International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

Forecasting and control of future values on Spatial-Temporal prediction

M. Naveen Babu, JBIET-Hyderabad 1; B. Ravindra Kumar, JBIET-Hyderabad 2;
Motte.Ravi, JBIET-Hyderabad 3; M.Ravi, JBIET-Hyderabad 4;

Abstract

The spatio-temporal prediction problem requires that one or more future values be predicted for input data obtained from sensors at multiple physical locations. Examples of this type of problem include weather prediction, flood prediction, network traffic flow, and so forth. In this paper we provide an overview of this problem, highlighting the principles and issues that come to play in spatio-temporal prediction problems. We describe some recent work in the area of flood prediction to illustrate the use of sophisticated data mining techniques that have been examined as possible solutions. Spatial-temporal prediction use of soft computing techniques in predicting floods (water level at a target location in a river). The techniques and the results find during performance evaluations.

Introduction

Forecasting future values for systems that contain both spatial and temporal features (spatial-temporal) is extremely complex. As an example, consider the problem of predicting precipitation at one location. The amount of previous rainfall in areas close to the target certainly affects this forecast. However, there are many other factors (temperature, time of day, wind direction, wind speed, and so forth) that impact the rainfall prediction. The area of spatial-temporal prediction has been the focus of much research in recent years (Jothityangkoon, Sivapalan, & Viney, 2000; Kelly, Clapp, & Rodriguez, 1998; Pokrajac, & Obradovic, 2001; Roddick, Hornsby, & Spiliopoulou, 2000; Singh, Chaplain, & McLachlan, 1999; Deutsch, & Ramos, 1986; Dougherty, Corne, & Openshaw, 1997;). Due to the excessive complexity of predicting these future values, common practice is to utilize domain experts with extensive experience in both forecasting and the problem domain itself. For example, for flood prediction, the *National Oceanic and Atmospheric Administration (NOAA)* actually employs specialists whose job is to understand the history and specifics of predicting floods on one river. A different domain expert may be hired for a different river. Due to the widespread use of domain experts, spatial-temporal prediction is extremely expensive, and due to the complexity of the nature of the problems, prediction accuracy is often low. Analysis of spatio-temporal systems is complex, since it consists of a large amount of irregular outcomes that incorporate space and time factors.

Flood Prediction

Flood prediction (forecasting) is an example of a spatio-temporal prediction problem whose solution can be addressed because the problem is automatically simplified due to the nature of the problem itself; that is, predicting a flood (or alternatively, a water level or flow value) at a particular point in a river has a well defined spatial aspect, namely, the flow of the river and the lay of the land. Figure1 illustrates this aspect. This figure shows the Serwent Catchmentas provided by the British National River Flow Archive. We don't need to worry about sensor data obtained for spots outside the catchment. In addition, we know the general direction in which water will flow within a catchment. While flood prediction simplifies the spatial influence issue, it does not eliminate it completely. Looking at Figure 1, we see that sensor readings at location 28043 definitely impact those at 28010, but we do not know what the exact influence is. Certainly it would be safe to assume that the impact is somewhat less than the readings at location 28055. But how much? Another issue here is the temporal lag between the readings. The time lag between the influence of a water level reading at 28043 is probably greater than that at 28055, but the actual values vary. There are many common spatio-temporal prediction problems similar to flood prediction. Traffic engineers examine the flow of traffic on highways to predict traffic delays and determine where best to spend funds to upgrade roadways. Network traffic engineers similarly examine flow of packets between sites to predict routing delays and revent network downtime. Similar spatio-temporal prediction problems include electric flow in electric rids and water usage in public water systems. Other spatio-temporal problems may not exactly fit into the flood prediction paradigm, but may be simplified

ISSN:2319-7900

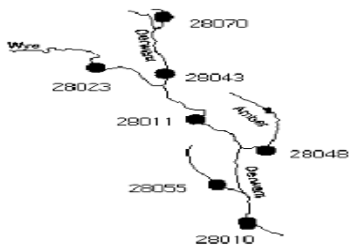

Figure 1. Derwent catchment of Upper Derwent River

by adding spatial influence assumptions to help develop solutions to the more general problem. When looking at predicting ocean water temperature, the movement of water may be approximated based on knowledge by experts as to the normal movements of ocean water. by adding spatial influence assumptions to help develop solutions to the more general problem. When looking at predicting ocean water temperature, the movement of water may be approximated based on knowledge by experts as to the normal movements of ocean water.

Spatio-Temporal Prediction Problem

In this section we briefly review some previous research in the area of spatiotemporal prediction. We classify the work into counting models, stochastic models, Markov models.

A.Counting Models

We view counting models to be algebraic formulations used to capture complex issues such as spatial influence, temporal lag, high dimensionality and hidden relationships. This is a common approach with current weather prediction and flood prediction systems. possible. Environment studies like flood prediction (Burnash, 1996; Comet, 2000), distribution of ice channels and distribution of hydrological parameters in oceans (AARI, 2004) can be approximated, and a prediction can be made by creating a deterministic mathematical counting model. A scientist, based on computation fluid dynamics equations, can design a model that simplifies the relationship among hydrological parameters. The most common approaches to solving the flood prediction problem are based on the use of counting mathematical models. One model is created for each flood prediction site, and it attempts to capture all of the unique features of the catchment at that site. These include such things as river structure upstream, flow rate and soil absorption. To use Values collected by sensory input include such things as water level upstream, temperature, humidity, time of day and rainfall at various points in the catchment. The National Weather Service, part of NOAA, uses an approach based on the *Sacramento Soil Moisture Accounting Model*(NOAA, 2004). This technique predicts water levels by measuring rainfall in the catchment and estimating runoff and soil absorption. the model, at least 20 different parameter values have to be estimated by the domain expert (Comet, 2000). Over time, the model is adjusted by modifying these parameter estimates.

B.Stochastic Models

Stochastic models have also been used to address the spatio-temporal prediction problem. While studying how to manage livestock's waste in a watershed, Cressi and Majure (1997) tried to design a spatio-temporal prediction model mainly based upon spatial statistics analysis. The whole area was divided into smaller grid surfaces, and their spatial characteristics were summed. To capture the temporal characteristics, they used a "three-day area of influence." In other words, data collected at an upper stream more than three days ago would not be considered to affect the lower stream's data. This model did generate a good prediction, unfortunately at the price of a "large variation of the predicted values" with a little modification of the input data. Although this overfitting problem in their work was attributed to the low sampling density in space and time, it is rational to suspect that spatial statistics, with the use of a straightforward timewindow assumption, is not sophisticated enough for a reliable spatio-temporal prediction. In order to perform prediction for a dynamic spatio-temporal system, a model needs to be constructed to incorporate both time and space variability. *AutoRegressive Integrated Moving Average (ARIMA)* time series modeling is one of the popular modeling techniques taking temporal aspects into account. Space variation is usually added statistically to it to reflect the dependence of the system outcome on relative direction as well as distance between locations (Box, & Jenkins, 1970). ARIMA is a powerful model for both stationary and nonstationary time series. The autoregressive portion represents the deviation of the current value of a stochastic process from its mean at time t with a linear aggregate of p previous values of the process and a random *a* drawing from a fixed distribution, which is assumed to be Normal and having mean zero, so called "white noise." Moving average models express the deviation linearly on q number of previous random *a* drawing (Box et al., 1970). Both models describe the stationary process; nonstationary summation is added to model the difference of the process from stationary (Box et al., 1970, 1994; Wei, 1989). In the spatio-temporal problem, the space lag also needs to be ncorporated into the model. *Space-Time AutoRegressive Integrated Moving Average (STARIMA)* is one popular model of this type. It uses a spatial hierarchical ordering of the neighbors of each site and a sequence of *NXN* weighting matrices for N locations to model the influence

ISSN:2319-7900

that the different locations have on a given site (Pfeifer, & Deutsch,
1980a). STARIMA has been widely used on various spatio-temporal problem domains, such as hydrologic modeling (Deutsch et al.,1986) and crime analysis (Pfeifer et al., 1980b).

C.Markov Models and Variants

A stochastic process is a *Markov process* when it satisfies the Markov property. Isaacson (1976) states that "a stochastic process {Xk}, K = 1,2,… with state space S = {1, 2, 3…} is said to satisfy the *Markov property* if for every n and all states i1, i2, … in it is true that P[Xn = in| Xn-1 = in-1, Xn-2 = in-2, …, X1 = i1] = P[Xn = in| Xn-1 = in-1]," where P is the transition probability. Simply speaking, the Markov property declares that the transition probability from current state in-1 to next state in depends only on the current state of the process and has nothing to do with the earlier states in the history of the process. A Markov process is called *Markov chain* if the state space is countable or finite. A Markov chain model, which we will call *Markov model (MM)* in the rest of this chapter, is constructed with states and transitions that can be visualized as a weighted directgraph with collection of *m* vertices, *S*, and directed edges, *E*:
$S = \{Nk \mid K = 1,2, \dots m\}$, and $E = \{ <Ni,Nj> \mid i \in 1,2, \dots, m, j \in 1, 2, \dots, m\}$
With a vertex and an edge in the graph corresponding to a state and a state transition in MM respectively, the weight on each edge of graph is then the transition probability a_{ij} = P(*Nj* | *Ni*) of an MM. If we consider MM as a complete graph, the transition probability distribution can be represented by an m × m matrix, so called *transition matrix*. In real life, there are no systems completely satisfying the Markov property, so this restriction is often loosely defined and assumed. Once an MM is chosen to be the model for a system, it is constructed by defining the appropriate state representations and transition probabilities. The state representation of an MM is usually chosen by domain experts to well represent the property of the system modeled. It is expected that the number of states in an MM is enough to simulate the different states of the system but not so many that there is no significant difference between MM states. To develop a model to simulate the daily rainfall amount based on historical observation, Haan, Allen and Street (1976) used rainfall amount as state representation and grouped observed rainfall amount into six classes (states), which was found to be a reasonable choice of clustering to model the given data after experimenting on several class boundary settings. A statistical method was adopted by Yapo et al.(1993) to cluster observed streamflows when constructing a flood prediction model, in which the K-mean clustering algorithm was used to "minimize the total sum of the square distances. from a streamflow value to cluster center" in order to find the optimal number of intervals and enough streamflow data in each interval. Once the states in the model are decided, the state transition probabilities are usually decided by the ratio of n_{ij}/n_i, where n_{ij} is the number of times that state transits from state i to state j, and ni is the number of times the system is in i.

Data Mining Techniques Used for Flood Prediction

There are other data mining techniques in predicting floods (or to be more precise, water level at a target location in a river). They are: HMM, EMM.We briefly introduce these techniques and the results found during performance evaluations of them.

A.Hidden Markov Models

There have been several approaches to using HMMs in flood prediction. One approach uses multiple HMMs, each representing a discrete state of the prediction site. For example, one HMM could represent the occurrence of a flood, and another could represent an average river condition. In each model, the upstream measurements are treated as *observations*, and the time (relative to a starting time) as states. Then, during prediction, experts choose the model that best matches or *recognizes* the given observations using the *forward-backward procedure* (Rabiner et al., 1986). We call this a *Recognition-Based Model*. The second approach draws an analogy between the components of an HMM – an observable sequence of symbols and a related but hidden state sequence – and the components of the flood prediction problem – an observable sequence of upstream river conditions and a hidden (unknown) sequence of future river conditions at the target site. Given an observation of the present condition of the upstream sites, the flood prediction problem is to *uncover* the best corresponding state sequence that represents the river conditions at the prediction site. Here it helps to think of the states as *related to* rather than *causing* the occurrence of the observation symbols. The accuracy of this model might reveal a few things about the nature of this relationship. We call this a *Viterbi-Based Model*. Our initial experiments examining these two approaches using the water levels as measurements did not yield very positive results. In particular, the Viterbi- Based Model had many states, including some that were used too infrequently during training. Results improved when we considered the *change* in water level relative to the first measurement in a fixed

ISSN:2319-7900

time window because this reduced the variability of the measurements. But the HMMs still did not perform well compared to some existing NN techniques. Thus, we do not consider the use of HMMs further in this chapter.

B.Extensible Markov Models

We also examined the use of the *EMM* in flood prediction. In our experiments, the EMMSim algorithm was implemented using four different similarity measures (Dice, Jaccard, Cosine and Overlap). The threshold of similarity measurement determines whether a new node need to be added to the model; that is, if the similarity between an input reading and each of the states in current EMM is below the threshold, a new node is created representing a new state that is significantly different from current existing states. Our model was built and tested for flood prediction using data of river sensor readings (Dunham et al.,2004) obtained from www.ccg.leeds.ac.uk/simon/nndown.html, which provides real information of water levels at a catchment (Ouse Catchment) in the United Kingdom. The accuracy of the EMM prediction, which in this case is the water level at a designated location one hour ahead of time, was measured using *Root Means Square (RMS)* and *Normalized Absolute Ratio Error (NARE)* as described below:

$$RMS = \sqrt{\frac{\Sigma_{i=1}^{N} (O(t) - p(t))^2}{N}}$$

$$\text{NARE} = \frac{\Sigma_{i=1}^{N} |O(t) - p(t)|}{\Sigma_{i=1}^{N} O(t)}$$

Here O(t) is the observed value and P(t) is the predicted value at time t; N is the total number of test data and t is the time variable. EMM prediction performance was compared with a Neural Network based prediction system, *River Level Forecasting (RLF)* that is an "Artificial Neural Networks for Flood Forecasting" available on the same Web site (Openshaw et al., 1998). Experiments showed that the number of states in the EMM grows at a sub linear rate and levels off once the model has learned the current river behavior. If the river behavior changes, the model will begin learning again (Dunham, 2004). When comparing the performance of EMM to RLF on prediction accuracy, Table 1 shows the EMM performance is better. There are several issues about EMM that deserve further investigation. First, even though the number of states tends to converge when more data are provided, it did not stop growing. To solve this problem, presumably a preset maximum number of states could be used to restrict the growth of number of EMM states, while the learning of links and transition probabilities are still carried on. Second, the water level data that was used to build and test EMM is pretty stable. More modifications of the EMM algorithm might be required when it is applied to data that varies strongly. Possible solutions would be i) choosing more sophisticated similarity algorithms; ii) including other environmental factors beyond water level to better represent the status of the system, iii) making multiple models to match different segments of the varied data, as proposed by Haan in 1976, where different MM models were built for each month. Third, algorithms of states merging and splitting could be included in EMM to closely model the dynamic problem domain.

Conclusions

The spatio-temporal prediction problem is extremely difficult. Conventional solutions using counting models are too labor intensive. By simplifying some of the issues, such as spatial influence specialized subproblems may be examined. We have evaluated data mining techniques to attack the flood prediction problem. Using two new error measurement metrics, we have shown that HMMs do not appear to be better than NNs. However, more sophisticated data mining modeling techniques (EMM, STIFF, NN-ACC) can yield better results than previously proposed methods. While the

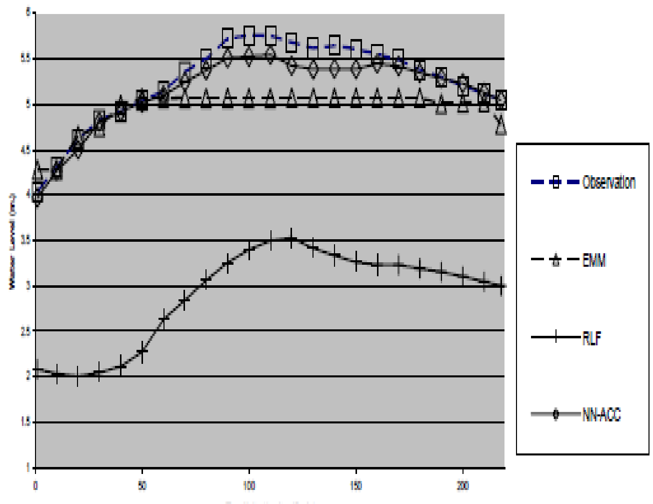

Figure 2. Comparison of NN-ACC, EMM and RLF Models, one-hour prediction

flood prediction problem is a subproblem of the more general spatio-temporal prediction problem, there are many real-world applications of this type. Although we cannot generalize our results to any type of spatio-temporal prediction problem, we feel that they are promising enough that more research is warranted. Currently in the real world, it appears that techniques to address spatio-temporal prediction seem to center around the more simplistic, more understood counting techniques. There does not appear to be any acceptance in the real world to more sophisticated, less understood data mining technqiues. However, we feel that in the future this really should change. Due to the very nature of the problem, and its applicability to many real-world applications, future study to examine better data mining solutions is needed. The potential benefits are quite high.

Model	NARE	RMS
EMM	0.065	0.413
RLF	0.447	2.374
NN-ACC	0.0239	0.145

Table 1. Comparison NN-ACC with EMM and RLF flood prediction models

We propose that future research is needed in the following areas:

• Creation of more sophisticated data mining techniques to model the complex spatio-temporal problems, or at least subproblems thereof.

• Evaluation of other data mining forecasting techniques to the spatiotemporal problem.

• Evaluation of combining data mining tools to attack the spatio-temporal prediction problem.

References

[1]Arctic and Antarctic Research Institute (2004). Russian Federal Service for Hydrometeorology and Environmental Monitoring. Retrieved 2004 from *www.aari.nw.ru*

[2]United States Geological Society. (2004). *Steam gaging and flood forecasting*. Retrieved August 2004 from *http://water.usgs.gov/widFS_209-95/mason-weiger.html*

[3]NOAA (2004). Sacramento soil moisture accounting model. Retrieved from *www.nwstc.noaa.gov/HYDRO/RFS/RFS503b.html*

[4]Li, Z., Liu, L., & Dunham, M.H. (2003a). Considering correlation between variables to improve spatiotemporal forecasting. *Proceedings of the PAKDD'03.*

[5]Li, Z., Dunham, M.H., & Xiao, Y. (2003b). STIFF: A forecasting framework for spatio-temporal data. In O.R. Zaïane, S.J. Simoff, & C. Djeraba (Eds.),*Mining multimedia and complex data.* Springer Verlag. *Lecture Notes in Artificial Intelligence Volume 2797,* pp. 183-198.

[6]Li, Z., Dunham, M.H. & Xiao, Y. (2002,). STIFF: A forecasting framework for spatio-temporal data. *Proceedings of the KDMCD*, PAKDD'02. Springer Verlag. *Lecture Notes in Artificial Intelligence Volume 2797*.

[7]Stroud, J., Muller, P., & Sanso, B. (2001). Dynamic models for spatio-temporal data. *Journal of the Royal Statistical Society Series B*, *63*, 673-689.

[8]Burnash, R., & Ferral, L. (1996, July), *Conceptualization of the Sacramento Soil Moisture Accounting Model.* NWSRFS Users Manual, Part II.3,National Weather Service, NOAA, DOC, Silver Spring, MD.

[9]Bishop, C. (1994). *Neural networks for pattern recognition.* Oxford: Oxford University Press.

[10]Box, G., Jenkins, G., & Reinsel, G. (1994). *Time series analysis*. Englewood Cliffs, NJ: Prentice Hall.

[11]Baum, E. (1989). What size net gives valid generalization? *Neural Computation,1*, 151-160.

[12]Box, E.P.G., & Jenkins, M.G. (1970). *Time series analysis: Forecasting and control*. San Francisco: Holden-Day.

Biographies

M.NAVEEN BABU received B.E degree in Computer Science and Engineering from the Osmania University, Hyderabad, Andrapradesh, in 2004, and M.Tech. in Computer Science and Engineering from the J.N.T. U, Hyderabad, Andrapradesh, in 2010.Presently working as Asst.Professor in Computer Science and Engineering Department in J.B.I.E.T-Hyderabad
Email: mnaveen456@gmail.com

B.RAVINDRA KUMAR received B.Tech degree in Computer Science and Information Technology from the J.N.T.U , Hyderabad, Andrapradesh, in 2004, the M.Tech. degree Computer Science and Engineering from the J.N.T. U , Hyderabad, Andrapradesh, in 2008 and currently pursuing Ph.D Degree from J.N.T. U , Hyderabad, Andrapradesh. Presently working as Asst.Professor in Computer Science and Engineering department in J.B.I.E.T-Hyderabad.
Email: b.ravindra08@gmail.com

MOTTE.RAVI received M.C.A degree in Computer Application from the Osmania University, Hyderabad, Andrapradesh, in 2007, M.Tech in Computer Science and Engineering from the J.N.T. U, Hyderabad, Andrapradesh, in 2011. Presently working as Asst.Professor in Computer

Science and Engineering department in J.B.I.E.T-Hyderabad.
Email: mrv_ravi@yahoo.com

M.RAVI received B.Tech degree in Computer Science and Information Technology from the J.N.T.U , Hyderabad, Andrapradesh, in 2004, and M.Tech in degree Software engineering from the J.N.T. U, Hyderabad, Andrapradesh, in 2009. Presently working as Asst.Professor in Information technology department in J.B.I.E.T-Hyderabad
Email: ravikumar.96522@yahoo.com