COMP 480/580 — Probabilistic Algorithms and Data Structure Nov 26, 2024

Lecture 26

Lecturer: Anshumali Shrivastava *Scribe By: Angel Romero, Saakshi Gupta, Joseph Mohanty, Rohith Kumar Kaza, Lize Shao*

1 Introduction to Markov Chains

Markov Chains are mathematical systems that undergo transitions from one state to another in a probabilistic manner. They are named after the Russian mathematician Andrey Markov, who introduced this concept in the early 20th century.

1.1 What is a Markov Chain?

A Markov Chain is a model used to represent systems that move between different states based on probabilities. These probabilities depend only on the current state, not on the sequence of states that preceded it. This is known as the **Markov Property** or the **memoryless property**

.

Key Features of Markov Chains:

- **States:** The system can be in one of a finite or infinite number of possible states.
- **Transition Probabilities:** Each state has a set of probabilities that define how likely it is to move to other states.
- **Stochastic Process:** The transitions between states occur in a random, probabilistic manner.

1.2 Why Are Markov Chains Useful?

Markov Chains are widely used to model real-world systems that involve uncertainty or randomness. For example:

- **Queueing Systems:** Modeling customer behavior in lines at supermarkets or call centers.
- **PageRank Algorithm:** Ranking web pages in search engines by modeling a user's random navigation on the internet.
- **Weather Prediction:** Predicting whether it will be sunny, rainy, or cloudy based on current conditions.

2 Applications of Markov Chains

2.1 Predicting Customer Behavior

Markov Chains help in performing detailed Customer Behavior Analysis.
The state representation is

Brand States:- The different brands/categories customers choose from.
Customer Segments:- Loyal customers, price-sensitive customers, quality-focused customers. Markov Chains are used to are used to track customer movements between brands, seasonal variational analysis.
Given below is an example of the transition matrix

From or To	Kellogg's	General Mills	Post	Private Label
Kellogg's	0.6	0.2	0.1	0.1
General Mills	0.3	0.5	0.1	0.1
Post	0.2	0.2	0.4	0.2
Private Label	0.15	0.25	0.1	0.5

Insights can be drawn to improve supply chain management, Inventory management, seasonal product planning etc.

2.2 Markov Chains in PageRank

PageRank is a fundamental algorithm used in web search engines to rank web pages based on their importance. It is modeled using Markov Chains, particularly through the Random Surfer Model.

Random Surfer Model:

- A web user (random surfer) navigates the internet by randomly clicking on links.
- At each step, the user either:
  - Follows a hyperlink to a connected page with probability $(1 - \alpha)$, or
  - "Jumps" to a random page on the internet with probability α (Random Jump Factor or Reset Factor).

Mathematical Representation:
Let n be the total number of web pages. Define a transition matrix M for the web graph:

$$M[i][j] = \begin{cases} \frac{1}{\text{out-degree}(i)}, & \text{if there is a link from } i \text{ to } j, \\ 0, & \text{otherwise.} \end{cases}$$

The PageRank vector r satisfies:

$$r = \alpha \cdot \frac{1}{n} \cdot \mathbf{1} + (1 - \alpha) \cdot M^T \cdot r,$$

where:

- α is the random jump factor, typically 0.15,
- $\mathbf{1}$ is a vector of ones (used to create the uniform random jump vector).

Proof of Convergence:

1. The matrix M is stochastic, meaning each row sums to 1. However, it may not be irreducible or aperiodic.

2. Adding the random jump factor (α) ensures that the modified transition matrix

$$P = \alpha \cdot \frac{1}{n} \cdot \mathbf{1} + (1-\alpha) \cdot M$$

   is irreducible and aperiodic. This is because every state (web page) can be reached from any other state with positive probability.

3. By the Perron-Frobenius theorem, such a matrix has a unique stationary distribution. Iterative multiplication of P converges to this stationary distribution.

4. The stationary distribution of P is the PageRank vector r, satisfying $P^T \cdot r = r$.

Applications in Web Search:

1. **Ranking of Web Pages:** Pages with higher PageRank values are deemed more important and are ranked higher in search results.

2. **Handling Dead Ends and Spider Traps:** Random jumps (α) ensure that the algorithm can handle dead ends (pages with no outgoing links) and spider traps (sets of pages that only link to each other).

3. **Personalized PageRank:** By modifying the random jump vector, the algorithm can prioritize specific categories of pages or user preferences.

4. **Link Analysis in Networks:** Beyond web search, PageRank is used to analyze social networks, citation networks, and any graph-based systems.

Example:
Suppose there are 3 pages A, B, and C, with the following links:

$$A \to B, \quad A \to C, \quad B \to C, \quad C \to A.$$

The transition matrix M is:

$$M = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

The modified matrix P with $\alpha = 0.15$ is:

$$P = 0.15 \cdot \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} + 0.85 \cdot \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

Iterative multiplication with the vector r converges to the unique stationary distribution, which represents the PageRank values for A, B, and C.

Conclusion:
Markov Chains form the mathematical foundation of the PageRank algorithm. The introduction of the random jump factor ensures convergence and robustness, making it one of the most widely used algorithms for web ranking and graph analysis.

3 Limitations in Markov Chains

While predicting or analyzing behavior using Markov Chain, the **memory-less assumption**, i.e., assuming that future states depend on the current state can sometimes lead to information loss due to simplification in the over-complex real-world analysis, where the dependency of future states extends beyond the immediate previous state.

State Space Explosion-Computational Complexity: As the number of states increases the state space explodes, the complexity of the model grows exponentially.

Large transition Matrix requires more resources to compute the results. Increased memory issues with storage and calculation of the transition probabilities. Lastly, there are also numerical stability issues with high dimensional state spaces.

Model Validation and Verification can be challenging. Extensive testing may be required, continuous model refinement and cross-validation techniques may be required. Small changes in model parameters can greatly affect the predictions, robust methods are needed for assessing the model's reliability.