Proceedings of the 2nd International Conference on Mathematical Physics and Computational Simulation
DOI: 10.54254/2753-8818/36/20240593

Applying the Markov chain in natural language processing and three-pool model

Xunyang Wang[1,3], Yueheng Zhang[2]

[1]Beijing City International School, Beijing, 100022, China
[2]Department of Economics, University of Birmingham, Birmingham, B29 7DL, United Kingdom

[3]2022129101@bcis.cn

Abstract. This paper delves into the application of Markov chains in Natural Language Processing (NLP), and the Markov Chain Monte Carlo (MCMC) methodology relevant to the three-pool model. The former outlines the basic principles of Markov chains, highlighting their utility in predicting word sequences in language modelling and text generation, despite certain limitations. Also, the former describes mathematical frameworks like n-gram models that enhance prediction accuracy by considering multiple preceding words. It acknowledges challenges in NLP such as oversimplification and emotional depth, as well as computational issues in higher-order models. It concludes by discussing the integration of Markov chains with other models to mitigate these limitations, and their enduring relevance in computational linguistics. The later investigates the MCMC methodology, a seminal development in the field of statistical inference, which is especially useful when analysing complicated systems when traditional statistical techniques are inadequate. Moreover, this later explores the fundamental concepts of MCMC, clarifies how it is inherently related to Markov chains, presents the three-pool model that is commonly applied to models of physical, chemical, or ecological systems, and discusses how MCMC can be used to analyse these models.

Keywords: Markov chain, Monte Carlo, Natural language processing, Three-pool model.

1. Introduction

Originally introduced in 1906 by Russian mathematician Andrey Markov, Markov Chains are a type of stochastic model that satisfies the Markov Property, also known as the Memoryless property. In less esoteric terms, describe a sequence of events in which the probability of each event depends only on the event directly prior to it. This means that the probability of each future event is independent of all events save for the event directly preceding it [1]. Markov Chains can primarily be divided into two groups, Discrete Time Markov Chains (DTMC) and Continuous Time Markov Chains (CTMC). DTMCs have events that operate in discrete time while CTMCs have their events operate within a continuous time-space, as their names would imply. Note that due to no widely agreed-upon nomenclature within relevant literature, some papers may refer to CTMCs as Markov Processes, rather than a Markov Chain that operates in continuous time.

The formal definitions for DTMCs and CTMCs are as follows. A set of random variables $X_1, X_2, X_3, \ldots, X_n$ that have the Markov Property, that is, $Pr(X_{n+1} = x \mid X_1, X_2 = x_2, \ldots, X_n = x_n) =$

DOI: 10.54254/2753-8818/36/20240593

$Pr(X_1 = x_1, \ldots, X_n = x_n)$ are considered a Markov Chain with discrete time. A stochastic process $X(t): t \geq 0$ with discrete state space S is considered a continuous time Markov Chain if for all $t \geq 0, s \geq 0, i \in S, j \in S$, $P(X(s+t) = j | X(s) = i, X(u): 0 \geq u < s) = P(X(s+t) = j | X(s) = i) = P_{ij}(T)$ [2]. Markov Chains are a widely used tool within the field of statistics and form the basis of many processes such as the MCMC process and the Poisson process and is used in both Bayesian and Frequentist methods.

In more specificity, MCMC processes are a group of algorithms that seek to create samples from the probability distribution of a continuous random variable by constructing a Markov Chain of the continuous random variable. Markov Chains are also frequently used outside of statistics in more applied mathematics. It is used within chemistry to map chains of molecular reactions in a mathematical way [3], and it is used within Biology as a Hidden Markov Model (HMM) to model nucleotide structures [4]. Within this paper, however, the authors intend to utilise and explore MCMC processes and NLP through the Three-Pool Process.

2. The three-pool model

2.1. Introduction of three-pool model

For comprehending and quantifying the cycle of chemical elements, like carbon in ecosystems, a key conceptual landscape is the three-pool model. It is often used in ecology, biogeochemistry, and environment research. The ecosystem is divided into three main pools or compartments by the model. Each of them stands for a distinct position or state of the chemical element [5].

The pools mainly encompass three parts. Firstly, it is the active pool. This encompasses the components cycling swiftly in the environment, like those discovered in animals and plants, and in the soil microbial biomass. The chemical elements in this pool own a relatively short turnover time. Besides, the time varies within day from some days to some years. Secondly, the slow pool. There exist chemical elements in this pool which cycle in the environment more slowly. It is comprised of soil organic debris decomposing more slowly than the materials in the active pool. Chemical elements in the pool may have long turnover times that vary between years and decades. Thirdly, the passive pool. This consists of components cycling in the environment exceedingly slowly. It is composed of organic matter or highly stable organic stuff, which are buried deeply like charcoal. In this pool, the element turnover times can range between centuries and millennia. The above-described model is particularly applied to ecological research since it makes the transformations between disparate element states easier to view and gauge.

2.2. Rationale for using MCMC in three-pool model analysis

It is necessary to understand the complex correlations and transformations of elements between the pools, which brings about the MCMC application to three-pool models. Because these models become non-linear and poly-dimensional, conventional statistical approaches perhaps cannot sufficiently capture their features. As the models are proficient in controlling the complex probability frameworks, MCMC offers an intricate approach with the aim of estimating the relative parameters, quantifies uncertainties, and improves the understanding of the system dynamics [6].

By virtue of its versatility, MCMC can contain various information, like isotope ratios and flux measurements. They are important to ecological modeling. The MCMC offers a more intricate picture of the processes, which underlies the three-pool model through establishing a Markov chain simulating the transformations between nations, efficiently explaining the variability and indeterminacy intrinsic in ecosystems. Furthermore, through the model, an intricate system is captured where components nonlinearly move between nations and own intricate correlations between two pools. This complexity makes the traditional analysis hard to use. On the other side, MCMC approaches are expert at tackling difficulties. They conquer that through cautiously investigating the big parameter space. Accordingly, it is possible to completely understand and characterize the interactions and dynamics in the three-pool framework.

DOI: 10.54254/2753-8818/36/20240593

2.3. Analysis of three-pool models through MCMC

Apart from the states aligning with each pool, the transformation probabilities between two states are stated as parts of the MCMC analysis on the models. The specific characteristics of every pool can be considered by means of MCMC calibration, reflecting the cyclical essence of the ecosystem. Through making it easier to investigate the parameter space, the approach makes it possible to measure the rates such as decay within the slow pool or cumulation within the passive pool.

The detailed steps are listed below. The first step is to define the Mathematical Model. First, the equations need to be modelled. The differential equations describe each pool's dynamics, and they should be defined as [7]

$$\frac{dC_1}{dt} = input_1 - k_{12}C_1 + k_{21}C_2 - k_{13}C_1 + k_{31}C_3 \quad (1)$$

$$\frac{dC_2}{dt} = input_2 + k_{12}C_1 - k_{21}C_2 + k_{32}C_3 - k_{23}C_2 \quad (2)$$

$$\frac{dC_3}{dt} = input_3 + k_{13}C_1 + k_{23}C_2 - k_{31}C_3 - k_{32}C_3. \quad (3)$$

In the formulas, (C_i) means the quantity of materials in pool (I), $(input_i)$ means the external input to pool (I), and (k_{ij}) means the transfer rates between pools. The Second is parameterization, which determine which parameters require data-driven estimation. These could be the (k_{ij}) transfer rates in the equations, as well as the inputs if they are not directly measured.

The second step is to prepare the data. Gather or create data that accurately depicts the system's behavior. Time series measurements of pool sizes or fluxes between pools may fall under this category.

The third step is to choose a software framework. There are a variety of software packages that can be utilized, like PyMC3 in Python, JAGS via rjags, and Stan using the rstan package in R, to implement MCMC. The decision is based on the experience and the requirements of the model.

The fourth step is to implement the model. Firstly, the model needs to be written. Give a definition of the model in the language of the selected framework, including the likelihood of the data given the parameters and the prior distributions for each parameter. As an example, a model could be built that can block in Stan that contains the differential equations needed to calculate the likelihood of the observed data. Secondly, prior to receiving the data, there is a need to select priors that represent people's understanding of or presumptions about the parameters. If there is no information about prior states already, these could be as straightforward as uniform priors or more detailed distributions.

The fifth step is to run MCMC simulations. When running the MCMC simulation, samples will come from the parameter posterior distributions. Typically, this entails deciding how many iterations to run in each chain, how many chains to run in total, and how many iterations to discard as burn-in.

The sixth step is to diagnose and validate. Firstly, there is a need to do is check the convergence. Make use of the software package's diagnostic tools to make sure the MCMC chains have converged to the desired distribution. The R-hat statistic and visual inspection of trace plots are two frequently used diagnostic techniques. Secondly, there is the need to analysis the posterior. To estimate the parameters, such as the mean, median, or credible intervals, analyze the posterior samples. As a result, there can be better understanding of which parameter values match the data and model the best. Thirdly, it is model checking. Use posterior predictive checks or a comparison of the model's predictions and observed data to validate the model [8].

The seventh step is interpretation and reporting. The analysis of the findings should consider the application or scientific question. This includes talking about the parameter estimations, their uncertainties, and how the dynamics of the system are understood in light of them. Extensive information about the transitions of elements between pools is needed for MCMC modeling. The Markov chain's transition probabilities, which MCMC employs to model the system's behavior over time, are based on this data. Due to its complexity and the random system characteristics, standard methods cannot accurately produce estimations for parameters, such as transition rates or residence

DOI: 10.54254/2753-8818/36/20240593

lengths. Nonetheless, MCMC may offer estimations for the parameters through implementing thousands or millions of times.

3. Mathematical framework of Markov chains in NLP

3.1. Introduction of NLP

The application of Markov chains in NLP typically involves constructing a transition matrix that represents the probabilities of moving from one word to another. This matrix is built from a corpus of text by analyzing word sequences and calculating the likelihood of a word following another.

The authors shall define $P_{i,j}$ as the probability of transitioning from word i to word j. However, in real estimation, the result can be constrained by traditional Markov chains since it is impossible to predict one word just based on the previous one. Therefore, in NLP model, n-gram model is utilized which is an extension of Markov chain. In this model, it is considered that there are $n-1$ words that can be used to predict the nth word. To be specific, if n equals to 4, it represents the fifth word that can be predicted by using the previous four words. If one defines the states of different words as $W = w_1, w_2, w_3, w_4, \dots w_n$ in an n-gram model, it can be further written as $P(W) = P(w_1) \times P(w_2|w_1) \times P(w_3|w_1, w_2) \times \dots \times P(w_n|w_1, w_2, \dots, w_{n-1})$ [9].

One of the most wide-used applications of Markov chains in NLP is in text generation. By starting with an initial word or phrase, a transition matrix is used to probabilistically judge subsequent word. Markov chains can generate new sentences that mimic the style of the input text. This approach has been used in various applications, from automated story generation to simulating dialogue in virtual assistants.

Beyond text generation, Markov chains serve as the basis for more complex language models. They are used to predict the likelihood of a sequence of words, which is essential for tasks like speech recognition, auto-completion in text editors, and machine translation. The simplicity of Markov chain models makes them faster and more computationally efficient. Although they are often outperformed by more complex models, such as neural networks, for tasks requiring understanding of long-term dependencies.

3.2. Challenges and limitations

Even though it is available to use Markov chain, there are still significant limitations in NLP. It can either oversimplify the model or be too complicated to predict the occurrence of words and phrases. If the probability of state only depends on the previous state, it can result in texts that lack coherence especially in long passages. What is more, human writing is always emotional, and it is impossible to capture the complexities of human language. Writers may not have fixed emotions when they write, and the intensity of emotions is closely related to the use of words. Therefore, it seems that it is easier to analyze existing works such as Shakespeare's articles instead of predicting how a writer might write. In the context of the higher-order n-gram models, the number of words and phrases grows exponentially. The explosion of the possible states can lead to data sparsity problems, which makes it more difficult to process the model with increasing computational bases [10].

To overcome these limitations, researchers have integrated Markov chains with other statistical and machine learning models. HMMs, for example, extend Markov chains by incorporating hidden states that can model parts of speech or other linguistic features, providing a richer understanding of language structure. More recently, the occurrence of deep learning has exceeded Markov chains in many NLP tasks. However, Markov chains remain a valuable tool for certain applications due to their simplicity and efficiency.

Markov chains have played a fundamental role in the development of NLP, offering a straightforward yet powerful method for modeling language and generating text. While their application in NLP systems has been overshadowed by the rise of deep learning, they are still serving as a significant tool for understanding the nature of language. As NLP advances, it is likely that Markov chains are still widely used, either in their pure form or integrated into more complex models, which all proved their enduring value in the field of computational linguistics.

DOI: 10.54254/2753-8818/36/20240593

4. Conclusion

In this paper, the origin of Markov chain is demonstrated at the beginning. It can be divided into two main groups, which are discrete-time and continuous-time Monte Carlo. A major part of the essay concentrates on the applications of Markov chain. Markov chain is a basic model for many applications and there are also many Markov chain-based models which can be utilized in different areas. Firstly, the three-pool model with Markov chain is introduced to present extensive applications in areas of physics, chemistry and so on. It is typically used to describe a system that has three interconnected states and these states transition between each other with certain probabilities. The necessities of the three-pool Markov chain are given, along with the detailed modeling steps. Apart from this, natural language processing is another application. By using the Markov chain model, words and phrases will be predicted which can be used to generate articles and make it easier to research existing literature. Certainly, limitations for this application are obvious as the human mind is unpredictable, and language is sometimes art than science. The whole essay is to show that Markov chain is widely used, and its extensions can help build more useful models to solve more complex problems.

Authors Contribution

All the authors contributed equally and their names were listed in alphabetical order.

References

[1] Ching, W.-K., Huang, X., Ng, M. K., & Siu, T.-K. (2013). Markov chains. International Series in Operations Research and Management Science, Springer.

[2] Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1998). Markov chain Monte Carlo in practice: Interdisciplinary statistics. Chapman & Hall.

[3] Anderson, D. F., and Kurtz, T. G. (2011). Continuous Time Markov chain models for Chemical Reaction Networks. Design and Analysis of Biomolecular Circuits, Springer.

[4] Yoon, B.-J. (2009). Hidden markov models and their applications in biological sequence analysis. Current Genomics, 10(6), 402–415.

[5] Bird, S., Klein, E., and Loper, E. (2009). Natural language processing with Python: analyzing text with the natural language toolkit, O'Reilly Media, Inc.

[6] Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2), 257-286.

[7] LI Hong-man. (2023). Research on Construction Cost Estimation of Highway Engineering Based on Markov Chain. Journal of Liaoning University of Technology (Natural Science Edition), 43(3), 201-205.

[8] Meng Ping, Wang Guohua, Guo Hongzhe, Jiang Tao. (2023). Identifying cancer driver genes using a two-stage random walk with restart on a gene interaction network, Computers in Biology and Medicine, 158, 106810.

[9] Li, Hongman. (2023). Research on Construction Cost Estimation of Highway Engineering Based on Markov Chain. Journal of Liaoning University of Technology (Natural Science Edition), 43(3), 201-205.

[10] Zhang, Guoqi, Hou, Yue, Wang, Kangbo. (2023). Vulnerability Analysis of Ship in Preliminary Design Stage Based on Markov Chain, Ship Engineering, 45(3), 67-72.