pseudo code of my algorithm

1. Train the Bayesian Machine (Transform function)

```
Let set:
$I = \{1,2,3 \ldots, i \ldots.\}$  the set of intersections
$S = \{1,2,3, \ldots, s, \ldots\}$ the set of streets
$V = \{1,2,3, \ldots, v, \ldots\}$ the set of vehicles
$T = \{1,2,3, \ldots t, \ldots\}$ the set of time slot
$Roads$ as the set of streets that vehicles pass
$LocaToStr$ as the set of streets that vehicles pass in each time slot
        $LocaToStr(v,t) = 0$ means in time slot t, vehicle v is on one intersection
$LenR$ as the set of length of each vehicle's routine

For each $v$ in $V$
    $Start\_time$ ← arrival time slot of v
    $End\_time$ ← departure time slot of v
    $t$: From $Start\_time$ to $End\_time$
        $LocaToStr(v,t)$ ← the street that $v$ passes in current slot $t$(Function Get_street)
        $if\ LenR(v) = 0\ or\ LocaToStr(v,t) \neq Roads(v, LenR(v))$
          $LenR(v)$ ← $LenR(v) + 1$
          $Roads(v, LenR(v))$ ← $LocaToStr(v,t)$
        $end$
    End
End

$maxl$ the maximum prediction length
$PD\_likehood(s, sta)$ the set of the probability that vehicle go to street $s$ passing streets $sta$
    Ps: the definition of $sta$ is in the middle code $Transform.m$

For each $v$ in $V$
    $sl$: From 1 to $LenR(v)$
        $pl$: From 1 to $maxl$
            Calculate $sta$ of $sl, pl$
            $PD\_prior(sl, sta)$ ← $PD\_prior(sl, sta) + 1$
        End
    End
End
For each $v$ in $V$
    $sl$: From 1 to $LenR(v)$
        $pl$: From 1 to $maxl$
            Calculate $sta\_now$ of $sl, pl$
            Calculate $sta\_pre$ of $sl - 1, pl - 1$
          $PD\_likehood(sl, sta\_now)$ ← $PD\_prior(sl - 1, sta_{pre})/PD\_prior(sl, sta\_now)$
        End
    End
End
```

$Pararoad$ the set of properties of roads

Ps: The definition of $Pararoad$ is in the middle code $Transform.m$

$Crossroad$ the set of properties of intersections

Ps: the Definition $Crossroad$is in the meddle code $Transform.m$

```
For each $v$ in $V$
    $Start\_time$← arrival time slot of v
    $End\_time$← departure time slot of v
    $cs$ ← 1
    $t$: From $Start\_time$ to $End\_time$
        If $LocaToStr(v,t) \neq Road(v,cs)\ and\ ocaToStr(v,t) \neq 0$
            $ParaRoad$ ← $ParaRoad + Calc\_property(Velocity, Position)$
            $CurrentIns$ ← City.Street(Road(v, cs), 2)
            $Crossroad(CurrenstIns)$ ← $Crossroad(CurrenstIns) +$ Time Cost of $CurrentIns$
            $cs$ ← $cs + 1$
        Else
            $Velocity$ ← $[Velocity, speed(v,t)]$
            $Position$ ← $[Position, postion(v,t)]$
        End
    End
    $ParaRoad$ ← $ParaRoad + Calc\_property(Velocity, Position)$
    $CurrentIns$ ← City.Street(Road(v, cs), 2
    $Crossroad(CurrenstIns)$ ← $Crossroad(CurrenstIns)$ +Time Cost of $CurrentIns$
End

For each $s$ in $S$
    Calculate the average $ParaRoad$
End

For each $i$ in $I$
    Calculate the average $Crossroad$
End
```

2. Sumo Prediction

Input:

$Pastroad$ the set of intersections that vehicle has passed

$CurrentLocation$ the current location of vehicle

$e.g$: based on the City you just sent me

$Pastroad = [14, 15,16,17]$ $CurrenLoation = [2300, 1244]$

$Suftime$: the remaining time of vehicle
Other necessary input are showed in code

$nowRoad$: the set of vehicle has passed
$Trace$: the set of vehicle will pass predicted by Bayesian Machine
$Prob$: the probability of vehicle' $Trace$
$FinaLocation$: the final location of vehicle
$Remaintime$: the remaining time of vehicle to pass current street
$prdl$： the most proper prediction length, current is two

```
$Len$ ← length of $Pastroad$
$lenow$ ←$Lenow$
Use $Pastroad$ and $CurrentLocation$ find $nowRoad$
$while(1)$
    $j$: From $max(1, lenow - prdl)$ to $lenow - 1$
        Calculate the $sta$ of $nowRoad(lenow)$
    End
    $l$: For 1 to 3
        $suf$←the next street that go from street $nowRoad(lenow)$ in direction $l$
        $nextStreet$←the $suf$ that has $max\{PD\_likehood(suf, sta * 4 + l)\}$
    End
    $Remaintime$← the remaining time of vehicle to current street $nowRoad(lenow)$
    $if\ Remaintime \leq Suftime$
        Calculate $FinaLocation$
        Break
    Else
        $if\ nextStreet = 0$
            $FinaLocation$←$[-1, -1]$ //The vehicle will go out of the city
            Break
        Else
            $Suftime = Suftime - Remaintime$
            $if\ Suftime \leq Crossroad(currentIns)$ //the vehicle can't pass the intersection
                $FinaLocation$←Location of $currentIns$
                Break
            Else
                $Suftime$← $Suftime - Crossroad(currentIns)$
                $lenow$← $lenow + 1$
                $nowRoad$← $[nowRoad, nextStreet]$
                $Trace$← $[Trace, nextStreet]$
                $Prob$← $[Prob, the\ probability\ of\ nextStreet\ based\ on\ past\ roads]$
            End
        End
    End
$end$
```