Algorithm 1 Decision Tree Algorithm

Input: Training Set D = $\{(x_1, y_1), (x_2, y_2), ...(x_d, y_d)\}$;
Attribute Set A = $\{a_1, a_2, ...a_d\}$;

Output: A decision tree root at node

procedure TreeGenerate(D, A)
create node;
if all the samples in D belong to the same class C **then**
label the Node as a leaf of class C; **return**
end if
if $A = \phi$ OR all the samples in D have the same value in A **then**
label the Node as a leaf of the class to which most samples belong;
end if
choose the optimal attribute a_*;
for each $a \in a_*$ **do**
create a branch for the Node;
let D_v represent the set in which all the samples have the value a;
if $D_v = \phi$ **then**
label the branch node as a leaf of the class to which most samples belong; **return**
else
take TreeGenrate(D_v, $A - a_*$) as branch node;
end if
end for

Algorithm 2 Decision Tree Algorithm

Input: Training Set D = $\{(x_1, y_1), (x_2, y_2), ...(x_d, y_d)\}$;
Attribute Set A = $\{a_1, a_2, ...a_d\}$;

Output: A decision tree root at node

procedure TreeGenerate(D, A, threshold)
create node;
if all the samples in D belong to the same class C **then**
label the Node as a leaf of class C; **return**
end if
if *depth* >= *threshold* OR all the samples in D have the same value in A **then**
label the Node as a leaf of the class to which most samples belong;
end if
choose the optimal attribute a_*;
for each $a \in a_*$ **do**
create a branch for the Node;
let D_v represent the set in which all the samples have the value a;
if $D_v = \phi$ **then**
label the branch node as a leaf of the class to which most samples belong; **return**
else
take TreeGenrate(D_v, A) as branch node;
end if
end for