www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 5 Issue 5 May 2016, Page No. 16467-16470

Weather prediction using CPT+ algorithm

Dr. Bhawna Mallick[1], Kriti Raj[2], Himani[3]

[1]Uttar Pradesh Technical University, Galgotias College of Engg & Tech.,
Greater Noida, UP, India
bhawna.mallick@galgotiacollege.edu

[2]Uttar Pradesh Technical University, Galgotias College of Engg & Tech.,
Greater Noida, UP, India
kritiraj31may@gmail.com

[3]Uttar Pradesh Technical University, Galgotias College of Engg & Tech.,
Greater Noida, UP, India
himanichaprana@gmail.com

Abstract: ***Weather forecasting has been one of the most scientifically and technologically challenging problems around the world in the last century. One of the most popular methods used for weather prediction is data mining. Over the years many data mining techniques have been used to forecast weather such as Decision tree and Artificial neural network. In this paper we present a new technique for weather prediction namely CPT+ which is highly efficient than all the methods used till date for prediction.***
Here we present a novel prediction model for weather prediction which losslessly compresses the training data so that all relevant information is available for each prediction. Our approach is incremental, offers a low time complexity for its training phase and is easily adaptable for different applications and contexts.
Keywords: Weather forecasting, Next item prediction, Accuracy, Compression.

1. Introduction

Weather forecasting is a vital application in meteorology and has been one of the most scientifically and technologically challenging problems around the world in the last century. Weather forecasting entails predicting how the present state of the atmosphere will change. Present weather conditions are obtained by ground observations, observations from ships and aircraft, radio-sounds, Doppler radar, and satellites.
This information is sent to meteorological centers where the data are collected, analyzed, and made into a variety of charts, maps, and graphs. Modern high-speed computers transfer the many thousands of observations onto surface and upper-air maps. Computers draw the lines on the maps with help from meteorologists, who correct for any errors. A final map is called an analysis. Computers not only draw the maps but predict how the maps will look sometime in the future. The forecasting of weather by computer is known as numerical weather prediction. Climate is the long-term effect of the sun's radiation on the rotating earth's varied surface and atmosphere. The Day-by-day variations in a given area constitute the weather, whereas climate is the long-term synthesis of such variations. Weather is measured by thermometers, rain gauges, barometers, and other instruments, but the study of climate relies on statistics. Nowadays, such statistics are handled efficiently by computers. A simple, long-term summary of weather changes, however, is still not a true picture of climate. To obtain this requires the analysis of daily, monthly, and yearly patterns.
Climate change is a significant and lasting change in the statistical distribution of weather patterns over periods ranging from decades to millions of years. It may be a change in average weather conditions or the distribution of events around that average (e.g., more or fewer extreme weather events). The term is sometimes used to refer specifically to climate change caused by human activity, as opposed to changes in climate that may have resulted as part of Earth's natural processes. Climate change today is synonymous with anthropogenic global warming. Within scientific journals, however, global warming refers to surface temperature increases, while climate change includes global warming and everything else that increasing greenhouse gas amounts will affect.

2. Related Work

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a huge number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a particular application, such as pattern recognition or data classification, through a learning process. The artificial neuron is an information processing unit that is fundamental to the operation of a neural network.
A Decision Tree is a flow-chart-like tree structure. Each internal node denotes a test on an attribute. Each branch represents an outcome of the test. Leaf nodes represent class distribution. The decision tree structure provides an explicit set of "if-then" rules (rather than abstract mathematical equations), making the results easy to interpret. In the tree structures, leaves represent classifications and branches represent conjunctions of features that lead to those classifications. In decision analysis, a decision tree can be used visually and explicitly to represent decisions and decision making. The concept of information gain is used to decide the splitting value at an internal node. The splitting value that would provide the most information gain is chosen. Formally, information gain is defined by entropy. In other to improve the accuracy and generalization of classification and regression trees, various techniques were introduced like boosting and pruning.

Boosting is a technique for improving the accuracy of a predictive function by applying the function repeatedly in a series and combining the output of each function with weighting so that the total error of the prediction is minimized or growing a number of independent trees in parallel and combine them after all the trees have been developed. Pruning is carried out on the tree to optimize the size of trees and thus reduce over-fitting which is a problem in large, single-tree models where the model begins to fit noise in the data.

3. Proposed Scheme

In this section we present a model for lossless weather prediction that is CPT+. Given a set of training sequences, the problem of sequence prediction consists in finding the next element of a target sequence by only observing its previous items. The number of applications associated with this problem is extensive. It includes applications such as web page pre-fetching, consumer product recommendation, weather forecasting and stock market prediction. The literature on this subject is extensive and there are many different approaches. Two of the most popular are PPM (Prediction by Partial Matching) and DG (Dependency Graph) . Over the years, these models have been greatly improved in terms of time or memory efficiency but their performance remains more or less the same in terms of prediction accuracy. Markov Chains are also widely used for sequence prediction. However, they assume that sequences are Markovian. Other approaches exist such as neural networks and association rules. But all these approaches build prediction lossy models from training sequences. Therefore, they do not use all the information available in training sequences for making predictions. In this paper, we propose a novel approach for sequence prediction that use the whole information from training sequences to perform predictions. The hypothesis is that it would increase prediction accuracy.

3.1 Compact Prediction Tree

The Compact Prediction Tree (CPT) is a recently proposed prediction model [5]. Its main distinctive characteristics with respect to other prediction models are that (1) CPT stores a compressed representation of training sequences with no loss or a small loss and (2) CPT measures the similarity of a sequence to the training sequences to perform a prediction. The similarity measure is noise tolerant and thus allows CPT to predict the next items of subsequences that have not been previously seen in training sequences, whereas other proposed models such as PPM and All-K-order-markov cannot perform prediction in such case. The training process of CPT takes as input a set of training sequences and generates three distinct structures: (1) a Prediction Tree (PT), (2) a Lookup Table (LT) and (3) an Inverted Index. During training, sequences are considered one by one to incrementally build these three structures. For instance, Fig. 1 illustrates the creation of the three structures by the successive insertions of sequences s1 = (A,B,C) s2 = (A,B), s3 = (A,B,D,C), s4 = (B,C) and s5 = (E,A,B,A), where the alphabet Z = {A,B,C,D,E} is used. The Prediction Tree is a type of prefix tree (aka trie). It contains all training sequences. Each tree node represents an item and each training sequence is represented by a path starting from the tree root and ending by an inner node or a leaf. Just like a prefix tree, the prediction tree is a compact representation of the training sequences. Sequences sharing a common prefix share a common path in the tree. The Lookup Table is an associative array which allows to locate any training sequences in the prediction tree with a constant access time. Finally the Inverted Index is a set of bit vectors that indicates for each item i from the alphabet Z, the set of sequences containing i.

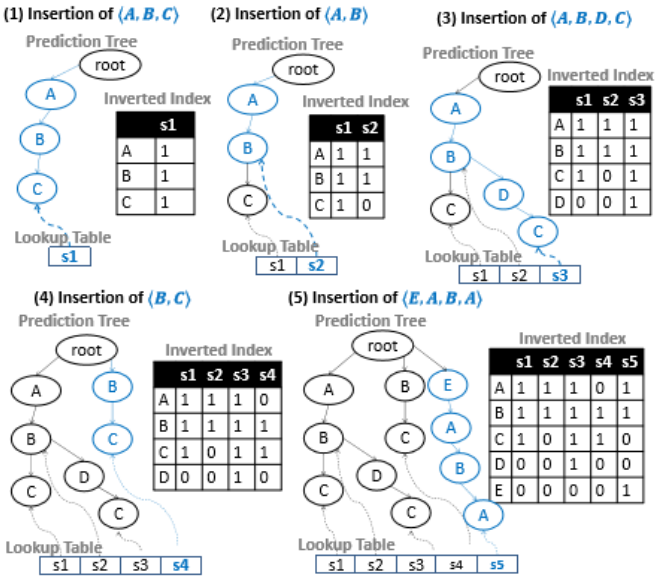

Fig.1. Building CPT structures

3.2 Compression Strategies

CPT has been presented as one of the most accurate sequence prediction model [5] but its high spatial complexity makes CPT unsuitable for applications where the number of sequences is very large. CPT's size is smaller than All-k-Order Markov and TDAG but a few orders of magnitude larger than popular models such as DG and PPM. CPT's prediction tree is the largest data structure and account for most of its spatial complexity. In this section, we focus on strategies to reduce the prediction tree's size.

Strategy 1 Frequent subsequence compression (FSC): In a set of training sequences, frequently occurring subsequences of items can be found. For some datasets, these subsequences can be highly frequent. The FSC strategy identifies these frequent subsequences and replace each of them with a single item. Let be a sequence $s = (i_1,i_2,...,i_n)$. A sequence $c = (i_{m+1},i_{m+2},...,i_{m+k})$ is a subsequence of s, denoted as c v s, iff $1 \leq m \leq m + k \leq n$. For a set of training sequences S, a subsequence d is considered a frequent subsequence iff $|\{t|t \in S \wedge d \text{ v } t\}| \geq$ minsup for a minimum support threshold minsup defined per dataset.

Strategy 2: Simple Branches Compression (SBC): Simple Branches Compression is an intuitive compression strategy that reduces the size of the prediction tree. A simple branch is a branch leading to a single leaf. Thus, each node of a simple branch has between 0 and 1 child. The SBC strategy consists of replacing each simple branch by a single node representing the whole branch. For instance, part (2) of Fig. 2 illustrates the prediction tree obtained by applying the DCF and SBC strategies for the running example. The SBC strategy has respectively replaced the simple branches D,C, B,C and E,x,A by single nodes DC, BC and ExA. Identifying and replacing simple branches is done by traversing the prediction tree from the leafs using the inverted index. Only the nodes with a single

child are visited. Since the Inverted Index and Lookup Table are not affected by this strategy, the only change that needs to be done to the prediction process is to dynamically uncompress nodes representing simple branches when needed.

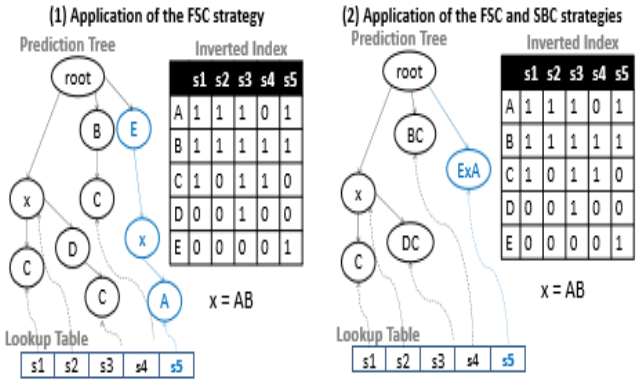

Figure 2: Compression Strategies

3.3 Time Reduction Strategy

Strategy 3: Prediction with improved Noise Reduction (PNR): As previously explained, to predict the next item in+1 of a sequence s = (i1,i2,...,in), CPT uses the suffix of size y of s denoted as Py(s) (the last y items of s), where y is a parameter that need to be set for each dataset. CPT predicts the next item of s by traversing the sequences that are similar to its suffix Py(s). Searching for similar sequences is very fast (O(y)). The more y and k are large, the more subsequences need to be considered, and thus the more the prediction time increases. For a prediction task, items in a training sequence may be considered as noise if their sole presence negatively impact a prediction's outcome. The PNR strategy is based on the hypothesis that noise in training sequences consists of items having a low frequency, where an item's frequency is defined as the number of training sequences containing the item. For this reason, PNR removes only items having a low frequency during the prediction process.

Algorithm 1: The prediction algorithm using PNR

input : Py(s):a sequence suffix,CPT:CPT's
a.structures, TB: a noise ratio,
b. MBR: minimum number of CT updates
output: x: the predicted item(s)

1. queue.add(Py(s));
2. **while** updateCount < MBR ∧ queue.notEmpty()
3. **do**
4. suffix = queue.next();
5. noisyItems = selectLeastFrequentItems(TB);
6. **foreach** noisyItem ∈ noisyItems **do**
7. suffixWithoutNoise = removeItemFromSuffix
8. (suffix, noisyItem);
9. **if** suffixWithoutNoise.length > 1 **then**
10. queue.add(suffixWithoutNoise);
11.end
12. updateCountTable(CPT.CT,
13. suffixWithoutNoise);
14. updateCount++;
15.end
16. return performPrediction(CPT.CT);
17.end

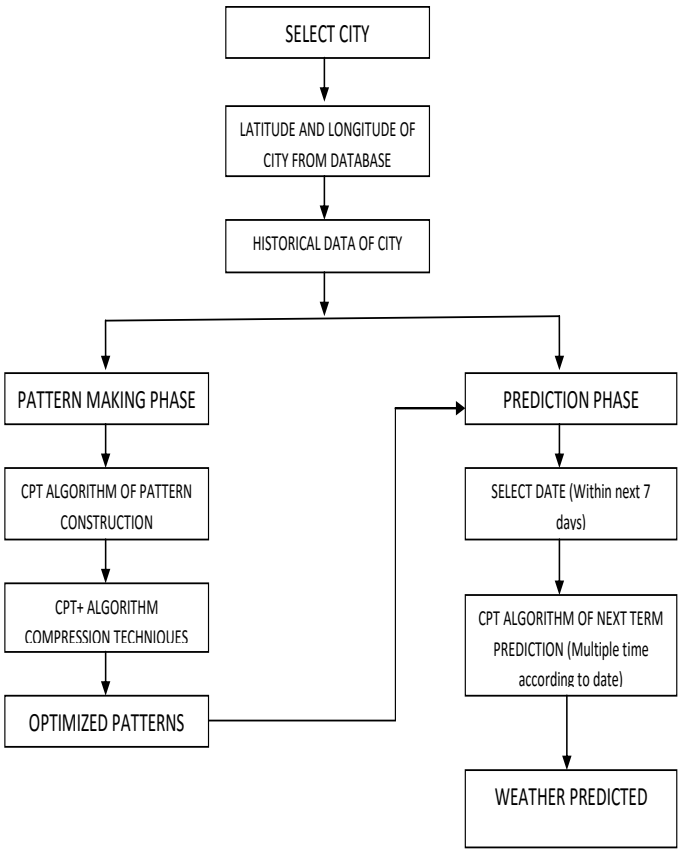

Figure 3: Prediction Architecture

4. Result and Analysis

We designed a framework to compare our approach with state-of-the-art approaches on all these datasets. Each dataset is read in memory. Sequences containing less than three items are discarded. The dataset is then split into a training set and a testing set, using the 10-fold cross-validation technique. For each fold, the training set is used to train each predictor. Once the predictors have been trained, each sequence of the testing set is split into three parts; the context, the prefix and the suffix as shown in Fig. 2. The prefix and suffix size are determined by two parameters named PrefixSize (p) and SuffixSize (s). The context (c) is the remaining part of the sequence and is discarded. For each test sequence, each predictor accepts the prefix as input and makes a prediction. A prediction has three possible outcomes. The prediction is a success if the generated candidate appears in the suffix of the test sequence. The prediction is a no match if the predictor is unable to perform a prediction. Otherwise it is a failure. We define three measures to assess a predictor overall performance. Local Accuracy (eq. 1) is the ratio of successful predictions against the number of failed predictions.

$$Local\ Accuracy = |successes|/(|successes|+|failures|) \quad (1)$$

Coverage (eq. 2) is the ratio of sequence without prediction against the total number of test sequences.

$$Coverage = |no\ matches|/|sequences| \quad (2)$$

Accuracy (eq. 3) is our main measure to evaluates the accuracy of a given predictor. It is the number of successful prediction against the total number of test sequences.

Accuracy =|*successes*|/|*sequences*| (3)

The above measures are used in our experiments as well as the spatial size (in nodes), the training time (in seconds) and the testing time (in seconds). The spatial size is calculated in nodes because the spatial complexity of all predictors can be represented in terms of nodes. This measure is meant to show the spatial complexity and is not used to determine the exact size of a model.

5. Conclusion

The proposed scheme in this paper enhances the technique of weather prediction compared to the existing methodologies. Our approach is incremental, offers a low time complexity for its training phase and is easily adaptable for different applications and contexts. Results show that CPT yield higher accuracy on most datasets (up to 12% more than the second best approach), has better training time .

Yet there is always a possibility for improvement in a certain approach. This approach may be enhanced at some points but some areas might still be improvised.

References

[1] Domenech, J., de la Ossa, B., Sahuquillo, J., Gil, J. A., Pont, A.: A taxonomy of web prediction algorithms. In: Expert Systems with Applications, no. 9, (2012).

[2] Papapetrou, P., Kollios, G., Sclaroff, S., Gunopulos, D.: Discovering Frequent Arrangements of Temporal Intervals. In: Proc. of the 5th IEEE International Conference on Data Mining, pp. 354-361 (2005).

[3] Pitkow, J., Pirolli, P.: Mining longest repeating subsequence to predict world wide web surng. In: Proc. 2nd USENIX Symposium on Internet Technologies and Systems, Boulder, CO, pp. 13-25 (1999).

[4] Prediction. In: Proc. 8th Intern. Conf. on Advanced Data Mining and Applications, Springer LNAI 7713, pp. 431-442 (2012).

[5] Padmanabhan, V.N., Mogul, J.C.: Using Prefetching to Improve World Wide Web Latency, Computer Communications, vol. 16, pp. 358-368 (1998).

[6] Sun, R., Giles, C. L.: Sequence Learning: From Recognition and Prediction to Sequential Decision Making. IEEE Intelligent Systems, vol. 16 no. 4, pp. 67-70 (2001)

[7] Zheng, Z., Kohavi, R., Mason, L.: Real world performance of association rule algorithms. In: Proc. 7th ACM intern. conf. on KDD, pp. 401-406 (2001).

[8] Cleary, J., Witten, I.: Data compression using adaptive coding and partial string matching. IEEE Trans. on Inform. Theory, vol. 24, no. 4, pp. 413-421 (1984)

[9] Deshpande, M., Karypis, G.: Selective Markov models for predicting Web page accesses, ACM Transactions on Internet Technology, vol. 4 no. 2, pp. 163-184 (2004)

[10] Fournier-Viger, P., Gueniche, T., Tseng, V.S.: Using Partially-Ordered Sequential Rules for Sequence

[1] 1998 world cup web site. IEEE Network, vol. 14, no. 3, pp. 30-37 (2000).

[2] Willems, F., Shtarkov, Y., Tjalkens, T.: The context-tree weighting method: Basic properties. IEEE Trans. on Information Theory, vol. 31, no. 3, pp. 653-664 (1995).

[3] ei, J., Han, J., Mortazavi-Asl, B., Wang, J., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: Mining sequential patterns by pattern-growth: the PrefixSpan approach. IEEE Trans. Known. Data Engin. 16(11), 1424–1440 (2004).

Author Profile

<Author Photo>

Taro Denshi received the B.S. and M.S. degrees in Electrical Engineering from Shibaura Institute of Technology in 1997 and 1999, respectively. During 1997-1999, he stayed in Communications Research Laboratory (CRL), Ministry of Posts and Telecommunications of Japan to study digital beam forming antennas, mobile satellite communication systems, and wireless access network using stratospheric platforms. He now with DDI Tokyo Pocket Telephone, Inc.