CHAPTER VI

3STSC: SHAPE-BASED STREAMING SUBSEQUENCE TIME SERIES CLUSTERING

In time series domain, streaming clustering algorithms are divided into two categories, i.e., streaming whole clustering (Rodrigues et al., 2006, 2008) and streaming subsequence clustering. For the streaming whole clustering, the new whole sequence is used to update the clustering result or cluster representatives, while for the streaming subsequence clustering, after the new data point is concatenated, a subsequence is extracted from a fixed-length sliding window, subsequence is normalized, and then the cluster representatives are updated from this subsequence. The naïve algorithm of the streaming problem is that the output of the algorithm is calculated from all previous input subsequences for every new incoming sequence. In this chapter, the streaming subsequence clustering is focused.

As shown in Chapter II, Keogh and Lin have proved that outputs from Subsequence Time Series Clustering (STSC) are meaningless; therefore, currently, no meaningful naïve algorithm for streaming clustering algorithm exists. In Chapter IV, 2STSC is proposed to return a meaningful clustering result, where Dynamic Time Warping (DTW) distance and Shape-based Averaging function are used as a distance measure and an averaging function instead of Euclidean distance and Amplitude Averaging function as in STSC, respectively. In this chapter, 2STSC is considered as a naïve algorithm of a streaming application. Since 2STSC calculates a clustering result from all previous subsequences, it is impractical since the computational time depends on the number of previous subsequences which increases over time.

In this chapter, Streaming Shape-based Subsequence Time Series Clustering (3STSC) is proposed to efficiently update the clustering result in constant time to the number of previous subsequences. Instead of calculating the clustering result from all previous subsequences as in 2STSC, 3STSC calculates the clustering result from the small number of stored subsequences. The algorithm of updating stored subsequences in 3STSC is the same as that of the Incremental Shape-based Averaging , where the number of stored subsequences is maintained not to exceed the maximum allowance number of stored subsequences. 3STSC then groups these stored subsequences into clusters using k-hierarchical clustering with Dynamic Time Warping (DTW) distance and Shape-based Averaging function as a distance measure and an averaging function. In other words, 3STSC returns a clustering result from a small set of stored subsequence which is much

faster than 2STSC which returns a clustering result from all previous subsequences.

In experimental evaluation, 3STSC shows superiority over 2STSC in terms of computational time, and the clustering result of 3STSC is also compared in terms of Shape-based Meaningfulness Measurement (SMM) when the parameters, i.e., the number of clusters (k), the length of sliding window (w), and the maximum allowance number of stored subsequences (α), are varied.

6.1 Related Work

Clustering time series data streams is divided into two categories, i.e., streaming whole clustering and streaming subsequence clustering. Streaming whole clustering is an incremental clustering, where a whole time series sequence arrives constantly. No sliding windows are involved in the algorithm. A new arriving whole sequence is used to update a clustering structure such as a tree of hierarchical clustering. Rodrigues et al. have proposed Online Divisive Agglomerative Clustering (ODAC) (Rodrigues et al., 2008) for time series data streams which implements splitting and merging operations for updating a tree-like hierarchy of clusters that do not depend on the number of data objects in the data stream. For streaming subsequence clustering, a set of clusters is returned for every incoming data point. However, no existing algorithm has been proposed yet. Although many subsequence clustering algorithms are proposed such as Density-based Subsequence Clustering (DSTSC) (Denton, 2005), Lag-based Subsequence Time Series Clustering (LSTSC) (Simon et al., 2006; Chen, 2007b,a), and Phrase-Analysis Subsequence Time Series Clustering (PASTSC) (Fujimaki et al., 2008), no extension of streaming applications has been introduced. In addition, as mentioned in Chapter IV, these subsequence clustering algorithms still do not produce meaningful clustering results.

Many problems on time series data streams such as subsequence matching, motif discovery, and stream monitoring have been increasingly the topics of interest. For subsequence matching (Sakurai et al., 2005; Niennattrakul and Ratanamahatana, 2009; Niennattrakul et al., 2009), a template query is given and a set of nearest subsequences is returned. Motif discovery for data streams (Mueen and Keogh, 2010) is a method to maintain the best-matched subsequence pair in a given time series sequence. Stream monitoring (Kontaki et al., 2008; Dai et al., 2006) is a method to find correlations among data streams.

In this study, streaming subsequence clustering is considered the first streaming subsequence clustering algorithm that produces meaningful clustering results.

6.2 Shape-based Streaming Subsequence Time Series Clustering

Shape-based Streaming Subsequence Time Series Clustering (3STSC) is an incremental subsequence clustering algorithm that returns a set of cluster representatives for every new data point arrival. Specifically, 3STSC first concatenates a new data point with the previous time series sequence. A new subsequence is then extracted with a fixed-length sliding window, and then the subsequence is normalized by z-normalization. Since the maximum allowance number of stored subsequences needs to be maintained, the set of stored subsequences is updated by a new sequence not to exceed the maximum allowance number. After the set of stored subsequences is updated, 3STSC then finds a clustering result using k-hierarchical clustering on these stored subsequences with Dynamic Time Warping (DTW) distance and Shape-based Averaging function as a distance measure and an averaging function, respectively. Additionally, the updating algorithm of the stored subsequences is similar to the Incremental Shape-based Averaging function. Note that the maximum allowance number of the stored subsequences is a user-defined parameter depending on the availability of computational power and storage. The overview of 3STSC is provided in Figure 6.1.

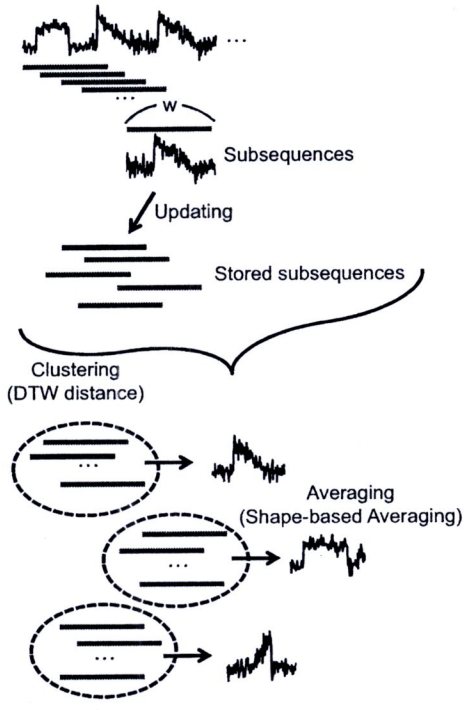

Figure 6.1: Overview of Shape-based Streaming Subsequence Time Series Clustering (3STSC).

Given a new data point s_t, the number of clusters k, the length of sliding window w, and the maximum allowance α of stored subsequences, 3STSC returns a set $\mathbb{C} = \{C_1, C_2, \ldots, C_k\}$ of clusters. 3STSC first concatenates s_t to a streaming time series $S = \langle s_1, s_2, \ldots, s_{t-1} \rangle$, and then a new subsequence $\mathcal{S} = \langle s_{t-w+1}, \ldots, s_{t-1}, s_t \rangle$ is extracted with the fixed-length sliding window

of length n. In addition, this new subsequence $\mathcal{S}_{norm}$ is normalized by z-normalization. 3STSC updates a set $\mathbb{T} = \{T_1, T_2, \ldots, T_\alpha\}$ of stored subsequences using an updating algorithm from Incremental Shape-based Averaging. After the set $\mathbb{T}$ is updated, subsequences in the set $\mathbb{T}$ are clustered and return a set $\mathbb{C} = \{C_1, C_2, \ldots, C_k\}$ of clusters using k-hierarchical clustering with Dynamic Time Warping (DTW) distance and Shape-based Averaging is returned. Each cluster C contains a set $\mathbb{M} = \{T_i \mid T_i \in \mathbb{T}\}$ of stored subsequences and a cluster representative R. The pseudo code of 3STSC is provided in Table 6.1.

Table 6.1: Pseudo code of Shape-based Streaming Subsequence Time Series Clustering (3STSC)

FUNCTION [$\mathbb{C}$] = 3STSC [$\mathbb{T}$, $\mathbb{W}$ s_t, k, w, α]
1. Update a streaming time series S by adding a new arriving data point s_t
2. $\mathcal{S}$ = EXTRACTLASTESTSUBSEQUENCE(S, w)
3. $\mathcal{S}_{norm}$ = ZNORMALIZE($\mathcal{S}$)
4. $\mathbb{T}$ = UPDATESTOREDSUBSEQUENCE($\mathbb{T}$, $\mathbb{W}$, $\mathcal{S}_{norm}$,α)
5. $\mathbb{C}$ = KHIERARCHICALCLUSTERING($\mathbb{T}$, k)
6. Return $\mathbb{C}$

K-hierarchical clustering used in 3STSC can be used with either complete linkage or average linkage function as an inter-cluster distance function which calculates the distance between two clusters defined as the following equations.

$$D_{complete}(C_i, C_j) = \max_{\mathcal{S} \in \mathbb{M}_i, \mathcal{S}' \in \mathbb{M}_j} Distance(\mathcal{S}, \mathcal{S}') \tag{6.1}$$

$$D_{average}(C_i, C_j) = \frac{1}{|\mathbb{M}_i|\,|\mathbb{M}_j|} \sum_{c \in C_i} \sum_{c' \in C_j} Distance(\mathcal{S}, \mathcal{S}') \tag{6.2}$$

where $D_{complete}$ and $D_{average}$ are complete and average linkage functions, respectively, C_i and C_j are any clusters, $\mathbb{M}_i$ and $\mathbb{M}_j$ are cluster members of C_i and C_j, respectively, and $\mathcal{S}$ and $\mathcal{S}'$ are sequences in $\mathbb{M}_i$ and $\mathbb{M}_j$, respectively. $Distance(\mathcal{S}, \mathcal{S}')$ returns a DTW distance between two sequences $\mathcal{S}$ and $\mathcal{S}'$.

To update stored subsequences, 3STSC utilizes the updating algorithm that is similar to Incremental Shape-based Averaging , where the number of stored subsequences is maintained not to exceed the maximum allowance number (α). Specifically, the smallest possible number of the maximum allowance number (α) is equal to the number of clusters (k). When a new subsequence $\mathcal{S}_{norm}$ arrives, the nearest stored subsequence T_{Best} to the new subsequence $\mathcal{S}_{norm}$ is averaged and the nearest stored subsequence T_{Best} is replaced with the averaged result, where the weight of the averaged result is increased by one. Pseudo code of the updating algorithm is provided in

Table 6.2.

Table 6.2: Updating stored sequences in 3STSC

FUNCTION [$\mathbb{T}$, $\mathbb{W}$] = UPDATESTOREDSEQUENCES [$\mathbb{T}$, $\mathbb{W}$, $\mathcal{S}_{norm}$, α]

1. Let n be a number of stored sequences in $\mathbb{T}$
2. If ($n < \alpha$)
3. Add $\mathcal{S}_{norm}$ in $\mathbb{T}$
4. Add $w = 1$ in $\mathbb{W}$
5. Else
6. $dist_{Best}$ = INFINITY
7. For each stored sequence T_i in $\mathbb{T}$
8. $dist$ = DTW-DISTANCE(T_i, S)
9. If ($dist < dist_{Best}$)
10. $dist_{Best} = dist$
11. $T_{Best} = T_i$
12. $w_{Best} = w_i$
13. End if
14. End for
15. $\mathcal{S}_{avg}$ = AVERAGINGFUNCTION(T_{Best}, $\mathcal{S}_{norm}$, w_{Best}, 1)
16. Replace T_{Best} with $\mathcal{S}_{avg}$
17. Replace w_{Best} with $w_{Best} + 1$
18. End If
19. Return [$\mathbb{T}$, $\mathbb{W}$]

Note that 2STSC is a special case of 3STSC when the maximum allowance number of stored subsequences (α) is set to positive infinity.

6.3 Experimental Evaluation

Shape-based Streaming Subsequence Time Series Clustering (3STSC) is proposed to find a set of cluster representatives incrementally. 3STSC is evaluated in two experiments. The first experiment shows speedup of 3STSC over 2STSC, where 3STSC updates a cluster representative for every new incoming sequence in constant time, but 2STSC recalculates a set of cluster representatives in every new incoming sequence. Since the result of 2STSC and 3STSC are not the same due to the incremental algorithm of 3STSC, the second experiment demonstrates the difference of clustering results between 2STSC and 3STSC. The last experiment shows that if computational power and memory storage are available, the clustering result of 3STSC will be close to that of 2STSC. Eight datasets used in these experiments are from the Time Series Data Mining Archives (TSDMA) (Keogh and Folias, 2011) shown in A.1 in Appendix A, where each dataset contains 2000 data points. Two examples of each dataset are provided in Figure 6.2.

6.3.1 First Experiment

The first experiment shows that 3STSC can return a set of clusters much faster than the naïve algorithm using 2STSC. At every new incoming data point, time to update cluster repre-

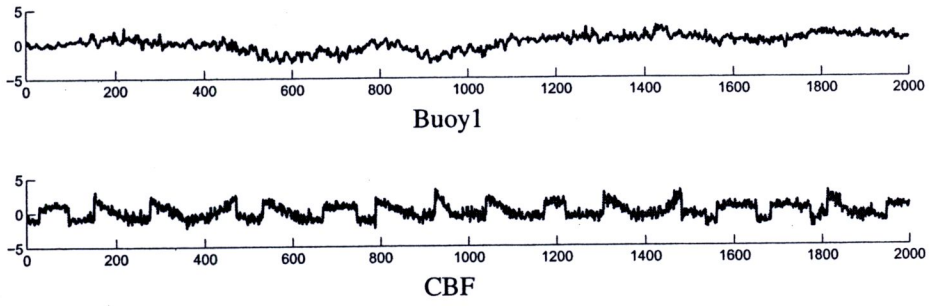

Figure 6.2: Some datasets from TSDMA used in the experiment.

sentatives of 3STSC and the naïve algorithm are captured. The number of clusters (k) and the sliding window (w) are varied, and the maximum allowance number (α) is set to be the number of clusters. In this experiment, two inter-cluster distances of k-hierarchical clustering, i.e., complete linkage and average linkage functions, and two averaging functions, i.e., CDTW and ICDTW, are utilized. Figures 6.3 and 6.4 show the computational time of between 3STSC and the naïve 2STSC algorithm when $k = 3$ and $w = 64$. The complete results are provided in Appendix G.

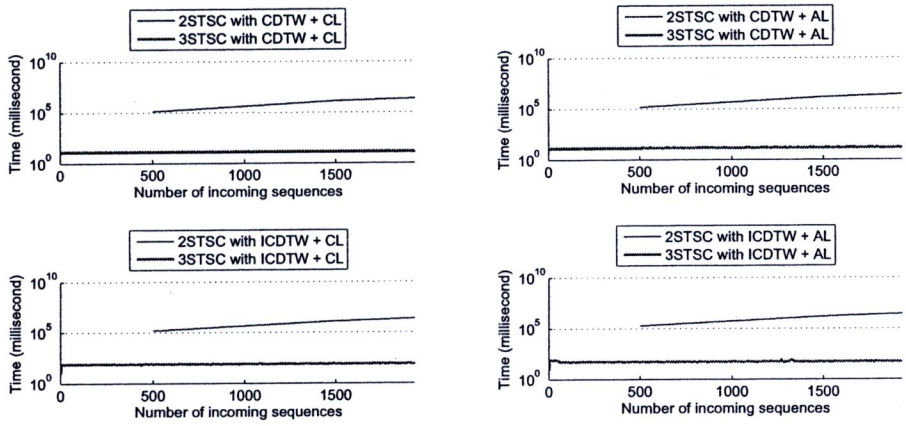

Figure 6.3: Computational time of 3STSC and 2STSC of Buoy1 when a new incoming sequence arrives.

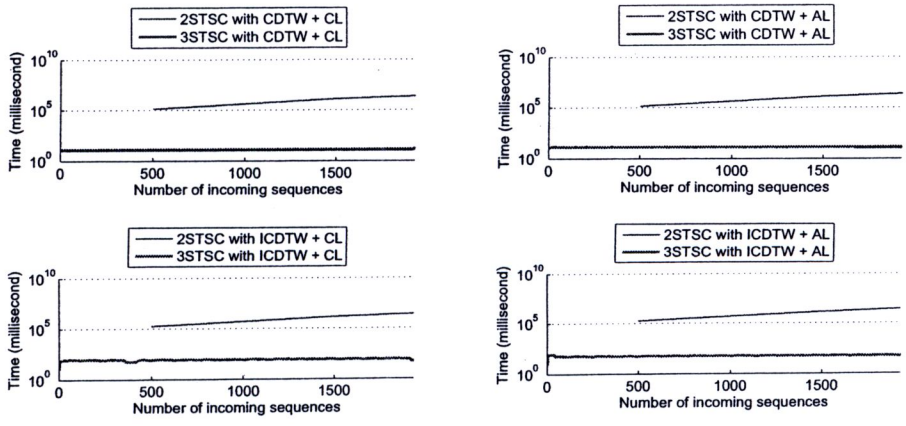

Figure 6.4: Computational time of 3STSC and 2STSC of CBF when a new incoming sequence arrives.

6.3.2 Second Experiment

The second experiment shows the quality of clustering results generated from 3STSC, when the maximum allowance number (α) is varied; the quality of clustering results increases when there is availability of computational power and storage. However, the quality of clustering results is a tradeoff to clustering time; the number of clusters (k) and the sliding window (w) are varied, and the maximum allowance number (α) are also varied to show speedup and clustering quality. The clustering quality is measured by Shape-based Meaningfulness Measurement (SMM) proposed in Chapter IV, which can be calculated from the following equation.

$$SMM\left(S, \mathbb{C}\right) = \frac{|\mathbb{S}| \cdot w}{\sum_{i=1}^{|\mathbb{S}|} \min\left(Distance\left(\mathcal{S}_i, R_j\right)\right), \forall R_j \in \mathbb{R}} \tag{6.3}$$

where $Distance\left(\mathcal{S}_i, R_j\right)$ is a DTW distance between two sequences $\mathcal{S}_i$ and R_j.

SMM ranges from zero to positive infinity and is a relative value that SMM must be compared between two algorithms at the same set of parameters to identify that with a given dataset which subsequence clustering algorithm produces more meaningful clustering results.

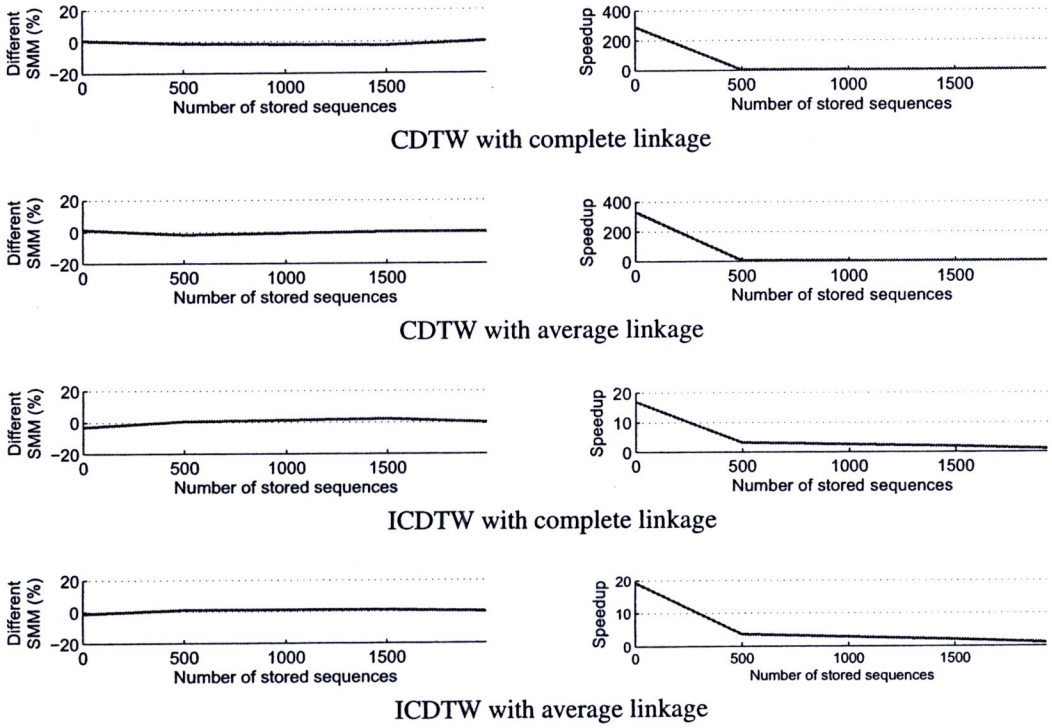

Figure 6.5: Percentage difference of SMM and speedup of 3STSC of Buoy1 when k = 3, w = 64, and number of stored sequences are varied.

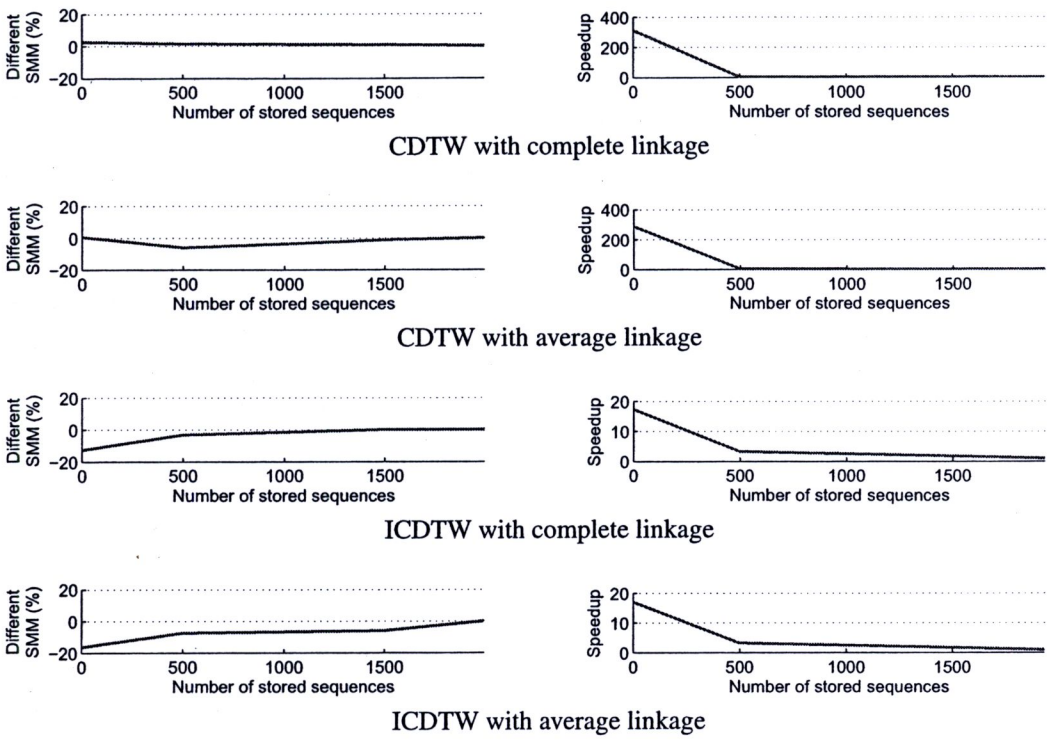

Figure 6.6: Percentage difference of SMM and speedup of 3STSC of CBF when $k = 3$, $w = 64$, and number of stored sequences are varied.

In this experiment, two inter-cluster distances of k-hierarchical clustering, i.e., complete linkage and average linkage functions, and two averaging functions, i.e., CDTW and ICDTW, are utilized. Figures 6.5 and 6.6 show SMM difference and the computational time of 3STSC of Buoy1 and CBF when inter-cluster distance and averaging function are varied. From the experiment results, SMM of both 3STSC and 2STSC are similar, which means 3STSC produces meaningful cluster representatives, while 3STSC can increase calculation speedup by 400 times.

6.4 Conclusion

In this chapter, Streaming Shape-based Subsequence Time Series Clustering (3STSC) is proposed to return a clustering result in real time, where the calculation complexity is constant to the number of previous subsequences. 3STSC is much faster than 2STSC in orders of magnitude, and with availability of computational power and storage, 3STSC returns comparable clustering quality to the naïve algorithm using 2STSC. In addition to 2STSC, 3STSC has the maximum allowance number of stored sequences to calculate a clustering result on this set of stored sequences, where the maximum allowance number is much smaller than the number of previous subsequences. 3STSC utilizes the updating algorithm of stored subsequences from the Incremental Shape-based Averaging, and k-hierarchical clustering with Dynamic Time Warping (DTW) distance and Shape-based Averaging as a distance measure and an averaging function,

respectively, where two inter-cluster distances, i.e., complete linkage and average linkage, and two averaging functions, i.e., CDTW and ICDTW, are utilized in 3STSC. 3STSC is considered the first streaming subsequence clustering that returns meaningful clustering results.