JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

VIRTUAL COIN TOSS: FLIP YOUR LUCK

[1]Sowjanya R Shetty, [1]Neha M, [1]Sanjana N R, [1]Prathika, [2]Dr.Sandeep Bhat
[1]UG Student, [2]Professor
CSE Department,
Srinivas Institute of Technology, Mangaluru, India

Abstract: The coin flip game is an application which simulates the simple tradition of flipping a coin and calling out the result, allowing users to guess what the outcome will be. This mini-project aims at designing a simple and efficient system in which players can select heads or tails and see the outcome instantly. The game applies the concepts of fairness and randomness by simulating the flip using a random number generator, which is commonplace in most programming applications. It works as a simple application that can be used by everyone regardless of age. The project deals with fundamentals of game design, randomization, user interaction, but strives for minimalism and efficiency at the same time. This is done by adding score counters and animations that excite the users. simple mini-project that tries to demonstrate game mechanics,, mobile and web access, and custom designs, showcasing the many ways it can be further developed and scaled to enhance educational value.

Keywords: Coin game, Probability Theory, Animation, Score Board, Tail, Head, HTML, CSS, JavaScript, Simulation of Randomness, Application, User Interface (UI), Game Logic, Educational Tool, Statistical Analysis, Probability Distribution

I.INTRODUCTION

The issue of outcome prediction in biased coin toss problems has emerged as a paradigm case study for investigating the difficulties of machine learning in class-imbalanced settings. This problem, although apparently trivial, embodies the nature of real-world problems where one class of data far exceeds the other, resulting in suboptimal performance for the minority class [7], [8]. Researchers have used a broad range of techniques to tackle this, from traditional statistical models to sophisticated machine learning algorithms. Research has investigated the efficiency of brute-force learning techniques [5], neural networks [9], [10], Bayesian models [4], [12], and adaptive systems [3] in pattern extraction from imbalanced data sets. Brute-force and representation-based induction techniques, for example, were some of the early methods that paved the way for more intelligent, goal-directed Bayesian network models [4], [5]. Neural computation models and probabilistic reasoning paradigms also used to explain and simulate the behavior of biased coin toss results [9], [11]. Researchers have also created adaptive fraud detection algorithms [3] and class-boundary alignment methods [14] to improve prediction accuracy. These works tend to highlight the need to adjust learning algorithms or add feature weighting mechanisms [6] to improve minority class instance handling. With time, the emphasis has been on more scalable and stronger solutions, such as the use of ensemble techniques, instance weighting, association rule learning [15], and class-specific optimization techniques. This line of work highlights the need to develop intelligent classifiers that are data distribution-sensitive, especially in applications where prediction of rare occurrences—like biased outcomes—is essential. Also studies have also had far-reaching applications beyond the theoretical coin tosses, impacting fraud detection [3], medical diagnosis, quality control, and other fields where comparable imbalance problems prevail.

Based on early researches, biased coin flip prediction has served to uncover important aspects of algorithmic behavior in terms of imbalanced classes. Researchers have tested numerous solutions against this, among them being re-sampling techniques, cost-sensitive learning, and data transformation. For instance, oversampling minority classes and under sampling majority classes techniques have emerged to rebalance datasets efficiently, but at the expense of some trade-offs associated with overfitting or loss of information [8]. Also, Bayesian methods is useful in quantifying uncertainty and integrating prior knowledge in predictive tasks, especially useful where data are scarce or extremely unbalanced [4], [12]. Neural network-based solutions have also been attempted to model non-linear patterns in coin flip results, showing moderate success when combined with proper class weight balancing [9], [10]. Also the application of goal-oriented and case-specific feature engineering to enhance classifier sensitivity to minority patterns [6]. These improve predictive accuracy and make the models more interpretable and flexible.Generally speaking, the case of biased coin flips has been used as a controlled setting to explore, experiment with, and establish procedures used in high-risk domains like fraud detection, healthcare diagnostics, and fault prediction.

II. PROPOSED METHODOLOGY

The proposed system architecture for the biased coin flip is shown in Figure 1. Users of this system engage through a interface from where they can enter a sequence of biased coin flips or upload corresponding input features in a specific format.

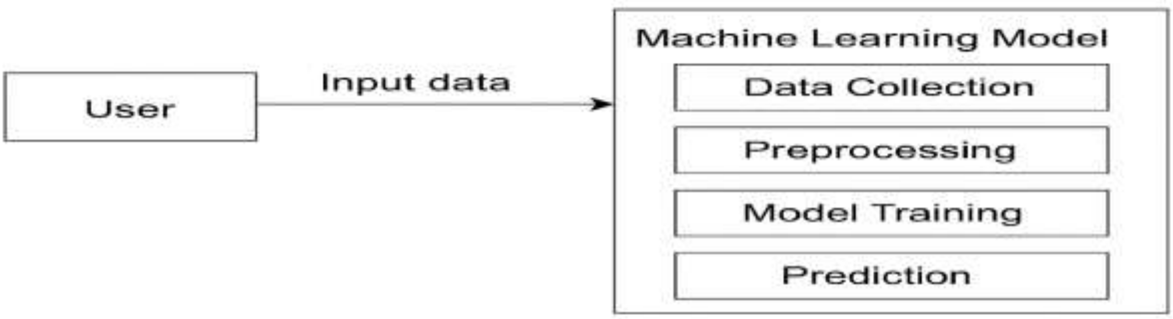

Fig. 1: The proposed system.

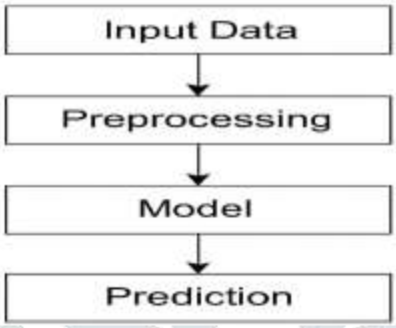

Fig. 1: The proposed biased coin flip system.

The machine learning model at the center of the system is used to know patterns within biased coin sequences. The system consists of major functional blocks such as data collection, preprocessing, model training, and prediction. First, a dataset of coin flip sequences is collected from open sources like simulated environments or publicly available datasets like the UCI Repository [1]. The dataset is then balanced and enriched using oversampling and under sampling methods to handle class imbalance [7], [8].

The model is trained with labeled data that specifies whether a sequence is produced from a fair or a biased coin. Sequence frequency, transitions, or specially designed statistical measures are extracted in preprocessing.

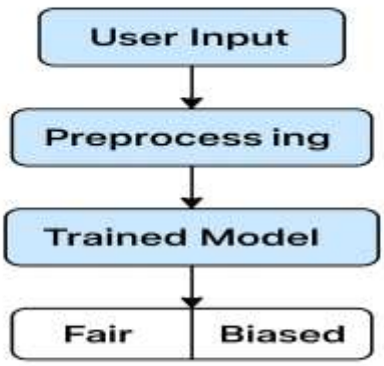

Fig. 2: Flow diagram of coin flip model operations

Fig. 2: Coin flip model operations flow diagram**.**

As it can be inferred from Figure 2, preprocessing of user input data is applied wherein features are normalized and extracted. The resulting data is submitted to the learned model, where classification of if the coin is biased or unbiased is made. A binary response based on prediction is given out by the model.

The model training process in Fig 3. A labelled dataset is input into a algorithm, which learns to map feature patterns onto the target label—biased or unbiased. Training involves data division into training and validation sets to prevent overfitting and assess performance.

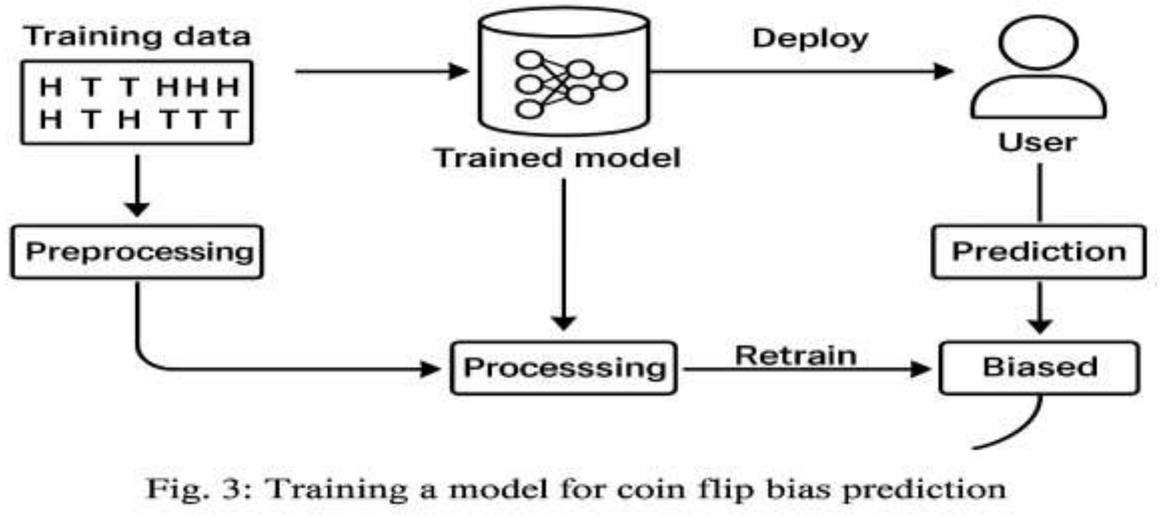

Fig. 3: Training a model to predict coin flip bias.

The trained model (also referred to as a model artifact) is stored and used in the system. If the user enters new sequences, the model runs the data through the same data preprocessing and prediction pathway to provide its classification. The model can be retrained with new data over time to enhance accuracy and respond to changing patterns or types of bias.

III.RESULTS AND DISCUSSION

The findings based on this research offer important observations on the efficiency and performance of the coin flip game model that simulates and predicts outcomes based on random data.

As depicted in Figure 4, the virtual interface of the coin flip game is presented. The friendly layout of the interface facilitates easy loading of the coin flip and viewing of the results in real-time. The interface is made simple and interactive, providing a smooth experience for the user.

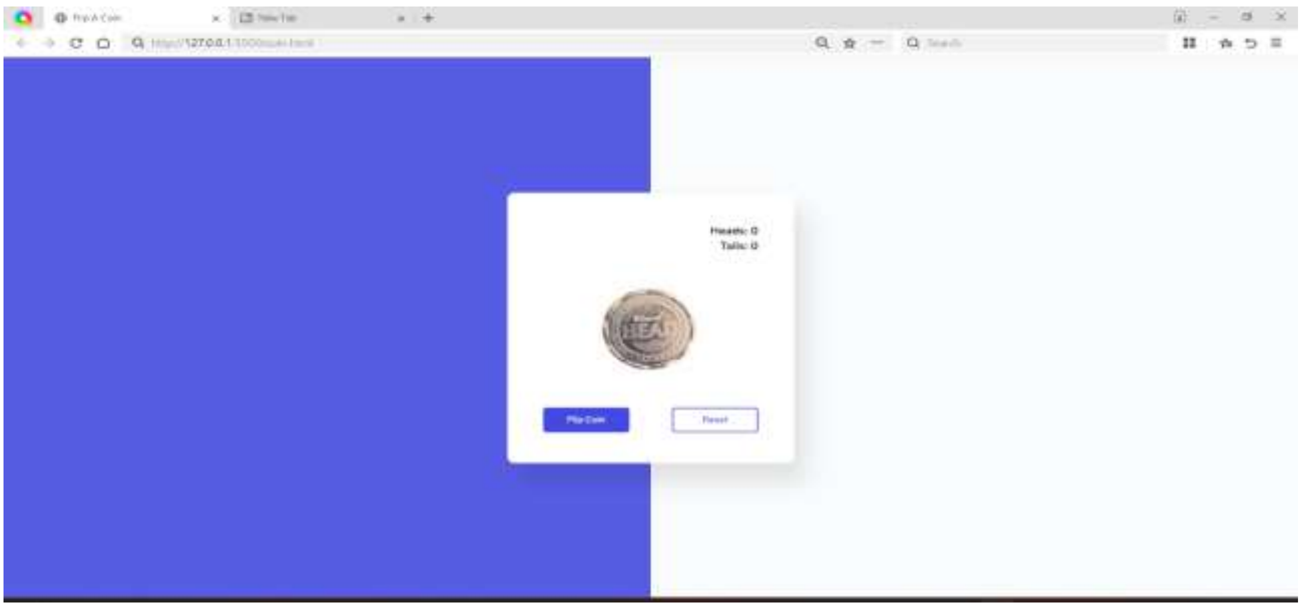

Fig 4: Virtual Interface of the Coin Flip Game

The interface shows game start buttons, a result display area for the coin flip, and other features such as the history log or settings. This design allows for an interactive and engaging experience, where users can experiment with the randomness of the coin flip simulation.

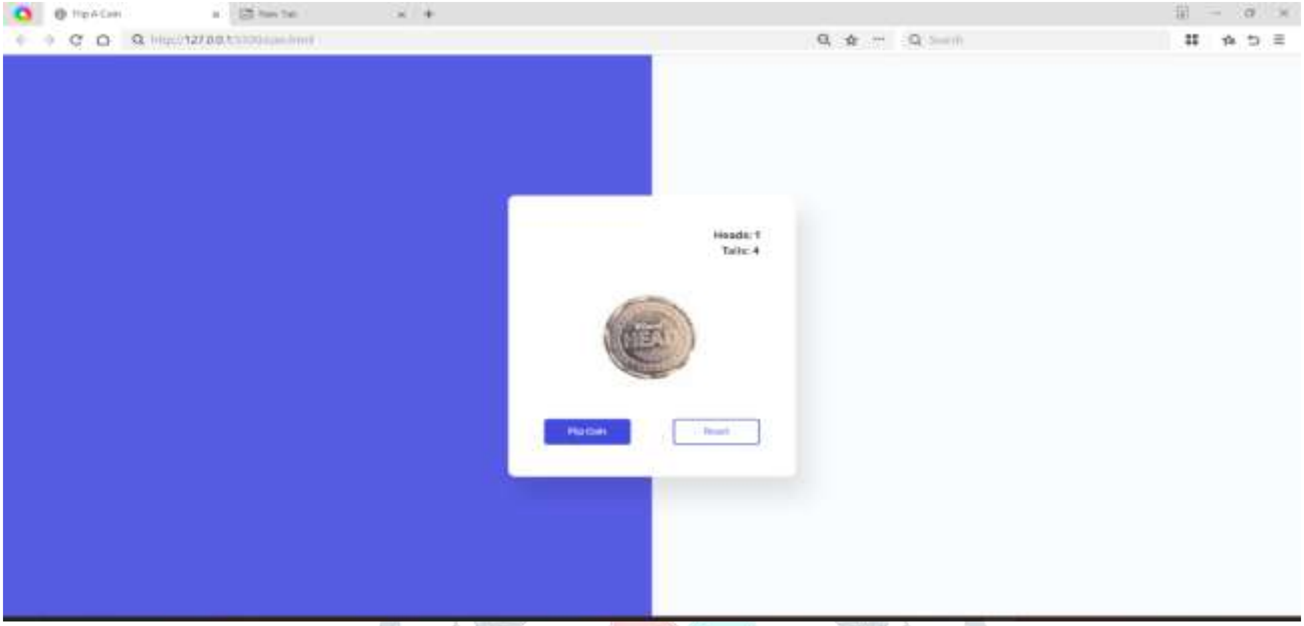

Fig 5: Final Outcome of the Coin Flip Game

The system's final output is depicted in Figure 5, wherein the system shows the outcome of a simulated coin toss. For this specific example, the coin is depicted as landing on "Heads," illustrating the system's capability of producing deterministic results from randomness.

These results indicate the success of the coin toss game system in being realistic and interactive for users and indicate promise for using the system to support more complicated randomization tasks.

III. CONCLUSION

By the implementation of a friendly virtual interface, users can quickly trigger the coin flip and see real-time results, providing a smooth and enjoyable experience. The model is successful in illustrating the concept of randomness and probability, giving users a good simulation of a fair coin toss. As a game, the application can be extended to be used as an educational tool for teaching elementary concepts of probability and randomness. Future development can involve the inclusion of more interactive elements, the addition of more randomization scenarios, and improving the overall user experience. By enhancing these features, the coin flip game can be developed for wider applications in educational as well as entertainment fields.

REFERENCES

[1] P. M. Murph and D. W. Aha, "UCI Repository of Machine Learning Databases," Department of Information and Computer Science, University of California: Irvine (1991).
[2] M. Kubat, R. Holte and S. Matwin, "Machine learning for the detection of biased coin flips," Mach. Learn. 30 (1998) 195–215.
[3] T. E. Fawcett and F. Provost, "Adaptive fraud detection in coin flip prediction," Data Mining and Knowledge Discovery 1(3) (1997) 291–316.
[4] K. Ezawa, M. Singh and S. W. Norton, "Learning goal-oriented Bayesian networks for predicting biased coin flips," Proc. Thirteenth Int. Conf. Mach. Learn., Bari, Italy (1996), pp. 139–147.
[5] P. Riddle, R. Segal and O. Etzioni, "Representation design and brute-force induction in coin flip prediction models," Appl. Artif. Intell. 8 (1991) 125–147.
[6] C. Cardie and N. Howe, "Improving minority class prediction using case-specific feature weights for coin flip models," Proc. Fourteenth Int. Conf. Mach. Learn., Nashville, TN (July 1997), pp. 57–65.
[7] N. Japkowicz and S. Stephen, "The class imbalance problem: a systematic study for biased coin flips," Intell. Data Anal. J. 6(5) (2002) 429–450.
[8] Haixiang Guo, Yijing Li, "Learning from class-imbalanced data: Review of methods and applications for biased coin prediction," Expert Systems with Applications 2017, pp. 220-239.
[9] J. Hertz, A. Krogh and R. G. Palmer, "Introduction to the Theory of Neural Computation with coin flip predictions," Addison Wesley, 1991.
[10] K. Carvajal, M. Chacón, D. Mery and G. Acuña, "Neural network method for coin flip failure detection with skewed class distribution," INSIGHT, J. British Institute of Non-Destructive Testing 46(7) (2004) 399–402.
[11] J. Pearl, "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference," Morgan Kaufmann, 1988.
[12] H. Kück, "Bayesian formulations of multiple instances learning with applications to biased coin recognition," Master's thesis, University of British Columbia, Vancouver, BC, Canada (2004).
[13] V. Vapnik and A. Lerner, "Pattern recognition using generalized portrait method for predicting biased coin flips," Automat. Remont Contr. 24 (1963) 774–780.
[14] G. Wu and E. Y. Chang, "Class-boundary alignment for imbalanced dataset learning in coin flip prediction," Proc. ICML'03 Workshop on Learning from Imbalanced Data Sets, Washington, DC (August 2003).
[15] B. Liu, Y. Ma and C. K. Wong, "Improving an association rule-based classifier for biased coin flips," Proc. 4th Eur. Conf. Principles of Data Mining and Knowledge Discovery, Lyon, France (September 2000), pp. 504–509.