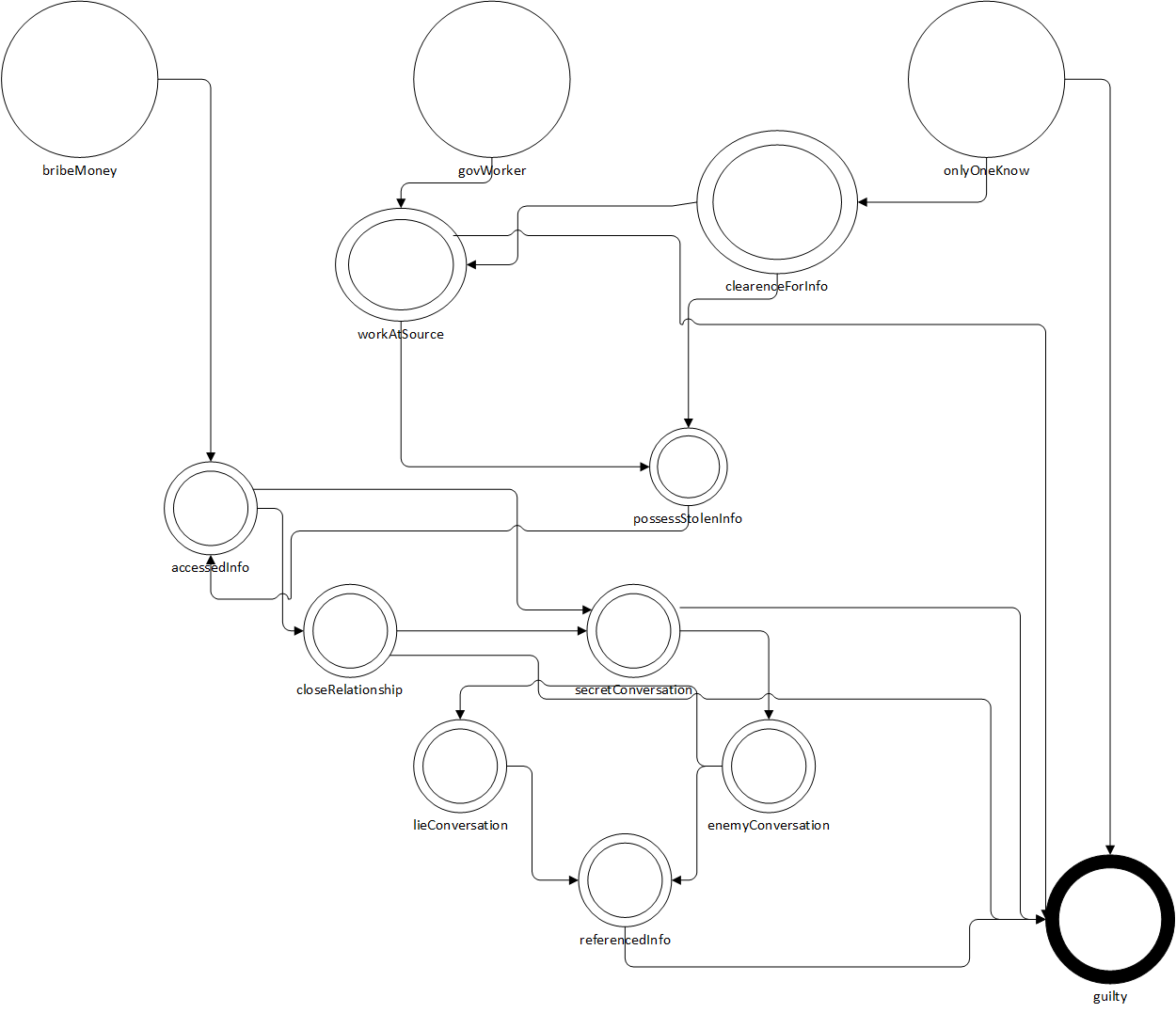
*Joel Wilhelm and Dallas Barnett*

*CSC 529- Spring ‘17*

Implementation of a Bayseian Network and Expert System

**Digital Drawing of Bayseian Network**



**Variable Names**

|  |  |
| --- | --- |
| Variable Computer Name | Variable English Name |
| accessedInfo | true if suspect was found to have accessed the information |
| bribeMoney | true if the suspect was found to receive bribes |
| clearenceForInfo | True suspect had the proper security clearance for the information |
| closeRelationship | true if the suspect had close ties to someone that did access the information |
| enemyConversation | true if the suspect had talked with an enemy |
| govWorker | True if the suspect was a government worker/contractor |
| guilty | Variable that holds the probability of the suspect being guilty |
| lieConversation | true if the suspect lied about the conversations in question above |
| onlyOneKnow | true if suspect was the only person that knew the information |
| possessStolenInfo | true if suspect was in the possession of the stolen information |
| referencedInfo | true if the suspect seemed to have referenced the information in the conversations above |
| secretConversation | true if the suspect had been in secret talks with someone |
| workAtSource | true if the suspect worked at/for the source of the information |

**Probability Table**

|  |  |
| --- | --- |
| Variable Name | Probability Value |
| accessedInfo | p(accessedInfo, [bribeMoney], 0.4).  p(accessedInfo, [not(bribeMoney)], 0.01). |
| bribeMoney | p(bribeMoney, ASK). (initial probability) |
| clearenceForInfo | p(clearenceForInfo, [onlyOneKnow], 0.999).  p(clearenceForInfo, [not(onlyOneKnow], 0.85). |
| closeRelationship | p(closeRelationship, [accessedInfo], 0.1).  p(closeRelationship, [not(accessedInfo], 0.74). |
| enemyConversation | p(enemyConversation, [secretConversation], 0.9).  p(enemyConversation, [not(secretConversation)], 0.05). |
| govWorker | p(govWorker, ASK). (initial probability) |
| guilty | prob(guilty, [onlyOneKnow], N).  prob(guilty, [not(onlyOneKnow), workAtSource, secretConversation, closeRelationship, referencedInfo], N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), secretConversation, closeRelationship, referencedInfo], N).  prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), closeRelationship, referencedInfo], N).  prob(guilty, [not(onlyOneKnow), workAtSource, secretConversation, not(closeRelationship), referencedInfo], N).  prob(guilty, [not(onlyOneKnow), workAtSource, secretConversation, closeRelationship, not(referencedInfo), N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), closeRelationship, referencedInfo, N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), secretConversation, not(closeRelationship), referencedInfo, N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), secretConversation, closeRelationship, not(referencedInfo), N).  prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), not(closeRelationship), referencedInfo, N).  prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), closeRelationship, not(referencedInfo), N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), not(closeRelationship), referencedInfo, N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), closeRelationship, not(referencedInfo), N).  prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), not(closeRelationship), not(referencedInfo), N).  prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), not(closeRelationship), not(referencedInfo), N). |
| lieConversation | p(lieConversation, [enemyConversation], 0.9).  p(lieConversation, [not(enemyConversation)], 0.001). |
| onlyOneKnow | p(onlyOneKnow, ASK). (initial probability). |
| possessStolenInfo | p(possessStolenInfo, [workAtSource, clearenceForInfo], 0.7).  p(possessStolenInfo, [not(workAtSource), clearenceForInfo], 0.2).  p(possessStolenInfo, [workAtSource, not(clearenceForInfo)], 0.01).  p(possessStolenInfo, [not(workAtSource), not(clearenceForInfo)], 0.001). |
| referencedInfo | p(referencedInfo, [lieConversation, enemyConversation], 0.9).  p(referencedInfo, [not(lieConversation), enemyConversation], 0.82).  p(referencedInfo, [lieConversation, not(enemyConversation)], 0.85).  p(referencedInfo, [not(lieConversation), not(enemyConversation)], 0.4). |
| secretConversation | p(secretConversation, [closeRelationship, accessedInfo], 0.4).  p(secretConversation, [not(closeRelationship), accessedInfo], 0.65).  p(secretConversation, [closeRelationship, not(accessedInfo)], 0.35).  p(secretConversation, [not(closeRelationship), not(accessedInfo)], 0.25). |
| workAtSource | p(workAtSource, [govWorker, clearenceForInfo], 0.8).  p(workAtSource, [not(govWorker), clearenceForInfo], 0.32).  p(workAtSource, [govWorker, not(clearenceForInfo)], 0.2).  p(workAtSource, [not(govWorker), not(clearenceForInfo)], 0.01). |

**Prolog Definition of BN**

parent(bribeMoney, accessedInfo).

parent(govWorker, workAtSource).

parent(onlyOneKnow, guilty).

parent(onlyOneKnow, clearenceForInfo).

parent(clearenceForInfo, workAtSource).

parent(clearenceForInfo, possessStolenInfo).

parent(workAtSource, possessStolenInfo).

parent(workAtSource, guilty).

parent(possessStolenInfo, accessedInfo).

parent(accessedInfo, closeRelationship).

parent(accessedInfo, secretConversation).

parent(closeRelationship, secretConversation).

parent(closeRelationship, guilty).

parent(secretConversation, enemyConversation).

parent(secretConversation, guilty).

parent(enemyConversation, referencedInfo).

parent(enemyConversation, lieConversation).

parent(lieConversation, referencedInfo).

parent(referencedInfo, guilty).

**Sample Queries**

prob(guilty, [onlyOneKnow], N).

prob(guilty, [not(onlyOneKnow), workAtSource, secretConversation, closeRelationship, referencedInfo], N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), secretConversation, closeRelationship, referencedInfo], N).

prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), closeRelationship, referencedInfo], N).

prob(guilty, [not(onlyOneKnow), workAtSource, secretConversation, not(closeRelationship), referencedInfo], N).

prob(guilty, [not(onlyOneKnow), workAtSource, secretConversation, closeRelationship, not(referencedInfo), N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), closeRelationship, referencedInfo, N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), secretConversation, not(closeRelationship), referencedInfo, N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), secretConversation, closeRelationship, not(referencedInfo), N).

prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), not(closeRelationship), referencedInfo, N).

prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), closeRelationship, not(referencedInfo), N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), not(closeRelationship), referencedInfo, N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), closeRelationship, not(referencedInfo), N).

prob(guilty, [not(onlyOneKnow), workAtSource, not(secretConversation), not(closeRelationship), not(referencedInfo), N).

prob(guilty, [not(onlyOneKnow), not(workAtSource), not(secretConversation), not(closeRelationship), not(referencedInfo), N).

**Forward Chaining Rules**

if military then the\_punishment\_for\_the\_suspect\_is\_death.

if not(military) and criminal and willing and serious then the\_punishment\_for\_the\_suspect\_is\_death.

if not(military) and criminal and willing and petty then the\_punishment\_for\_the\_suspect\_is\_life\_with\_possible\_parole.

if not(military) and criminal and not(willing) and not(ill) then the\_punishment\_for\_the\_suspect\_is\_10000\_dollar\_fine\_and\_20\_years.

if not(military) and criminal and not(willing) and ill then the\_punishment\_for\_the\_suspect\_is\_20\_years.

if not(military) and not(criminal) and willing and serious then the\_punishment\_for\_the\_suspect\_is\_death.

if not(military) and not(criminal) and willing and petty then the\_punishment\_for\_the\_suspect\_is\_10000\_dollar\_fine\_and\_30\_years.

if not(military) and not(criminal) and not(willing) and remorse and ill then the\_punishment\_for\_the\_suspect\_is\_5000\_dollar\_fine\_and\_mental\_health\_help.

if not(military) and not(criminal) and not(willing) and remorse and not(ill) then the\_punishment\_for\_the\_suspect\_is\_5000\_dollar\_fine\_and\_1\_year.

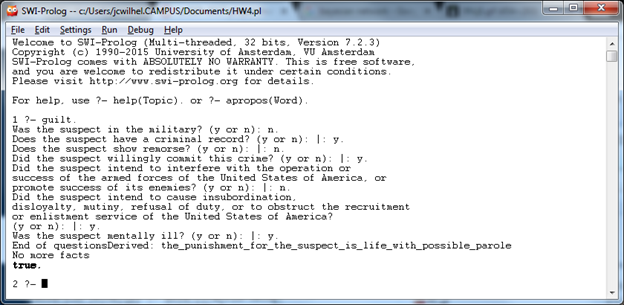
if not(military) and not(criminal) and not(willing) and not(remorse) and ill then the\_punishment\_for\_the\_suspect\_is\_7500\_dollar\_fine\_and\_5\_years.

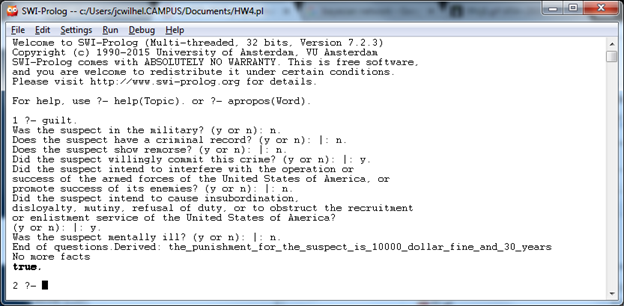
if not(military) and not(criminal) and not(willing) and not(remorse) and not(ill) then the\_punishment\_for\_the\_suspect\_is\_7500\_dollar\_fine\_and\_10\_years.

**FC Variable Definition Table**

|  |  |
| --- | --- |
| **TERM** | **EXPLANATION** |
| **ask\_user** | Actually starts calling the program to run. |
| **criminal** | Fact asserted if the suspect had a prior criminal activity. |
| **forward** | Utilizes the Bratko Forward Chaining interpreter. |
| **guilt** | Starts the Forward Chaining Rules by calling ask\_user and forward. |
| **ill** | Fact asserted if the suspect was mentally ill. |
| **military** | Fact asserted if the suspect was active in military. |
| **petty** | Fact asserted if the suspect intended to cause minor problems within the United States Armed Forces, such as disloyalty, or insubordination. |
| **remorse** | Fact asserted if the suspect showed remorse. |
| **serious** | Fact asserted if the suspect intended to cause severe harm towards the United States Armed Forces, and aid its enemies. |
| **willing** | Fact asserted if the suspect willingly committed the crime. |

**Sample Queries on System**





**Forward Chaining Question Possibly Asked**

Was the suspect in the military? (y or n):

Does the suspect have a criminal record? (y or n):

Does the suspect show remorse? (y or n):

Did the suspect willingly commit this crime? (y or n):

Did the suspect intend to interfere with the operation or success of the armed forces of the United States of America, or promote success of its enemies? (y or n):

Did the suspect intend to cause insubordination, disloyalty, mutiny, refusal of duty, or to obstruct the recruitment or enlistment service of the United States of America? (y or n):

Was the suspect mentally ill? (y or n):