A Sketch-based Clustering Algorithm for Uncertain Data Streams

Jingyu Chen
School of Computer Science and Technology, Xidian University, No. 2 South Taibai Road, Xi’an, China
Email: jychen@mail.xidian.edu.cn

Ping Chen
Software Engineering Institute, Xidian University, No. 2 South Taibai Road, Xi’an, China
Email: chenping@mail.xidian.edu.cn

Xian’gang Sheng
College of Information Engineering, Qingdao University, Qingdao, Shandong, China
Email: stzhibuwu@qqdu.edu.cn

Abstract—Due to the inaccuracy and noisy, uncertainty is inherent in time series streams, and increases the complexity of streams clustering. For the continuous arriving and massive data size, efficient data storage is a crucial task for clustering uncertain data streams. With hash-compressed structure, an extended uncertain sketch and update strategy are proposed to store uncertain data streams. And based on divergence and sketch metric, a sketch based similarity is given to measure objects distances. Then with core-sets and the max-min cluster distance measure, an initial cluster centers selection algorithm is proposed to improve the quality of clustering uncertain time series streams. Finally, the effectiveness of the proposed clustering algorithm is illustrated through the experimental results.

Index Terms—Sketch, Divergence, Clustering, Uncertainty, Data Streams

I. INTRODUCTION

With the development of sensor and internet technique, data stream, as a new form of data, widely exists in many applications such as: Web log analysis, network traffic monitoring, real-time traffic monitoring, etc. Many applications need to analyze and mine the massive, continuous and rapid arriving data streams.

For the continuous arriving and massive data size, data-stream processing algorithms often rely on concise, approximate sketch synopses of the data streams [1]. Sketch can provide approximate query and analysis results in real time way, and largely save the storage space. As the data in streams continuous arriving, the sketch must be update to represent the most recent data. For clustering data streams on the sketch, it is very important to measure distances in the sketch model.

In many applications, uncertainty often exists because of network failure, noise and sampling error, etc. Probabilistic observation data are common in uncertain data streams, and the probabilistic information should be stored for clustering. The probabilistic values of observation data are very important for clustering uncertain data streams, and should be reflected in the similarity calculation of clustering.

To address the problem of clustering uncertain data streams, we suggest a hash based sketch approach to reduce storage spaces and computational complexity. We use hash technique to compress data of objects to extended uncertain sketches, and update the data in sketches. Then, based on the KL divergence [2] and sketch *-metric [3], we present our sketch based similarity for objects distances measure. And then according the average probabilities value in sketches, we map objects to different core-sets. Based on core-sets, we measure the similarity of objects and select represent object for each core-set. By using the max-min cluster distance measure, the initial cluster centers selection algorithm is proposed to improve the quality of clustering. Based on outlier handling and UK-means [4], we design an algorithm to cluster uncertain data streams. Final, we verify the accuracy and efficiency of the proposed scheme via experiments.

The rest of the paper is organized as follows. Section II discusses related work. Section III presents an extended uncertain sketch model and divergence based similarity for uncertain data streams. Section IV outlines the core-sets based outliers handling and clustering methods. Simulation methodology and performance evaluation result and analysis are presented in section V and we conclude the work in section VI.

II. RELATED WORKS

Mining data streams has been attracting much attention in research and practice. Nowadays, the study about clustering data streams is mainly about similarity metrics. Uncertainty brings new challenges to clustering, since it increases the complexity of the measurement of similarity between uncertain data objects.

To handle the continuous arriving uncertain data stream, it is a common way to expand data streams mining algorithms through handling the uncertainty of...

Sketch is a popular method for handling huge and fast data streams. Sketch techniques use a sketch vector as a data structure to store the streaming data compactly in a small-memory footprint. The main advantage of using these sketch techniques [12, 13] is that they require a storage which is significantly smaller than the input stream length. Sketch techniques are used in stream data frequent items mining [14], clustering [15] and anomaly detection [16] recently. Papapetrou et al. present a novel sketching technique ECM-sketch that allows effective summarization of distributed data streams over sliding windows with probabilistic accuracy guarantees [16]. For measuring the distance between updatable sketches, Anceaume et al. present a novel metric Sketch*-metric that reflects the relationships between any two discrete probability distributions in the massive data streams [3].

Similarity between two probability distributions can be measured by the Kullback-Leibler divergence (KL divergence). Ackermann et al. [17] develop an approximation algorithm for the k-medoids problem with respect to an arbitrary similarity measure, such as squared Euclidean distance, KL divergence, Mahalanobis distance, etc. Banerjee et al. theoretically analyze the k-means based on Bregman divergences which is a general case of KL divergence [18]. Jiang et al. adopt the KL divergence to measure similarity between uncertain objects in both the continuous and discrete cases for clustering uncertain objects [2].

Our work is closely related to cluster uncertain data streams based on the sketches. In this paper, we construct hash-compressed representations for storing and analyzing the probabilities of uncertain data streams, and design an update strategy for sketches as sliding window changing. And, based on the KL divergence and Sketch*-metric, we present a similarity measurement method to calculate the distances between uncertain objects. In order to reduce the computation, we construct core-sets and present an initial cluster centers select algorithm to optimize clustering quality. As outliers may affect the quality of clustering results, we adopt the Local Distance-based Outlier Factor (LDOF) [19] to check outliers for core-sets.

III. DIVERGENCE BASED SIMILARITY

Uncertainty is widely present in data streams, and probabilistic data are common form in uncertain data streams. In many applications, such as sensors, traffic, stock, etc., there are many observation objects in the uncertain data streams. The uncertainty of the observation values may increase the complexity for analyzing these probabilistic data streams.

The probabilistic data streams may include $N$ uncertain objects $O_N = \{o_1, o_2, \ldots o_N\}$. The data set of an uncertain object can be described as a set of observation values, probabilities and times.

Definition 1 (Element) An element $e$ of an object $o_i$ is a basic probability data of the object in uncertain data streams $e = \{x, p(x), t\}$, where $x$ represents an observation value, $p$ represents the probability value of $x$, and $t$ represents observation time. We also use $p(x)$ to represent the probability value $p_i$ of $x$ at time $t$.

Definition 2 (Uncertain Item) An uncertain item $I(o_i)$ of an uncertain object $o_i$ at time $t$ includes all data at time $t$. $I(o_i) = \{(x_1, p_1, t_1), (x_2, p_2, t_2), \ldots\}$.

A. Extended Uncertain Sketch

For data streams, the massive data size will increase the computational and storage cost. Therefore, compressed storage is a common technique in data streams. Sketch is an efficient technique for compressed storing data streams. However, for the fast arriving and continuous varying, the sketch may be saturated and should be updated to represent most recent data. In [1], an ECM-sketch is proposed to summary data streams in sliding window model. Based on the ECM-sketch, we construct the extended uncertain sketch EU-sketch to store uncertain data streams in compression.

We construct an EU-sketch for each uncertain object to store the observation values of the object. Like the ECM-sketch, the core structure of EU-sketch is a modified Count-Min sketch. Each counter of EU-sketch is a bucket to store the probability value and the observation time.

The EU-sketch consists of a two dimensional array with $w$-c cells (counters) with a length of $h$ and width of $c$. As the count-min sketch, EU-sketch use $w$ pairwise independent hash functions $\{hr_0, \ldots, hr_{w-1}\}$, each hash function $hr_i(x): x \rightarrow \{0, \ldots, c\}$ maps data into uniformly random integer. Let $hr_i$ be a 2-universal hash function that for every two different items $i$ and $j$, the probability of collision $Pr\{hr(i) = hr(j)\} \leq 1/c$, is smaller than $1/c$.

Figure 1. Adding an element to the EU-sketch

Figure 1 shows the process of adding an element to a EU-sketch. For a new arriving element $e = \{x, p(x), t\}$, the EU-sketch adds $(p(x), t)$ to the $hr(x)$-th cell at the each $i$-th row. Since the each cell of EU-sketch is a bucket, as the probability value and time are added to the bucket.
continuously, we should remove the expired data in a bucket.

For the continuous arriving data, sliding windows are common technique to handle the most recent data. As sliding windows move continuously, the old data stored in buckets should be deleted timely.

In order to remove the expired data more efficiently, we should choose a time point $t'$. The time point $t'$ divides a bucket into two parts $A$ and $B$, as showed in figure 2. In figure 2, all data in part $A$ are added to the bucket before the time $t'$, and all data in part $B$ are added after the time $t'$.

![A bucket and the time point $t'$](image)

In order to improve the utilization of the bucket space, we will delete all the data in part $A$. We wish the deletion of $A$ will not affect the clustering results. Therefore, the time point $t'$ should be selected according the data size in part $A$ and $B$, to make the data in part $A$ and $B$ have much more amount than data in part $B$.

As sliding windows change, for a full bucket, we set $t = t_0 + 0.5 \times (t_n - t_0)$. If the amount of data in part $A$ is larger than amount in part $B$, delete all data in part $A$. Otherwise, let $t = t_0 + 4 \log (t_n - t_0)$, and delete all data in part $A$.

Through the bucket removal policy, we can remove the expired data and improve the quality of clustering results.

### B. Divergence-based Similarity

The uncertainty is a very important feature in uncertain data stream, and the probability value will impact quality of clustering results and should be reflected in the similarity.

In order to capture distribution difference between objects, we use KL divergence [2] to measure the statistical difference between two objects in data streams.

**Definition 3 (Kullback-Leibler divergence) The KL divergence is a robust metric for measuring the statistical difference between two data streams.**

Given $p$ and $q$ two distributions in discrete domain $\Omega$ with a finite number of values, the Kullback-Leibler divergence between $p$ and $q$ is then defined as:

$$D(p \| q) = \sum_{i \in \Omega} p_i \log \frac{p_i}{q_i}. \tag{1}$$

For uncertain data streams, the KL divergence can be used as a metric for measuring the similarity between two objects $o_1$ and $o_2$.

Since the data of objects are stored in the EU-sketches, the computation of similarity should be designed based on the EU-sketch. We combine the Sketch $*$-metric [3] and the KL divergence [2] to quantify the similarity between two uncertain objects.

**Definition 4 (Sketch-metric) Let $p$ and $q$ be any two $\Omega$-point distributions. Given a precision parameter $k$, and any generalized metric $\phi$ on the set of all $\Omega$-point distributions, there exists a Sketch-metric $\phi^*$ defined as follows:**

$$\phi^*_\rho = \max_{\rho \in \hat{\mathcal{P}}_{\|q\|}} (\phi(\hat{p}_\rho || \hat{q}_\rho)), \forall \rho \in \rho, \hat{p}_\rho(a) = \sum_{i \in \mathcal{P}} p(i) \tag{2}$$

where $\rho(\Omega)$ is the set of all partitions of an $\Omega$-element set into exactly $k$ nonempty and mutually exclusive cells.

To compute the Sketch-metric of two objects $p$ and $q$, two EU-sketches $UES_p$ and $UES_q$ are constructed. As the sketch $*$-metric algorithm defined in [3], each line of the EU-sketch corresponds to a $\rho$ partition, $1 \leq \rho \leq c$. Thus the similarity between two objects is the maximal value over all the $c$ partitions $\rho_i$ of the distance metric $\phi$. We apply the KL divergence as the distance metric $\phi$ to the $i$-th lines of the two EU-sketches $UES_p$ and $UES_q$, $1 \leq i \leq c$. The similarity of two uncertain objects $p$ and $q$ can be defined as

$$SD(p \| q) = \max (D (\hat{p}_\rho || \hat{q}_\rho)) \tag{3}$$

where $\hat{p}_\rho = UES[i], 1 \leq i \leq c$, and $D$ is the KL divergence defined in formula (1).

Based on the KL divergence and the sketch $*$-metric, the probability distribution of uncertain objects are considered in the similarity of two uncertain objects. The similarity $SD$ of two uncertain objects can also be regarded as the distance of two objects.

### IV. CORESET BASED UNCERTAIN CLUSTERING

UK-means is a known cluster algorithm based on K-means and is used on uncertain streams. According our analysis, the time complexity of UK-means is mainly due to the large number calculations of expected distance. Therefore, we use a partitions policy to reduce the expected cost to the set $C$.

**Definition 5 (Expected cost) Expected cost $Ecost(C, o_i)$ is a distance metric between a set $C$ of objects and an object $o_i$. $Ecost(C, o_i)$ is the sum of the expected distances between $o_i$ and all objects in $C$.**

$$Ecost(C, o_i) = \frac{1}{|C|} \sum_{o_j \in C} SD(o_j, o_i) \tag{4}$$

where $|C|$ is the number of objects in $C$.

**Definition 4(Represent object) A represent object of an objects set $C$ is the object $c_p$ that has the minimum expected cost to the set $C$.**

**Definition 6 (Core-set) A Core-set includes three parts: a set of uncertain objects $C$, the represent object $c_p$, and the expected cost $Ecost(C, c_p)$.**

To construct core-sets, we should partition all $N$ objects into different $m$ core-sets, and make the objects in same core-set as similar as possible. The $m$ is the number of core-set, and can be defined by user.

Since we calculate the similarity based on the EU-sketches, we also partition objects according the EU-
sketches. We count the average probabilities value of each line in the EU-sketch of an object, and get the value range \( ra \) of average probabilities value. For two objects, if the difference of average probabilities value is smaller than a threshold value \( \alpha \), the two objects should be put into a same core-set. Based on average probabilities value of each object, we partition all the \( N \) objects into \( m \) core-sets. Then, for each core-set \( CS_i \), we randomly select \( k = \log(|CS_i|) \) objects as \( k \) candidate represent objects from \( CS_i \). For a candidate object \( o_i \), we calculate the expected cost \( Ecost(CS_i, o_i) \) and find a candidate object that has the minimum expected cost to the core-set \( CS_i \).

The method of core-sets construction is shown as following.

```
Method of core-set construction

Input:
1. \( N \) EU-sketches of \( N \) uncertain objects
2. \( m \): the number of core-sets

Output:
\( m \) core-sets with \( m \) represent objects

End For
End while

For each object \( q \) and its EU-sketch \( UES_q \)
    Avg_q = Average(UES_q[0],...,UES_q[w-1])
End For

Get the minimal Average \( min\_Avg \) of all objects
Get the maximal Average \( max\_Avg \) of all objects

\( \gamma = \max(\text{max}_-\text{Avg}, \text{min}_-\text{Avg})/m \)

For each object \( q \) and its \( Avg_q \)
    For each core-set \( cs_i \)
        Get the \( Avg \) of \( cs_i \)
        If \( |Avg_i - Avg_q| < \gamma \)
            Add the object \( q \) to the core-set \( cs_i \)
        End If
    End For

For each object \( o_i \) in \( CS_i \)
    Add \( o_i \) to \( CS_i \)
End For

\( E_{id} = SD(o_i, o_j)/\alpha \)

Min_object(SD(o_i, o_j)) = \( o_j \)

End For

Output:
\( m \) core-sets \( CS_i \) and \( m \) represent objects \( RO \)
\( \alpha \): the threshold value

End While
```

We use \( m \) core-sets and represent objects to improve the efficient of clustering. For massive objects, the average number of objects in a core-set \( |CS| = 200k \) will get much better clustering results according [5].

**B. Initial Cluster Centers Selection**

The clustering results of UK-means largely depend on the selection of initial cluster centers. If the initial cluster centers are selected unreasonably, the clustering result is likely to be local minima.

To choose the initial cluster centers, we design a selection policy based on a max-min cluster distance algorithm to calculate distances among the objects in the core-set. The main idea of max-min cluster distance algorithm is to select maximum distance among the other represent objects of core-sets and minimal distance among other cluster centers.

The max-min cluster distance based initial cluster centers selection algorithm is shown as following.

First, the initial cluster centers selection algorithm randomly selected an object \( o_1 \) from \( m \) represent objects and set \( o_1 \) as the first initial cluster center \( ic_1 \). Then select a represent object \( o_2 \) which has the maximum distance with \( ic_1 \) as the second initial cluster center \( ic_2 \). Set the distance between \( ic_1 \) and \( ic_2 \) as an expected initial distance \( E_{id} \).

For other represents object \( o_i \), if \( \max(min(SD(o_i, ic_1), SD(o_i, ic_2),...,SD(o_i, ic_j))) > \alpha*average(E_{id}) \)

select \( o_i \) as \( ic_j \). If no object meets the test condition, then finish the algorithm.

```
Algorithm of initial cluster centers selection

Input:
\( m \) core-sets \( CS \) and \( m \) represent objects \( RO \)
\( \alpha \): the threshold value

Output:
a set of initial clusters centers \( IC \)

Randomly select a represent object \( ro_1 \)
Set \( ro_1 \) to the first initial clusters center \( ic_1 \) of \( IC \)
For each object \( ro_i \) in the set \( RO \) of represent objects
    Max_object(SD(ro_i, ic_j)) = ic_j
End For
Let \( SD(ic_1, ic_2) \) be an expected initial distance \( E_{id} \)
\( j = 2 \)
while(true)
    For each other objects \( o_i \) in \( RO \)
        Min_object(SD(o_i, ic_j)) = \( o_j \)
    End For
    m\( k = |MS| \)
    For each other objects \( m_o \) in \( MS \)
        if(Max(SD(o_i, m_o)) < \( \alpha*average(E_{id}) \))
            set \( m_o \rightarrow ic_1 \)
            \( j++ \)
        E_{id} = SD(ic_{j-1}, ic_j)
        else
            mk--
            end if
        End For
    End While

The initial cluster centers selection algorithm selects centers based on the test conditions \( \alpha*average(E_{id}) \)

The expected initial distance \( E_{id} \) is the distance \( SD(ic_j, ic_{j-1}) \) between the last two initial cluster centers \( ic_{j-1} \) and \( ic_j \). The threshold value \( \alpha \) is very important for the number of initial cluster centers. The smaller value is the test parameter \( \alpha \), the more initial cluster centers will generate. The threshold value \( \alpha \) should be choose properly.

**C. Handling Outliers**

In the construction of core-sets, since outliers may be selected as represent objects, these outliers will affect the quality of the clustering results. We must check the outliers and avoid to selecting them as the represent objects.

For a core-set, we select the represent object according the LDOF [19] among \( k \) candidate objects set \( KC \).

Definition 7 (KC distance) The KC distance of a candidate object \( co_0 \) is the average distance from \( co_0 \) to other \( k-1 \) candidate objects of the \( k \) candidate objects set \( KC \).

\[
d_{co} = \frac{1}{k-1} \sum_{o_k \in KC} SD(co_0, co_k) \quad (5)
\]

Definition 7 (KC inner distance) The KC inner distance is the average distance among objects in \( k \) candidate objects set \( KC \).
The local distance-based outlier factor of a candidate object \( c_0 \) is defined as:

\[
LDOF(c_0) = \frac{d_{ro}}{D_{KC}} \tag{7}
\]

Based on the LDOF, we change the represent objects selection method of core-sets through change candidate objects for each core-set. Then, For a candidate object \( c_0 \) of core-set \( C_s \), if the LDOF \( (c_0) > 1 \), we delete the candidate object \( c_0 \) and random select other object \( c_0 \) with LDOF \( (c_0) < 1 \) in the core-set \( C_s \) as a new candidate object. And we select the candidate object \( ro_i \) with LDOF \( (ro) > 1/2 \) as the represent object from the candidate objects for each core-set.

D. UESStream Clustering Algorithm

Based on the selection of initial cluster centers and UK-means algorithm, we build our UESStream algorithm for clustering uncertain data streams.

The UESStream algorithm includes four main phases.

1) Construct EU-sketch \( UES \) for an object \( p \). Each counter in EU-sketch stores probabilities value and time of an uncertain object, and delete the expired data according the time point.

2) Construct core-sets. Based on the core-set construction policy, partition objects to different core-sets and select represent object for each core-set with LDOF.

3) Select initial cluster centers. Based on max-min distances, UESStream selects initial cluster centers from represent objects.

4) Cluster the uncertain data stream. Based on initial cluster centers, UESStream use each initial cluster center to represent a cluster. As data of objects continuous arriving and the sliding window varying, UESStream calculate the distance between an object and a cluster center. Then, add the object to the nearest cluster, and recalculate the cluster center of the cluster.

The 1) to 3) phases are to construct granules and sequential sketches for compressing storage size. The 3) phase is to select initial cluster centers for improving clustering quality. In the 4) phase, the UESStream algorithm clusters the uncertain stream based on UK-means.

V. EXPERIMENT

This section shows the results from our experiment to validate the accuracy and efficiency of our proposed clustering algorithm.

A. Evaluation Standard

We will show the accuracy of our algorithm based on the evaluation standard of RandIndex [20]. For uncertain data set \( D \) (includes \( N \) objects), let \( T = \{T_1, T_2, ..., T_k\} \) represent the original clusters, and \( C = \{C_1, C_2, ..., C_m\} \) be the clusters produced by a clustering algorithm. Let \( a \) represent an object that is in a cluster of \( C \) and in a cluster of \( T \) either. Let \( b \) represent an object that is in a cluster of \( C \) but NOT in any cluster of \( T \).

\[
SRAND = \frac{a + b}{n(n-1)/2} \tag{8}
\]

The \( SRAND \) represent the degree of matching between \( T \) and \( C \). The greater of the \( SRAND \) value means the better of the clustering.

B. Experimental Results

We compare our UESSstream algorithm to UK-means by using the Census 1990, Tower, and Covertype data sets [21]. For each record \( x \) of an object \( o_i \) in a data set, we add 2 possible data \( \{x_1, x_2\} \) and probability value \( \{p_0, p_1, p_2\} \). \( \{p_0, p_1, p_2\} \) means the probability value of \( \{x, x_1, x_2\} \). According the following formulas, we use two parameters \( \alpha = 0.0.1 \) and \( \beta = 0.05 \) to calculate \( x^1 \) and \( x^2 \).

\[
x^1 = x^1(1+y^\alpha x^2), x^2 = x^2(1+y^\alpha x^2) \tag{9}
\]

where \( y \) is a random value in \([-1, 1]\).

The probability value \( \{p_0, p_1, p_2\} \) can be assigned following the normal distribution from the range \((0, 1)\) to each item in the data set, according the difference among \( \{x, x_1, x_2\} \).

Figure 3 shows the comparison of clustering quality between UESSstream and UK-means with the increase in the size of the Census 1990 data set. In figure 3, the \( SRAND \) value of UESSstream is higher than that of UK-means. Because the UESSstream increase the calculation of initial cluster centers selection to improve the clustering quality. The clustering quality UESSstream is also varies on the size increase of testing data set.

Figure 4 shows the comparison of time consuming between UESSstream and UK-means with the increase in the size of the Census 1990 data set. In figure 4, the time consuming of UESSstream is higher than that of UK-means. It will cost the UESSstream some time to increase the calculation of initial cluster centers selection to improve the clustering quality.

Figure 5 shows the comparison of memory space consuming between UESSstream and UK-means with the increase in the size of the Census 1990 set. In figure 5, the memory space consuming of UESSstream is smaller than that of UK-means. Because the UESSstream adopt
sequential sketch and granules based method to compress storage space.

Experiment results show that the UESStream algorithm can effectively cluster uncertain data streams. The UESStream benefits from using compressed storage space and initial cluster centers.

VI. CONCLUSION

This paper has addressed an extended uncertain sketch EU-sketch and sketch metric method for clustering uncertain data streams. First, for reducing memory consumption, we construct extended uncertain sketches EU-sketches to store continuous arriving uncertain data streams, and update sketches according sliding windows and data variation. And based on divergence and sketch metric, a sketch metric is given to measure the similarity between two uncertain objects. Then, based on the sketch metric, a core-set construction strategy is given to select represent objects and. And then, based on the Max-min cluster distance algorithm, an algorithm for selecting the initial cluster centers is proposed. And with the outliers handling policy, our UESStream algorithm for clustering uncertain data is given. Experiment results show that the UESStream algorithm can cluster uncertain data streams with better accuracy and efficiency. In future work, we plan to design distributed sketches and clustering algorithm for distributed uncertain data streams.

ACKNOWLEDGMENTS

The paper is supported by the Fundamental Research Funds for the Central Universities.

REFERENCES

[8] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander, "Optics: Ordering Points to Identify the


**Jingyu Chen** Born in 1976, she received the B.S. degree from the Southwest University of Science and Technology, in 1998, in computer and its applications, and the M.S. degree from Xidian University, Xi'an, China, in 2004, in computer software and theory.

She now is an associate professor at School of Software, Xidian University. Her research interests include: data streams management and mining, p2p data management and data integration.

**Ping Chen** Born in 1953, he received the the B.S. and Ph.D. degrees from the Xidian University, Xi'an, China, in Software Engineering.

Currently, he is a Professor and supervisor of a Ph.D. student at School of computer science and technology, Xidian University. His research interests include: software architecture technology and application, dynamic object technology, and reverse engineering.

**Xian’gang Sheng** Born in 1976, he received the B.S. degree from the Naval University of Engineering, PLA, in 1998, in wire communication, and the M.S. degree from Xidian University, Xi’an, China, in 2006, in communication and information system.

He now is a Lecturer at Computer Education Center, Qingdao University. His research interests include: multimedia communication, mobile computing, and XML storage technology.