

var xAxisData = [];

var data1 = [];

var data2 = [];

for (var i = 0; i < 100; i++) {

xAxisData.push('A' + i);

data1.push((Math.sin(i / 5) \* (i / 5 - 10) + i / 6) \* 5 + 60);

data2.push((Math.cos(i / 5) \* (i / 5 - 10) + i / 6) \* 5 + 60);

}

option = {

legend: {

data: ['上行', '下行']

},

toolbox: {

// y: 'bottom',

feature: {

magicType: {

type: ['stack']

},

dataView: {},

saveAsImage: {

pixelRatio: 2

}

}

},

tooltip: {},

xAxis: {

data: ['0:00','0:30','1:00','1:30','2:00','2:30','3:00','3:30','4:00','4:30','5:00','5:30','6:00','6:30','7:00','7:30','8:00','8:30','9:00','9:30','10:00','10:30','11:00','11:30','12:00','12:30','13:00','13:30','14:00','14:30','15:00','15:30','16:00','16:30','17:00','17:30','18:00','18:30','19:00','19:30','20:00','20:30','21:00','21:30','22:00','22:30','23:00','23:30'],

splitLine: {

show: false

}

},

yAxis: {

max: 120

},

series: [

{

name: '上行',

type: 'bar',

data: data1,

emphasis: {

focus: 'series'

},

animationDelay: function (idx) {

return idx \* 10;

}

},

{

name: '下行',

type: 'bar',

data: data2,

emphasis: {

focus: 'series'

},

animationDelay: function (idx) {

return idx \* 10 + 100;

}

}

],

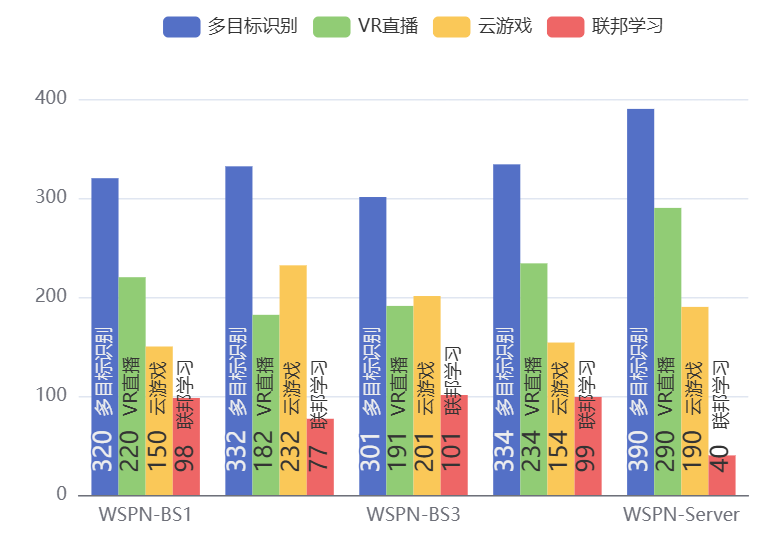
animationEasing: 'elasticOut',

animationDelayUpdate: function (idx) {

return idx \* 5;

}

};



const posList = [

'left',

'right',

'top',

'bottom',

'inside',

'insideTop',

'insideLeft',

'insideRight',

'insideBottom',

'insideTopLeft',

'insideTopRight',

'insideBottomLeft',

'insideBottomRight'

];

app.configParameters = {

rotate: {

min: -90,

max: 90

},

align: {

options: {

left: 'left',

center: 'center',

right: 'right'

}

},

verticalAlign: {

options: {

top: 'top',

middle: 'middle',

bottom: 'bottom'

}

},

position: {

options: posList.reduce(function (map, pos) {

map[pos] = pos;

return map;

}, {})

},

distance: {

min: 0,

max: 100

}

};

const labelOption = {

show: true,

position: app.config.position,

distance: app.config.distance,

align: app.config.align,

verticalAlign: app.config.verticalAlign,

rotate: app.config.rotate,

formatter: '{c} {name|{a}}',

fontSize: 16,

rich: {

name: {}

}

};

option = {

tooltip: {

trigger: 'axis',

axisPointer: {

type: 'shadow'

}

},

legend: {

data: ['多目标识别', 'VR直播', '云游戏', '联邦学习']

},

toolbox: {

show: true,

orient: 'vertical',

left: 'right',

top: 'center',

feature: {

mark: { show: true },

dataView: { show: true, readOnly: false },

magicType: { show: true, type: ['line', 'bar', 'stack'] },

restore: { show: true },

saveAsImage: { show: true }

}

},

xAxis: [

{

type: 'category',

axisTick: { show: false },

data: ['WSPN-BS1', 'WSPN-BS2', 'WSPN-BS3', 'WSPN-BS4', 'WSPN-Server']

}

],

yAxis: [

{

type: 'value'

}

],

series: [

{

name: '多目标识别',

type: 'bar',

barGap: 0,

label: labelOption,

emphasis: {

focus: 'series'

},

data: [320, 332, 301, 334, 390]

},

{

name: 'VR直播',

type: 'bar',

label: labelOption,

emphasis: {

focus: 'series'

},

data: [220, 182, 191, 234, 290]

},

{

name: '云游戏',

type: 'bar',

label: labelOption,

emphasis: {

focus: 'series'

},

data: [150, 232, 201, 154, 190]

},

{

name: '联邦学习',

type: 'bar',

label: labelOption,

emphasis: {

focus: 'series'

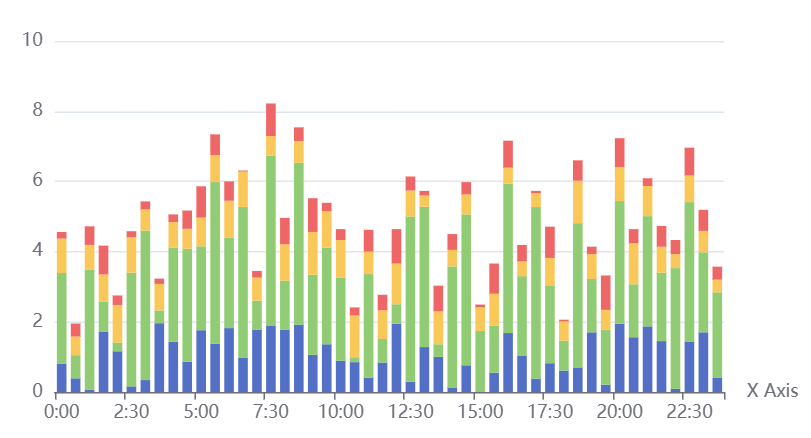
},

data: [98, 77, 101, 99, 40]

}

]

};



let xAxisData = [];

let data1 = [];

let data2 = [];

let data3 = [];

let data4 = [];

for (let i = 0; i < 48; i++) {

xAxisData.push('Class' + i);

data1.push(+(Math.random() \* 2).toFixed(2));

data2.push(+(Math.random() \* 5).toFixed(2));

data3.push(+(Math.random() + 0.3).toFixed(2));

data4.push(+Math.random().toFixed(2));

}

var emphasisStyle = {

itemStyle: {

shadowBlur: 10,

shadowColor: 'rgba(0,0,0,0.3)'

}

};

option = {

legend: {

data: ['0:00','0:30','1:00','1:30','2:00','2:30','3:00','3:30','4:00','4:30','5:00','5:30','6:00','6:30','7:00','7:30','8:00','8:30','9:00','9:30','10:00','10:30','11:00','11:30','12:00','12:30','13:00','13:30','14:00','14:30','15:00','15:30','16:00','16:30','17:00','17:30','18:00','18:30','19:00','19:30','20:00','20:30','21:00','21:30','22:00','22:30','23:00','23:30'],

left: '10%'

},

tooltip: {},

xAxis: {

data: ['0:00','0:30','1:00','1:30','2:00','2:30','3:00','3:30','4:00','4:30','5:00','5:30','6:00','6:30','7:00','7:30','8:00','8:30','9:00','9:30','10:00','10:30','11:00','11:30','12:00','12:30','13:00','13:30','14:00','14:30','15:00','15:30','16:00','16:30','17:00','17:30','18:00','18:30','19:00','19:30','20:00','20:30','21:00','21:30','22:00','22:30','23:00','23:30'],

name: 'X Axis',

axisLine: { onZero: true },

splitLine: { show: false },

splitArea: { show: false }

},

yAxis: {},

grid: {

bottom: 100

},

series: [

{

name: 'bar',

type: 'bar',

stack: 'one',

emphasis: emphasisStyle,

data: data1

},

{

name: 'bar2',

type: 'bar',

stack: 'one',

emphasis: emphasisStyle,

data: data2

},

{

name: 'bar3',

type: 'bar',

stack: 'one',

emphasis: emphasisStyle,

data: data3

},

{

name: 'bar4',

type: 'bar',

stack: 'one',

emphasis: emphasisStyle,

data: data4

}

]

};

myChart.on('brushSelected', function (params) {

var brushed = [];

var brushComponent = params.batch[0];

for (var sIdx = 0; sIdx < brushComponent.selected.length; sIdx++) {

var rawIndices = brushComponent.selected[sIdx].dataIndex;

brushed.push('[Series ' + sIdx + '] ' + rawIndices.join(', '));

}

myChart.setOption({

title: {

backgroundColor: '#333',

text: 'SELECTED DATA INDICES: \n' + brushed.join('\n'),

bottom: 0,

right: '10%',

width: 100,

textStyle: {

fontSize: 12,

color: '#fff'

}

}

});

});