SmokeSNET Model

Matt Smith

February 1, 2013

1 Current Attributes

As it stands, nodes have the attributes seen in figure 1. Edges are directed and their weight is a probability, representing their influence. Currently, all attributes are normally distributed and can be adjusted to more realistic values later in development. Nodes can also be constrained to a maximum number of in-edges, i.e. nodes which influence them. Within each simulation step, the following happens:

1. The local neighbourhood within three incoming hops is acquired for the given node. Influence between the nodes is calculated as the maximum influence across all possible connections of the two nodes, where influence over multiple hops is the product of the influence of each hop. For example, in Fig. 1, the influence of *Node C* on the *Current Node* is the maxiumum of $0.8 * 0.8$ and 0.1, where the best value here is 0.64 for *Node C* to *D* to *Current*. Note that even if a one-hop route exists, the maximum influence will be chosen for consistency across the neighbourhood. All nodes within the neighbourhood set are unique.

2. Metrics for this set are calculated for the current node relative to the neighborhood. Some general ones, such as the percentage of the neighbourhood which is giving up smoking, along with the percentage that currently smoke is calculated. For each node attribute, the influence weighted average seen in figure 2 is calculated.

3. The decision tree is run on the nodes. At present, this tree can be drawn as seen in figure 3. The relevant probability calculations are in figure 3. For decisions such as 'Is a Smoker?', the decision is based on the attribute value so no probability needs to be calculated. For each, a random number between 0 and 1 is generated - if it is lower than the calculated probability, then the decision is considered to have happened. The aim of the decision tree is to adjust the attribute values of the individual node.

4. Finally, a nodes connections are adjusted based on its attributes. Currently, the nodes have an upper limit of in-edges, representing influence on the node - (simulations run with 10 at the moment) to avoid

Name	Type	Represents
isSmoker	Boolean	True if they're a smoker, false otherwise
willpower	Double	A probability representing willpower, 0 being of strong willpower.
health	Double	A value between 0 and 1 for health, where 1 is perfect health and 0 is a smoking-related disease.
smokedPerDay	Integer	The number of cigarettes smoked per day
givingUp	Boolean	A side-status for non-smokers, where true means they are giving up. Every turn this can influence whether a person starts again, whether they become a ńormalńon-smoker and can affect others who are also giving up.
giveUpAttempts	Integer	The number of attempts at giving up.
stepsSinceGiveUp	Integer	The number of simulation steps since the last decision to give up.
sociable	Double	A probability representing how sociable someone is, with 1 being very sociable

Table 1: Model Attributes

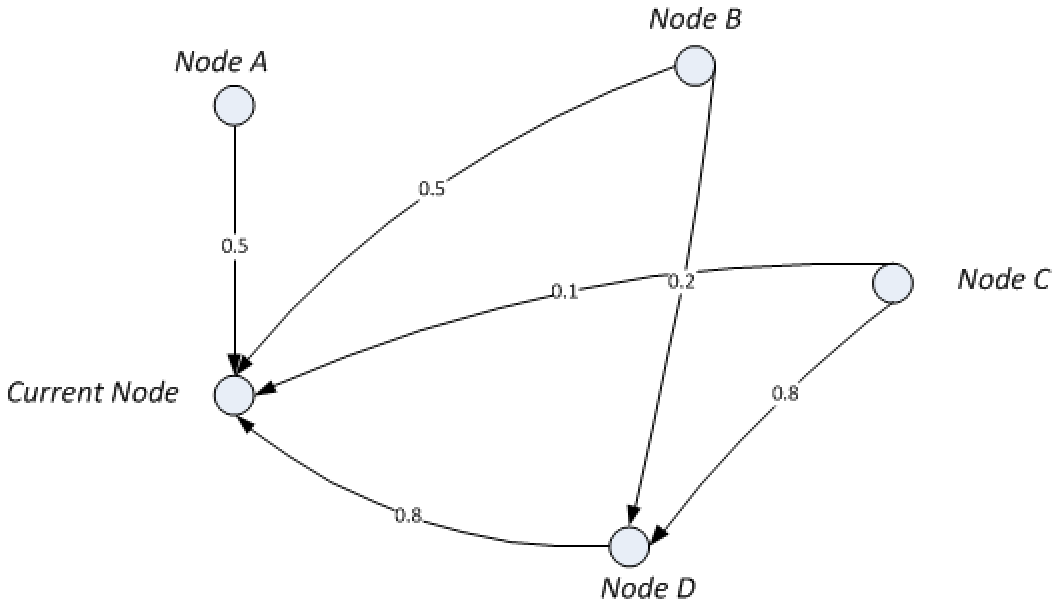

Figure 1: Network Diagram

$$\frac{\sum_{\forall n \in N} attribute_n \times influence_n}{\sum_{\forall n \in N} influence_n} \text{ where } N \text{ is the set of nodes}$$

Figure 2: Influence Calculation

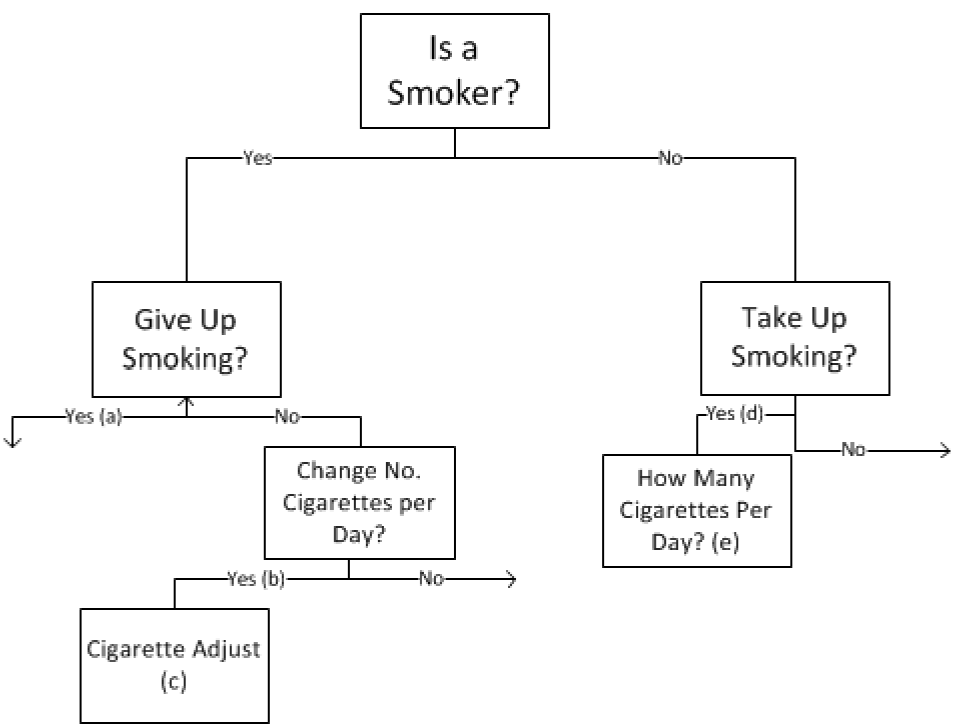

Figure 3: Decision Tree

$\forall n \in Nodes$

(a) p(give up smoking)=$(1 - health_n) \times \frac{|\text{Nodes giving up smoking}|}{|\text{Nodes}|}$

(b) If n smokes $\pm 10\%$ of the influence sum for cigarettes smoked, and nś health is within $\pm 10\%$ of the influence sum of health.

(c) $change$ = (influence sum of smoked per day $- smokedPerDay_n) \times influenceability_n$, else there is no change.

(d) p(take up smoking) $= health_n \times \frac{|\text{Nodes giving up smoking}|}{|\text{Nodes}|} \times \frac{|\text{Nodes who smoke}|}{|\text{Nodes}|}$

(e) If the influenced sum of the number of cigarettes per day is < 0, then $smokedPerDay = 5$, otherwise $smokedPerDay = \text{round}(\frac{\sum_{\forall n \in Neighbours} smokedPerDay_n \times influence_n}{\sum_{\forall n \in N} influence_n})$

Figure 4: Decision Tree Calculations

hugely saturating the graph. If a node reaches this maximum but still wishes to add an edge, it can remove a low influence edge e with a probability $p(removal) = (1 - \min_{\forall e \in \text{In Edges}}(influence_e))$.
Edges are selected for addition based on a scoring system of each node in the n-hop neighbourhood. The system uses multiple approaches to determine a score (i.e. a 'worthiness') for each of these nodes:

(a) Boolean scoring, where the node either gets all the points, or none of the points

(b) Linear scoring, in which the percentage difference of the node being considered from the source node is deducted from a points total, i.e. $score = maxPoints - \%\ \text{diff} \times maxPoints$ - if the target node is 100% or over different, then they will score zero on that comparison.

(c) Range scoring, where if one node is within a given range of another, a set score is given.

(d) Percent scoring, where the attribute is multiplied against a maximum score to get the net score.

The current scoring algorithm can be seen in figure 4. It is very basic and as such, gives little spread in the scores of nodes. The effect of this is that it becomes difficult to differentiate good connection candidates from average or poor ones - a way to circumvent this is to add in extra attributes and tune the scores per attribute until a reasonable spread is given.

The total score is divided by the maximum available score to give a value between 0 and 1. A threshold is then set to determine whether to add or remove the edge (i.e. if the score is less than 0.1, remove the edge, if it is between 0.1 and 0.2 do nothing, otherwise add it.

2 Upcoming Features

- A more complex decision tree, incorporating all attributes and causing gradual change over time.
- Network metric analysis during the simulation to check if the graph is converging to a 'small world' style graph. This is through looking at the average clustering coefficient of the graph, and depending on the results of this, further graph statistics.
- More attributes that affect smoking cessation being added.

```
for(Node n : Neighbours)
{
        if(current.isSmoker == n.isSmoker)
        {
                score + 1
                if(current.isGivingUp && n.isGivingUp)
                {
                        score + 2
                        if(n.stepsSinceGiveUp is within 50% of current.stepsSinceGiveUp)
                                score + 5
                        else
                                score + 2
                }
                if(current.isSmoker)
                {
                        score + linearScore(smokedPerDay)       //Max 5 pts
                }
        }
        score + linearScore(health)             //Max 5 pts
        score + linearScore(willpower)          //Max 5 pts
        score + linearScore(sociable)           //Max 5 pts

        score + percentScore(influenceability)  //Max 5 pts
        score + percentScore(persuasiveness)    //Max 5 pts

        return score/33
}
```

Figure 5: Scoring Algorithm