Dijkstra算法和Kruskal算法之间有共同点，最大的区别在于解决的问题不同，因而有不同的用途。

Dijkstra是解决「单源最短路径问题」的算法。这个问题是说，如何找到从某个特定的节点出发，通向其他节点的最短路径。它只着眼于点与点之间的路径问题，并不关注整个图，也就意味着对一个节点运行Dijkstra算法的结果与另一个节点的结果之间没有多少关系。

比如说，可以把城市的路口看作图的节点，把公路看做边，综合长度、拥堵程度等指标作为边的权重，就可以通过Dijkstra算法计算出从城市一点到另一点的最短路线。

Kruskal则是解决「最小生成树问题」的算法，即在一个连通的图里，如何去除图里的边，使得剩余的边仍能连接所有的节点，且这些边的权重之和最小。显然，满足这个要求的图不可能存在环，也就是一棵树，因此叫做生成树。这种算法与上面的相反，着眼于整个图的结构，并不关心某两个节点之间的路径是不是最短的。

这种算法的应用也很广泛，比如说有一个快递公司，每天都要开车把快递送到城市里的不同地点，怎样走才能不重复地经过每个节点，还能让总时间最短呢？我们就可以用Kruskal这样的最小生成树算法，找到一个最小生成树，只需要按这个路线走就行了。