Outlier Detection for Training-Based Adaptive Protocols

Hui Liu, Jialin He, Dinesh Rajan and Joseph Camp
Department of Electrical Engineering, Southern Methodist University, Dallas, TX
{huil, jhe, rajand, camp}@smu.edu

***Abstract*—An increasing number of adaptive protocols use training data to learn optimal parameter choices for adaptation in wireless communication networks. For instance, several recent papers have studied link adaptation protocols based on context information such as node velocity and SNR. However, a number of embedded sensors providing context information frequently report erroneous values, e.g., GPS errors and accelerometer lag, producing incorrect information about motion. As a result, the relationship between the context information and optimal parameter choices that the adaptive algorithm is attempting to establish is erroneous. In this paper, we propose an outlier detection algorithm, which detects the corrupted information due to system errors. The proposed outlier detection algorithm is based on an alternating minimization approach. To evaluate the performance of the proposed algorithm, we apply it to a link-level context-aware rate adaptation system. Numerical results on emulated channels and in-field testing demonstrate that the proposed algorithm increases the prediction accuracy of the optimal transmission mode by 87% and the throughput by 18%.**

I. Introduction

Context information from sensors and users' preferences are increasingly collected and used to improve the performance of the transmitter and receiver in context-aware algorithms [1]–[3]. Existing context-aware algorithms utilize the hints extracted from context information for the channel state and establish a model mapping the hints to the transmitter/receiver configurations. Vehicular networks can employ context awareness to traffic coordination, mobile entertainment, and driver safety [4]. Vehicular speed and distance to the neighboring node can assist rate adaptation to maximize the link throughput in vehicular networks [5]. A classification-based context-aware system has been proposed and applied to rate adaptation in [6].

The observed improvements from the existing context-aware algorithms rely on the accuracy of the collected context information. However, there are different degrees of uncertainty in the context data that is collected from various sources [7], [8]. It is impractical to expect that these errors in the collected data would be manually corrected. The established model based on the noisy data would weaken the influence of the correctly collected sensor information, potentially leading to incorrect and/or sub-optimal decisions about the transmitter/receiver configurations.

We refer to such erroneous/noisy data as outliers. In this paper, we propose and evaluate an algorithm for detecting such outliers. The proposed algorithm is based on an alternating minimization approach and provides an ordering of the contextual training data according to its likelihood of being an outlier. To the best of our knowledge, our work is the first to detect outliers in context information and apply it to a context-aware rate adaptation protocols. Our main contributions are summarized as follows:

- We identify the necessity of outlier detection in context information. The existence of outliers degrades the performance of a context-aware rate adaptation system.
- We propose an algorithm to determine the effective and usable training data according to its confidence of being an outlier. We show that a training set that has the noisy outlier removed from the original training set can achieve better performance.
- We experimentally investigate the performance of the outlier detection algorithm in both emulated channels and in-field trials. The selected training set improves the accuracy and throughput of rate adaptation by up to 87.5% and 18.11%, respectively.

The remainder of the paper is organized as follows. We first introduce the problem formulation and proposed algorithm in Section II. In Section III, we perform experiments on a broad set of channels and scenarios to evaluate the proposed outlier framework. Finally, we discuss related work in Section IV and conclude in Section V.

II. Framework for Outlier Detection

In this section, we introduce the algorithm for determining outliers in the contextual data used for adaptive transmission. While the proposed outlier detection method can be applied to any training-based context-aware algorithm, we demonstrate the utility of the outlier detection method in improving the performance of a link-layer rate adaptation algorithm [6].

A. Problem Formulation

In this paper, the context information we consider are SNR, velocity and channel type. Channel type is a simpler way to represent the main characteristics of the infinitely-varied propagation environment, such as fading and multipath [9]. Two channels are said to belong to the same channel type if their link-layer throughput performance of various modulation and coding schemes is *nearly* the same for all context information tuples.

The rate adaptation scheme based on traditional classification is trained and evaluated in two separate stages. In the first stage, a learning algorithm for classification establishes a mapping from the context information to the transmission mode with the highest throughput based on the data in the *training set*. Each data vector in the training set is in the format shown in Table I. The various transmission modes we consider are a combination of the modulation scheme, coding rate, and packet size. The classifier is represented by a function f, where,

$$f:\ \{Channel\ type, SNR, Velocity\} \rightarrow Best\ Mode. \quad (1)$$

An example of this classification-based rate adaptation model is given in Fig. 1.

In the second stage, the performance of the classifier is tested with independently-generated data from a *testing set*. The details of the data collection are discussed in Section III-A, and the numerical performance of the classifier is presented in Section III-B.

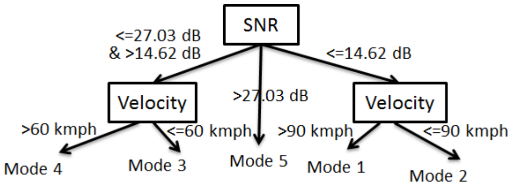

Fig. 1. An example classification-based rate adaptation decision structure.

TABLE I
UTILIZED CONTEXT DATA AND FORMAT FOR CLASSIFICATION

Attr 1	Attr 2	Attr 3	Class
Channel Type (ITU)	Node Velocity	SNR	Best Mode

In practical systems, training data could be corrupted by errors/noise in the measurements. Hence, before the training process, we can eliminate the noisy data points to improve the classifier performance. Let $\mathbf{X} = [\mathbf{x}'_1, \mathbf{x}'_2, ..., \mathbf{x}'_n]$ represent the original training set, where n represents the number of data points, each of which is a row vector in the format shown in Table I. Each vector $\mathbf{x}_i$ is of length m and thus $\mathbf{X} \in \mathbb{R}^{m \times n}$, where m is the number of attributes and equals 3 in all our experiments. Let $\mathbf{Y} \in \mathbb{R}^{l \times n}$ represent the optimum class indicator matrix, where l is the number of classes. If the j^{th} transmission mode achieves the highest throughput for the i^{th} data point given by $\mathbf{x}_i$, then $\mathbf{Y}(j,i) = 1$ and $\mathbf{Y}(\ell, i) = 0, \forall \ell \neq j$.

The purpose of the proposed algorithm is to detect the data points in $[\mathbf{X}, \mathbf{Y}]$ that are significant outliers of the pattern decided by the majority of points in the dataset.

B. Outlier Detection Algorithm

Linear regression modeling has proven to be an effective and simple way to capture the patterns in the training set [10]. This model is represented by appropriately selecting matrix $\mathbf{W} \in \mathbb{R}^{m \times l}$ according to the following:

$$\min_{\mathbf{W}} \quad \|\mathbf{W}^\top \mathbf{X} - \mathbf{Y}\|_F^2 + \alpha Tr(\mathbf{W}^\top \mathbf{W}), \quad (2)$$

where the parameter α controls the regularization weighting term, $Tr(\mathbf{W}^\top \mathbf{W})$, to avoid over-fitting.

Let $\mathbf{B} = \mathbf{W}^\top \mathbf{X} + \mathbf{Y} + \Omega$ be the residual matrix of size $n \times l$ that represents the location of the outlier terms in a dataset [10], where Ω is the random error matrix in the linear regression model. For example, each row of $\mathbf{B}$ corresponds to a data point and a large norm of $\mathbf{B}(i,:)$ suggests a significant deviation of the i^{th} data instance and can be used to detect outliers. Therefore, our outlier detection problem can be formulated as:

$$\min_{\mathbf{W},\mathbf{B}} \quad \|\mathbf{W}^\top \mathbf{X} - \mathbf{B}^\top - \mathbf{Y}\|_F^2 + \alpha Tr(\mathbf{W}^\top \mathbf{W}) + \beta \|\mathbf{B}\|_{2,1}, \quad (3)$$

where $\|\mathbf{B}\|_{2,1}$ is the $\ell_{2,1}$-norm of $\mathbf{B}$, which is defined as follows [11],

$$\|\mathbf{B}\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{l} \mathbf{B}^2(i,j)} = \sum_{i=1}^{n} \|\mathbf{B}(i,:)\|_2. \quad (4)$$

The purpose of $\ell_{2,1}$-norm on $\mathbf{B}$ is two-fold. First, the $\ell_{2,1}$-norm of a matrix ensures that $\mathbf{B}$ is sparse in rows, making it particularly suitable for data selection. The second reason is to prevent a trivial estimate $\mathbf{B} = \mathbf{Y}^\top$ yet derive a meaningful estimate of $\mathbf{W}$. The value of $\|\mathbf{B}(i,:)\|_2$ represents the confidence of each data being an outlier. We then sort the data in ascending order based on $\|\mathbf{B}(i,:)\|_2$ and select the top-k ranked data points.

It can be easily shown that the problem in (3) is jointly convex in $\mathbf{W}$ and $\mathbf{B}$. The first term is convex with respect to $\mathbf{W}$ and $\mathbf{B}$ according to the definition of the Frobenius norm [12]. Since $Tr(\mathbf{W}^\top \mathbf{W}) = \|\mathbf{W}\|_F^2$, the second term is also convex with respect to $\mathbf{W}$. Based on the definition of $\ell_{2,1}$ [12], the third term is convex with respect to $\mathbf{B}$. Since there is no product term involving $\mathbf{W}$ and $\mathbf{B}$ in (3), the convexity property readily follows. The joint convexity suggests that an alternating optimization can be used to find the global optimum [13]. Using an alternate-minimization approach, we optimize $\mathbf{W}$ and $\mathbf{B}$ iteratively by keeping the other matrix fixed.

1) Computing $\mathbf{W}$*:* For a fixed $\mathbf{B}$, the optimal $\mathbf{W}$ can be obtained by minimizing the following problem,

$$\min_{\mathbf{W}} \quad \mathcal{J}(\mathbf{W}) = \|\mathbf{W}^\top \mathbf{X} - \mathbf{C}\|_F^2 + \alpha Tr(\mathbf{W}^\top \mathbf{W}), \quad (5)$$

where $\mathbf{C} = \mathbf{B}^\top + \mathbf{Y}$ is a constant matrix. The solution to (5) is obtained by setting the derivative of (5) equal to zero.

For simplicity, the objective function in (5) can be written as:

$$\mathcal{L}_W = Tr(\mathbf{W}^\top \mathbf{X}\mathbf{X}^\top \mathbf{W} - 2\mathbf{W}^\top \mathbf{X}\mathbf{C}^\top) + \alpha Tr(\mathbf{W}^\top \mathbf{W}).$$

The derivative of $\mathcal{L}_W$ is given by,

$$\frac{\partial \mathcal{L}_W}{\partial \mathbf{W}} = 2\mathbf{X}\mathbf{X}^\top \mathbf{W} - 2\mathbf{X}\mathbf{C}^\top + 2\alpha \mathbf{W}. \quad (6)$$

Setting the derivative to zero, we obtain the following update rule,

$$\mathbf{W} = (\mathbf{X}\mathbf{X}^\top + \alpha \mathbf{I})^{-1} \mathbf{X}\mathbf{C}^\top. \quad (7)$$

2) Computing $\mathbf{B}$*:* Now, for a given value of $\mathbf{W}$, the optimal $\mathbf{B}$ can be obtained by minimizing the following problem,

$$\min_{\mathbf{B}} \quad \mathcal{J}(\mathbf{B}) = \|\mathbf{H} - \mathbf{B}^\top\|_F^2 + \beta \|\mathbf{B}\|_{2,1}, \quad (8)$$

where $\mathbf{H} = \mathbf{W}^\top \mathbf{X} - \mathbf{Y}$.

Since $\|\mathbf{B}\|_{2,1}$ is not differentiable [12], the problem in (8) is not smooth. In this paper, we employ Nesterov's method [14] to solve the non-smooth optimization problem in (8) by optimizing its equivalent smooth convex reformulation [12].

By converting the non-smooth term, i.e., $\|\mathbf{B}\|_{2,1}$, as a constraint, (8) can be formulated into the following optimization problem,

$$\min_{\mathbf{B}\in\mathcal{Z}} f(\mathbf{B}) = \|\mathbf{H} - \mathbf{B}^\top\|_F^2, \tag{9}$$

where $\mathcal{Z} = \{\mathbf{B}| \ \ \|\mathbf{B}\|_{2,1} \le z\}$ and $z \ge 0$ is the radius of the $\ell_{2,1}$-ball, and there is a one-to-one correspondence between β and z.

Theorem 1: The problem in (9) is a constrained smooth convex optimization problem,

Proof: The objective function, $f(\mathbf{B})$, is convex and differentiable. It is easy to verify that $\|\mathbf{B}\|_{2,1}$ satisfies the three norm conditions and is thus a valid norm. Since any norm defines a convex set [15], $\mathcal{Z}$ is a closed and convex set. Our problem defines a convex and differentiable function in a closed and convex set, which implies that problem (9) is a constrained smooth convex optimization problem. ■

In the t-th iteration, the Nesterov's method is based on two variables $\mathbf{B}_t$ and $\mathbf{V}_t$, where $\mathbf{B}_t$ is the approximate solution, and $\mathbf{V}_t$ is the search point. The search point $\mathbf{V}_t$ is the affine combination of $\mathbf{B}_{t-1}$ and $\mathbf{B}_t$ and is given by

$$\mathbf{V}_t = \mathbf{B}_t + \gamma_t(\mathbf{B}_t - \mathbf{B}_{t-1}), \tag{10}$$

where γ_t is a combination coefficient. The approximate solution $\mathbf{B}_{t+1}$ is computed as a "gradient" step of $\mathbf{B}_t$ as

$$\mathbf{B}_{t+1} = \min_{\mathbf{B}\in\mathcal{Z}} \|\mathbf{B} - \mathbf{U}_t\|_F^2, \tag{11}$$

where $\mathbf{U}_t = \mathbf{V}_t - \frac{1}{\lambda_t}\nabla f(\mathbf{B}_t)$, and λ_t is the step size. This step size is determined by linear search according to the Armijo-Goldstein rule [15], and $\nabla f(\mathbf{B}_t)$ is the gradient of $f(\mathbf{B}_t)$, given by

$$\nabla f(\mathbf{B}_t) = \mathbf{B}_t - \mathbf{H}^\top. \tag{12}$$

Problem (11) can be decomposed into n sub-problems as follows:

$$\mathbf{b}_{t+1}^j = \min_{\mathbf{b}^j\in Z^j} \|\mathbf{b}^j - \mathbf{u}_t^j\|_2^2, \tag{13}$$

where $\mathbf{u}_t^j$, $\mathbf{b}^j$ and $\mathbf{b}_{t+1}^j$ are the j-th rows of $\mathbf{U}_t$, $\mathbf{b}$ and $\mathbf{b}_{t+1}$, respectively. The constraint Z^j is defined on $\|\mathbf{b}^j\|$. Each of these sub-problems has a closed form solution as follows,

$$\mathbf{b}_{t+1}^j = \begin{cases} \left(1 - \frac{\beta}{\lambda_t\|\mathbf{u}_t^j\|}\right)\mathbf{u}_t^j & \text{if } \ \|\mathbf{u}_t^j\| \ge \frac{\beta}{\lambda_t} \\ 0 \quad \text{otherwise} \end{cases} \tag{14}$$

For notational convenience, define $f_{\lambda,\mathbf{V}_t}$ as,

$$f_{\lambda,\mathbf{V}}(\mathbf{B}) = f(\mathbf{B}) + \langle\nabla f(\mathbf{B}), \mathbf{V} - \mathbf{B}\rangle + \frac{\lambda}{2}\|\mathbf{V} - \mathbf{B}\|_F^2; \tag{15}$$

then the detailed algorithm is shown in Algorithm 1.

Algorithm 1 Outlier Detection Algorithm

```
Input: $\{\mathbf{X}, \mathbf{Y}, \alpha, \beta\}$
Output: $k$ most relevant data points
Initialize $\eta_0 = 0$, $\eta_1 = 1$, $\mathbf{B}_1 = \mathbf{B}_0$, $t = 1$
Initialize $\mathbf{W}_1 = (\mathbf{X}\mathbf{X}^\top + \alpha\mathbf{I})^{-1}\mathbf{X}\mathbf{Y}^\top$
while Not convergent do
    Set $\mathbf{H} = \mathbf{W}_t^\top\mathbf{X} - \mathbf{Y}$
    Set $\mathbf{V}_t = \mathbf{B}_t + \frac{\eta_{t-1}-1}{\eta_t}(\mathbf{B}_t - \mathbf{B}_{t-1})$
    Set $\nabla f(\mathbf{B}_t) = \mathbf{B}_t - \mathbf{H}^\top$
    loop
        Set $\lambda = 2^j\lambda_t$ and $\mathbf{U}_t = \mathbf{V}_t - \frac{1}{\lambda}\nabla f(\mathbf{B}_t)$
        Compute $\mathbf{B}_{t+1} = \min_{\mathbf{B}\in\mathcal{Z}} \|\mathbf{B} - \mathbf{U}_t\|_F^2$
        if $f(\mathbf{B}_{t+1}) \le f_{\lambda,\mathbf{V}_t}(\mathbf{B}_{t+1})$ then
            $\lambda_{t+1} = \lambda$, break
        end if
        $j = j + 1$
    end loop
    Set $\eta_{t+1} = \frac{1+\sqrt{1+4\eta_t}}{2}$ and $\mathbf{C} = \mathbf{B}_{t+1}^\top + \mathbf{Y}$
    Update $\mathbf{W}_{t+1} \leftarrow (\mathbf{X}\mathbf{X}^\top + \alpha\mathbf{I})^{-1}\mathbf{X}\mathbf{C}^\top$
    Set $t = t + 1$
end while
Sort each data according to $\|\mathbf{B}(i,:)\|_2$ in ascending order
and select the top-$k$ ranked ones.
```

III. Experimental Evaluation

In this section, to evaluate the proposed outlier detection algorithm for context information, we directly compare rate adaptation performance when using the original training set versus the training set after the removal of outliers. To do so, we perform our experiments on emulated and in-field channels and show that eliminating the outlier from the training of context-aware rate adaptation can improve the accuracy and throughput by up to 87.5% and 18.11%, respectively.

A. Experimental Set-up with Emulated Channels

In order to evaluate the proposed outlier detection algorithm for context information, we set up an experimental system with two wireless nodes communicating across emulated channels. We use a Linux-based Gateworks 2358 with Ubiquiti, 802.11-compliant radios as nodes. We use an Azimuth ACE-MX channel emulator to generate controllable and repeatable channel models including a broad range of industry-standard models for our experiments [16].

There are 18 transmission modes (i.e., 18 classes for the purposes of classification) corresponding to the combination of 9 modulation and coding rates ({6, 9, 11, 12, 18, 24, 36, 48, 54} Mbps) and 2 packet sizes ({100, 1000} Bytes). The transmission power is 27 dBm. The training set contains 40 scenarios corresponding to 5 velocities ({0, 30, 60, 90, 120} kmph) and 8 attenuations of transmission power ({0, 6, 12, 18, 24, 30, 36, 42} dB). For each of these scenarios, we test the performance on four different channel types specified by the ITU [9]. Since the emulator allows for repeatable and controllable channel emulation, we identify the transmission mode with the highest throughput using an exhaustive search in each of these 40 scenarios and 4 channel types. Based on the training set, we use the C4.5 learning algorithm to generate the classification-based rate decision structure [6]. The selected attenuations of the transmission power and velocity values are evenly spaced to cover as many scenarios as possible. We then create a testing set by randomly generating values of the power attenuation, velocity and channel type and studying their performance across different modes. This exhaustive search across all different modes allows us to quantify the accuracy of the decisions made using the classification-based rate adaptation algorithm.

In this paper, the throughput, G_{th}, for each mode is calculated as

$$G_{th} = (1 - PER) * R * \frac{l_{payload}}{l_{packet}}, \tag{16}$$

where R is the physical data rate at one mode, PER is the packet error rate, and $l_{payload}$ and l_{packet} are the payload length and packet length, respectively.

B. Performance of Outlier Detection Algorithm

The proposed algorithm sorts the contextual data collected in different scenarios according to the likelihood of being an outlier. Then, we gradually delete the data with the highest likelihood of being an outlier and test the trained classifier in the classification-based rate adaptation system. We first test the performance of the proposed outlier detection algorithm by setting α and β to 0.01 and 0.001, respectively. The parameter α is introduced to avoid over-fitting and it is empirically set to 0.01 [17], [18]. More details about the selection of β are introduced in Section III-C. The accuracy of predicting the mode with the highest throughput is shown in Fig. 2. As noisy data is deleted and the ratio of the used training data decreases from 100%, the accuracy of the prediction improves. When the ratio of the used training data decreases from 78% to 69%, the rate prediction accuracy increases by 13%. This dramatic change of performance indicates a potentially reliable threshold for outlier elimination of this training set. The best accuracy performance is achieved when the ratio of the used training data is around 56% of the original training set, outperforming that with the original training set around 54%. However, as the ratio of the used training data continues to decrease beyond 50%, the accuracy decreases since the lack of training data degrades the accuracy of the classification-based rate adaptation system.

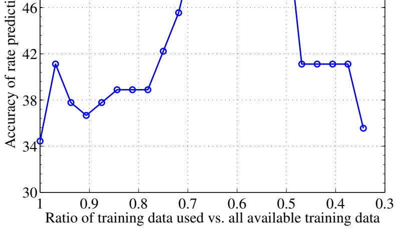

Fig. 2. Rate prediction accuracy with outlier detection ($\beta = 0.001$).

The throughput performance of outlier detection is shown in Fig. 3. From the figure, the initial decrease in the ratio of the used training dataset leads to an increase in the achievable throughput and reduces the gap to the maximum achievable throughput. When the selected data is between 69% to 65% of all training data points, the difference from the maximum possible throughput is the smallest and the throughput improvement over rate prediction based on the original training set is 18.11%.

Comparing Figures 2 and 3, we observe that the throughput improvement and the accuracy share a similar trend that an eventual decline follows a rise, but their detailed variations are different. Table II illustrates a simple testing set to help us understand this observation. Consider two different classifiers **a** and **b** that are trained based on different algorithms/training datasets. Let these classifiers be tested with the data given in Table II and let their prediction results be $\{1, 2, 3, 2\}$ and $\{2, 1, 1, 3\}$, respectively. Thus, the accuracy and sum of the throughput are 50% and 38.06 Mbps for the classifier **a**, respectively. Similarly, the accuracy and sum of the throughput are 25% and 36.07 Mbps for **b**. We can see that the improvement with respect to accuracy of **a** over **b** is 100%, however, the throughput improvement is only 5.52%. There are two main reasons for this discrepancy: (1) the data points, predicted incorrectly by one classifier with higher accuracy, might be correctly classified by the other classifier with lower accuracy, such as data point 3 in Table II; (2) although both classifiers may fail to predict the optimal transmission mode, the performance of the predicted modes are different where one has higher throughput than the other. For example, both **a** and **b** fail to predict data point 4; however, the throughput of the mode predicted by **b** is higher than that of **a**.

TABLE II
AN EXAMPLE TEST SET

Data Points	Throughput		
	Mode 1	Mode 2	Mode 3
1	12.10	9.55	8.97
2	10.40	13.50	9.94
3	13.3	6.58	10.81
4	4.90	1.65	2.82

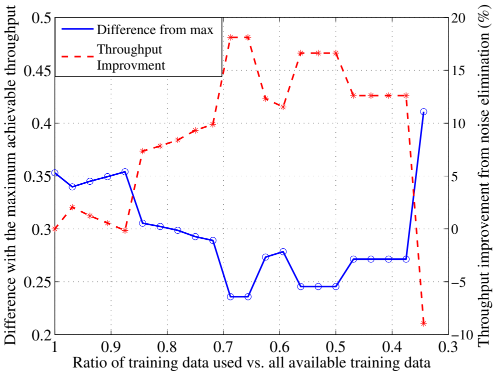

Fig. 3. The throughput gain from outlier detection ($\beta = 0.001$).

C. Effect of β and Ratio of Used Training Data

The parameter β controls the sparsity of the outlier matrix $\mathbf{B}$. A sparse $\mathbf{B}$ could result in fewer outliers being detected, while a non-sparse matrix could result in removing reliable data points as well. Another important parameter is the ratio of the used training data. In this section, we quantify the performance of our algorithm for various values of β and the ratio of the training data used.

In Fig. 4, we show the accuracy of predicting the mode with the highest transmission throughput for different values of β and different ratios of the training data used with respect to the original set. This figure demonstrates that the performance is fairly robust with respect to the choice of β. For each fixed β, decreasing the ratio of the training data used leads to an initial increase in the accuracy. Subsequent decreases in the dataset size lead to a decrease in performance due to the limited amount of effective data that is used for training. This behavior can be used to select the optimal size of used training set. We also observe that between the two parameters, the performance is more sensitive to the ratio of the used training data based on the proposed algorithm. The throughput improvement from the outlier detection scheme, which uses different β values, is shown as in Fig. 5. We have similar observations about the throughput performance as these for the accuracy performance.

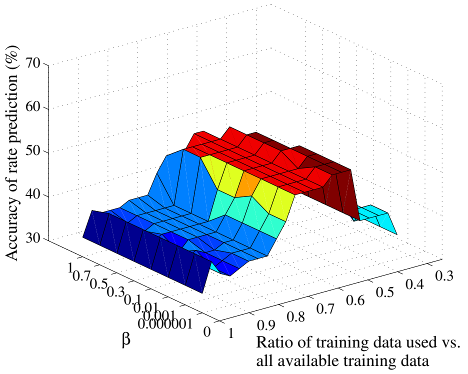

Fig. 4. The rate prediction accuracy gain with different β values.

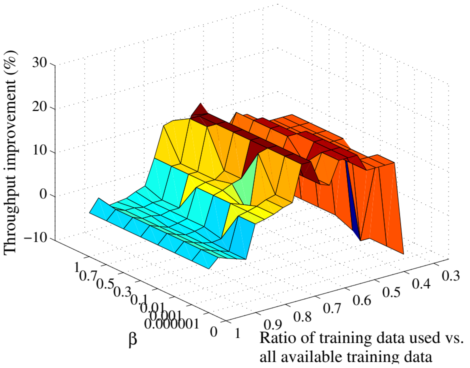

Fig. 5. The throughput gain with different β values.

D. Other Methods to Reduce the Used Training Data

In this section, we demonstrate the usefulness of selecting entries with smaller values of the norm of B. It is well known that the size of the training set can critically affect the performance of classification algorithms. Thus, we construct two baseline methods: (1) randomly select data vectors from the training set, and (2) sort training data according to $\|\mathbf{B}(i,:)\|_2$ in *descending* order, selecting the top-k ranked ones. In other words, our method selects contextual data according to $\|\mathbf{B}(i,:)\|_2$ of smaller values, while the second baseline method selects data in an opposite way of the proposed algorithm.

The accuracy of the rate prediction for the proposed algorithm and the baseline cases are shown in Fig. 6 with $\beta = 0.001$. Clearly, the first baseline method does not improve performance from randomly selected data points, and the accuracy decreases dramatically when the ratio of the training data used is below 0.4. The accuracy with the second baseline method consistently decreases as the ratio decreases. The performance of the proposed method improves first by deleting the noisy data and decreases because of short of training data. The proposed method consistently outperforms the baseline methods, which confirms its effectiveness in identifying outliers.

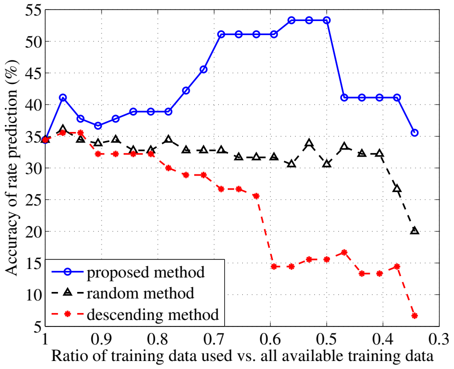

Fig. 6. The performance of rate prediction accuracy with different data point sorting order ($\beta = 0.001$).

E. Evaluation with In-Field Channels

In order to evaluate the proposed outlier detection algorithm with in-field data, we use an experimental setup with two wireless nodes communicating through the air at 2.4 GHz. One of the nodes is located on top of a mobile car, and the other is set on top of a building near our university campus in Dallas, Texas. We collect the throughput data with context information. In this test, 8 transmission modes are available, corresponding to rates $\{6, 9, 12, 18, 24, 36, 48, 54\}$ Mbps. Based on the location and surroundings where the context information is collected, we divide the region around the building into several sub-regions and assume different sub-regions have different channel types.

To generate the training set and testing set, we transmit using each mode for 625 ms in one loop. The mode with the highest throughput in one loop is selected as the optimum mode. We collect measurements in eight sub-regions and prepare a dataset of 123 points, from which we randomly select 20 points for testing. We input the context information of the testing scenario into the classification-based rate adaptation decision structure and check if the output agrees with the best mode we previously inferred.

The training set of 103 points can predict the best mode of testing data with accuracy of 40%. The effects of outlier detection on the accuracy of rate prediction and the improvement

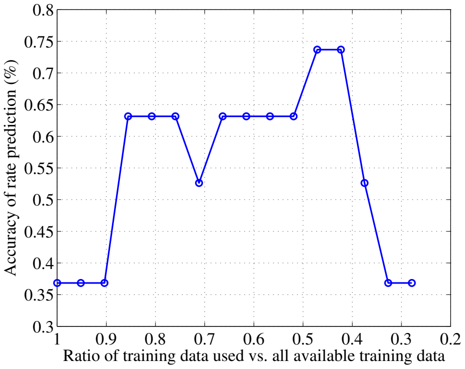

Fig. 7. Rate prediction accuracy with in-field data ($\beta = 0.001$).

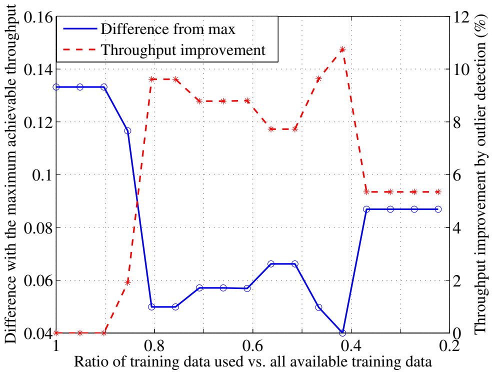

Fig. 8. The throughput gain from outlier detection with in-field data ($\beta = 0.001$).

in throughput are shown in Fig. 7 and Fig. 8, respectively. As the ratio of the used training data decreases, the accuracy and throughput performance of rate prediction has the same behavior as that of our experiments with emulated channels; *i.e.*, the accuracy initially increases with outlier elimination and then decreases because of the lack of training data. When the ratio of the training data used is 47% of the original training set, the accuracy and throughput improvement from outlier detection are 87.5% and 9.60%, respectively. Typically, in-field data contains more errors/outliers than emulated data because of complex in-field transmission interference and measurement inaccuracies. The used training set after outlier elimination achieves high accuracy of rate prediction and can significantly improve the performance of context-aware rate adaptation systems in real-world applications.

IV. Related Work

Context-Aware Wireless Systems. Using context information to improve the performance of wireless networks is a developing field. Prior work has applied machine learning to link adaptation to maximize throughput [5], [6], [19]. In these works, an offline training phase establishes the model connecting the features of the channel and rate related settings. In contrast, we identify the effect of the outlier in context information and propose an algorithm to detect and eliminate outliers in the training set for context-aware applications.

Detecting Imperfect Context Information. A framework has been proposed in prior work that matches the spectral pattern of received signal with the stored reference spectral characteristics of WLAN and microwave devices to detect noise for transmission adaptation [20]. The spectrum characteristics are also used to classify environmental noise after a training process [3].

V. Conclusion

In this work, we present an algorithm for outlier detection of context information. We test the algorithm with a link-level context-aware rate adaptation framework. To perform the outlier detection, we first sort the contextual data based on the confidence of being an outlier and, then we remove the data most likely to be outliers. We evaluate the proposed algorithm both in emulated channels and in-field channels and find that our technique improves the rate prediction accuracy and throughput by up to 87.5% and 18.11%, respectively.

In future work, we will study the threshold and the optimum k for automatic outlier elimination in context information, and examine the tradeoff between the speed of convergence and outlier detection performance.

Acknowledgment

This work was supported by the National Science Foundation under grants CNS-0958436, CNS-1040429, and CNS-1150215.

References

[1] M. Baldauf, S. Dustdar, and F. Rosenberg, “A survey on context-aware systems,” *Journal of Ad Hoc and Ubiquitous Computing*, 2007.

[2] C. Bolchini, C. Curino, E. Quintarelli, F. Schreiber, and L. Tanca, “A data-oriented survey of context models,” *SIGMOD Record*, 2007.

[3] L. Ma, D. Smith, and B. Milner, “Environmental noise classification for context-aware applications,” in *Proc. of DESA*, 2003.

[4] Z. Wang and M. Hassan, “Context-aware channel coordination for DSRC,” in *GLOBECOM Workshops*, 2008.

[5] P. Shankar, T. Nadeem, J. Rosca, and L. Iftode, “CARS: Context-aware rate selection for vehicular networks,” in *Proc. of ICNP*, 2008.

[6] J. He, H. Liu, P. Cui, J. Landon, O. Altintas, R. Vuyyuru, D. Rajan, and J. Camp., “Design and experimentation of context-aware link-level adaptation,” in *Proc. of INFOCOM Mini-Conference*, 2012.

[7] X. Lin, B. Cheng, and J. Chen, “A situation-aware approach for dealing with uncertain context-aware paradigm,” in *Proc. of GLOBECOM*, 2009.

[8] M. Krause and I. Hochstatter, “Challenges in modelling and using quality of context (QoC),” *Mobility Aware Technologies and Applications*, 2005.

[9] T. Sorensen, P. Mogensen, and F. Frederiksen, “Extension of the ITU channel models for wideband (OFDM) systems,” in *Proc. of VTC*, 2005.

[10] J. Tang and H. Liu, “CoSelect: feature selection with instance selection for social media data,” *Proc. Of SDM*, 2013.

[11] J. Tang and H. Liu, “Unsupervised feature selection for linked social media data,” *Proc. Of KDD*, 2012.

[12] S. Boyd and L. Vandenberghe, *Convex optimization*, 2004.

[13] J. Tang and H. Liu, “Feature selection with linked data in social media,” *Proc. Of SDM*, 2012.

[14] Y. Nesterov and I. Nesterov, *Introductory lectures on convex optimization: A basic course*, 2004.

[15] J. Nocedal and S. Wright, *Numerical optimization*, Springer, 1999.

[16] “Aximuth ACE - MIMO channel emulator,” http://www.azimuthsystems.com, 2011.

[17] C. Bishop, *Pattern recognition and machine learning*, 2006.

[18] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2005.

[19] R. Daniels and R. Heath, “Online adaptive modulation and coding with support vector machines,” in *Proc. of EW*, 2010.

[20] K. Chowdhury and I. Akyildiz, “Interferer classification, channel selection and transmission adaptation for wireless sensor networks,” in *Proc. of ICC*, 2009.