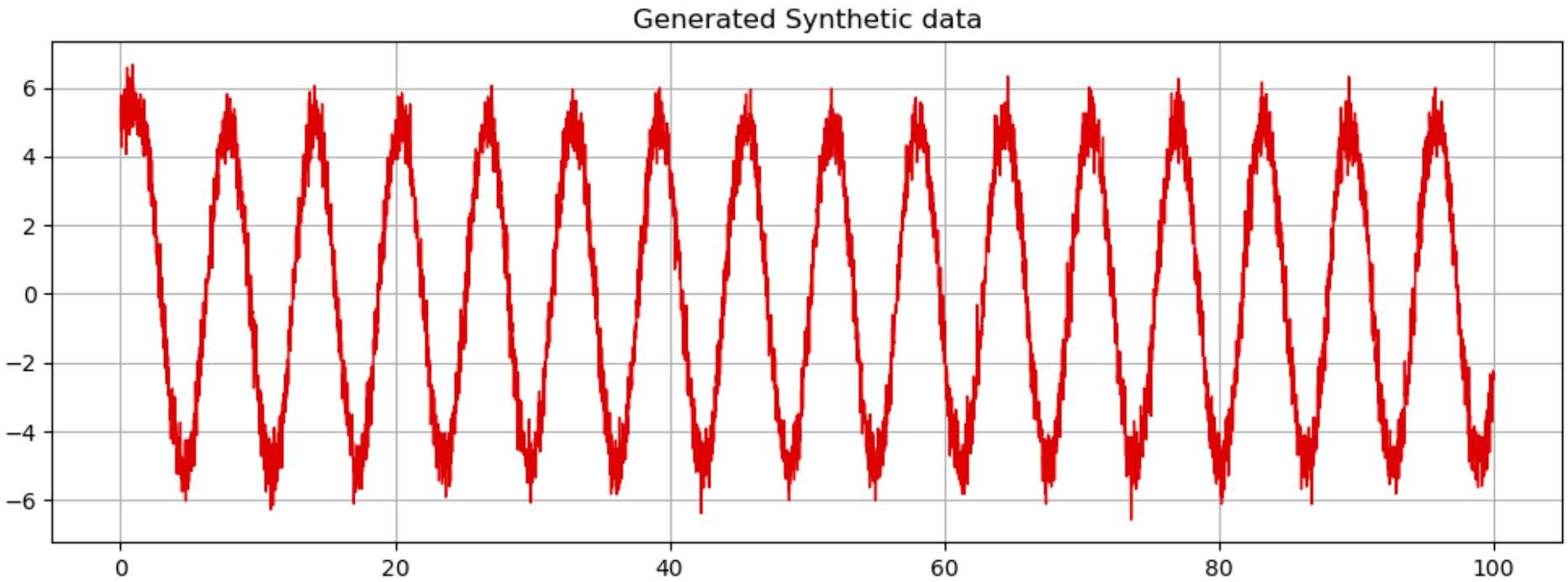
Intuition: CNN-LSTM

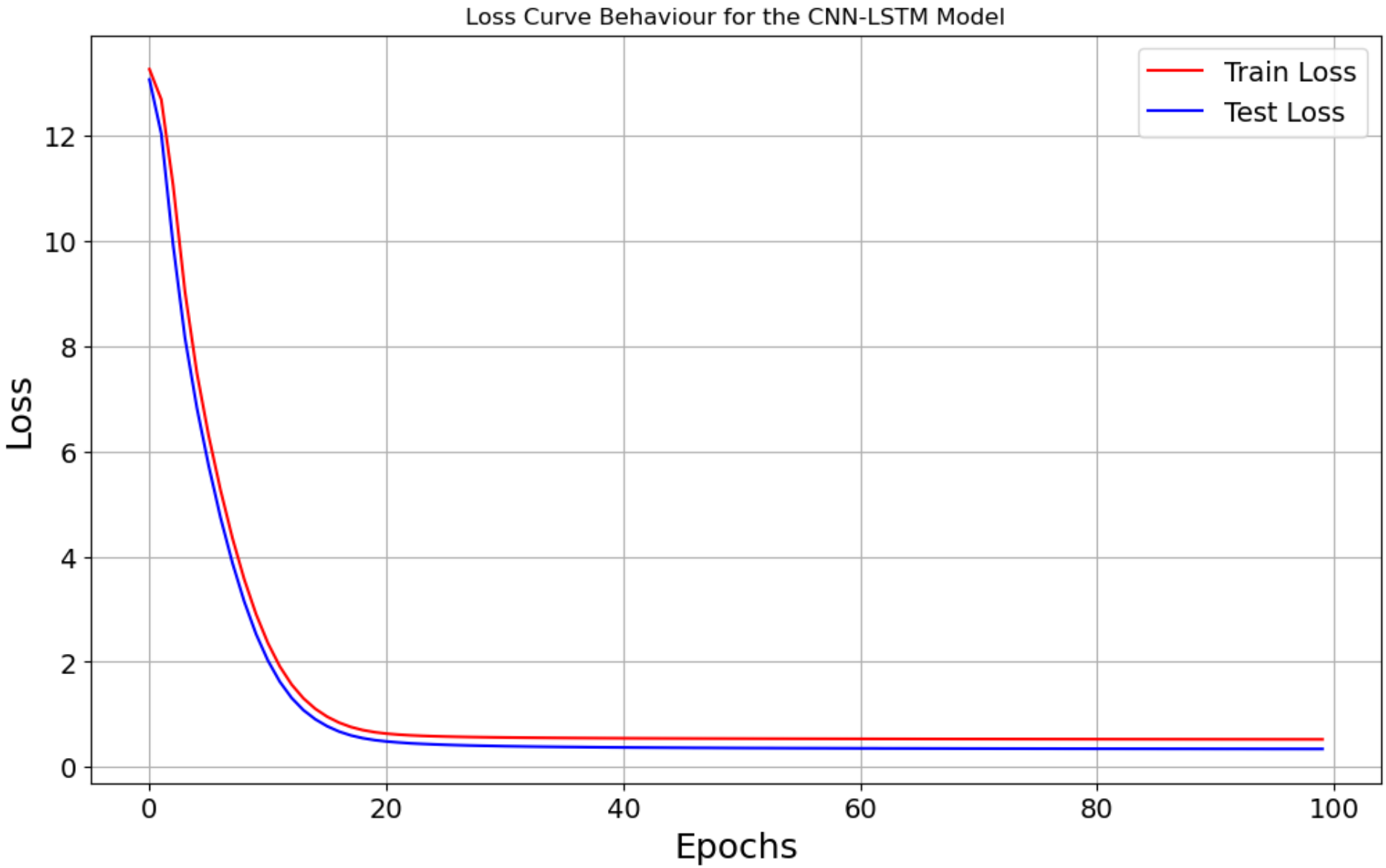


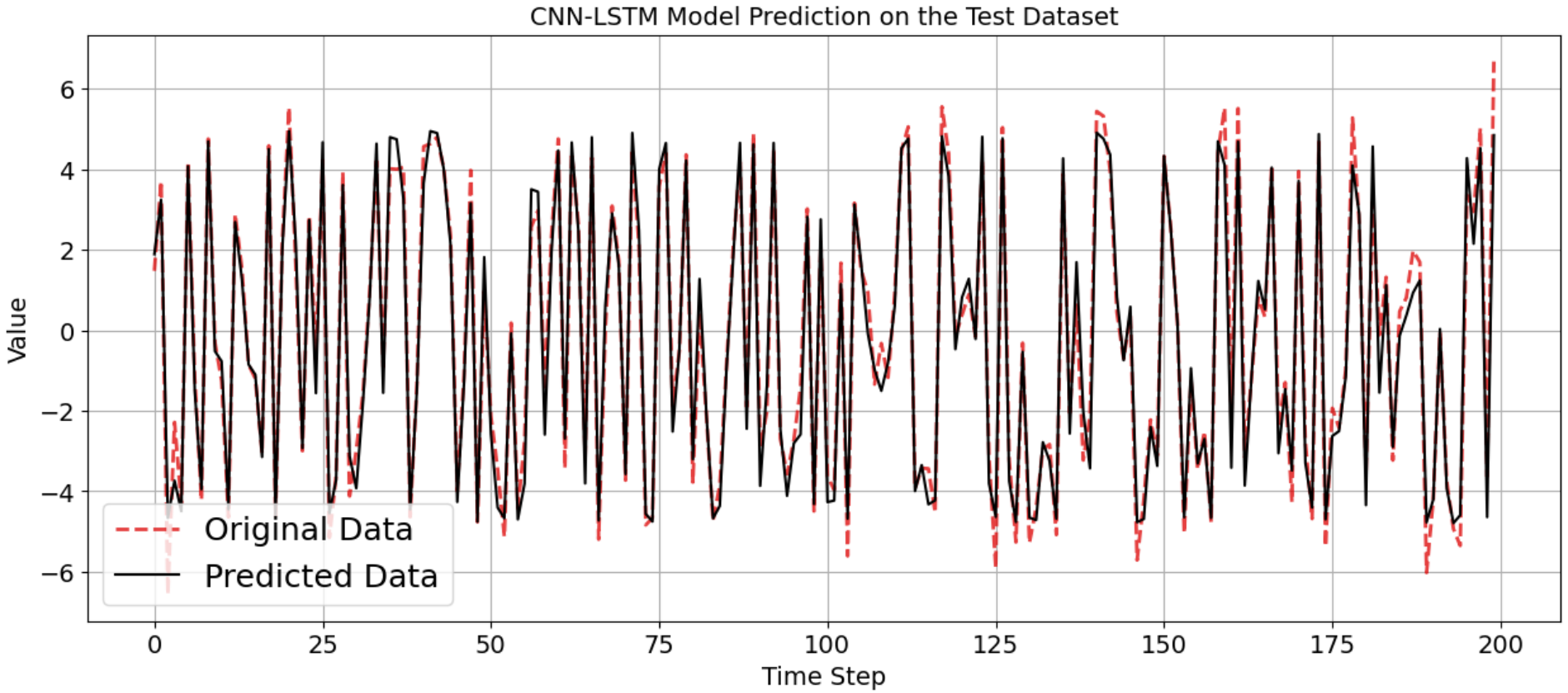
Initial Results

Synthetic Noisy Dataset

$$\text{Noisy Data} = 5\sin(t) + 0.5\mathcal{N}(0, \mathbf{I}_n) + 4.5\cos(t)e^{-0.5t}$$




Loss Curve Behaviour for the CNN-LSTM Model
Train Loss
Test Loss
Loss
Epochs
0
2
4
6
8
10
12
0
20
40
60
80
100


CNN-LSTM Model Prediction on the Test Dataset
Value
Time Step
Original Data
Predicted Data
6
4
2
0
−2
−4
−6
0
25
50
75
100
125
150
175
200

References

- Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., Hou, T. Y., & Tegmark, M. (2025). KAN: Kolmogorov-Arnold Networks.
- Hussain, S., Aziz, A., Hossen, M., Aziz, N., Murthy, G., & Mustakim, F. (2022). A novel framework based on CNN-LSTM neural network for prediction of missing values in electricity consumption time-series datasets. Journal of Information Processing Systems, 1(1), 115-129.
- Lu, B.-Y., Li, J., Chen, Y.-Z., & Xu, H. (2017). Evaluation of the television dramas ranking using the Bayes' theorem. Advances in Social Science, Education and Humanities Research, 90. 3rd Annual International Conference on Social Science and Contemporary Humanity Development (SSCHD 2017).
- Cybenko, G. V. (1989). Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals and Systems, 2, 303–314.
- Bishop, C. M. (2006). Pattern recognition and machine learning (Information Science and Statistics)*. Springer-Verlag.