Top-down approach to compare the moral theories of Deontology and Utilitarianism in Pac-Man game setting

Presented by: Niyati Rawal

Supervisor: Dr. Joan Casas-Roma
More and more tasks are getting increasingly automatized as decisions are being made by autonomous agents.

These decisions will, and already have, consequences that can cause great good or harm to individuals and society.

There is a need to ensure that these decisions are in line with moral values.
Moral Theories: Deontology and Utilitarianism

• Deontology focuses on the intention behind an action and/or the nature of an act. Acts may be required, forbidden or permissible. Eg.: acts like killing, stealing etc. are forbidden.

Alexander and Moore (2016)
Moral Theories: Deontology and Utilitarianism

- Deontology focuses on the intention behind an action and/or the nature of an act. Acts may be required, forbidden or permissible. Eg.: acts like killing, stealing etc. are forbidden. Alexander and Moore (2016)

- Utilitarianism focuses on the consequences of an action alone. The goal is to maximize happiness for the greatest number of moral beings. Sinnott-Armstrong (2015)
Theory of Dyadic Morality

Every moral or immoral action involves two entities:

1. An agent (source of action)
2. A patient (receiver of action)

Gray and Schein (2012)
Research Goal and Approach

To computationally model and compare the moral theories of Deontology and Utilitarianism in a common game setting using the top-down approach.
Research Goal and Approach

To computationally model and compare the moral theories of Deontology and Utilitarianism in a common game setting using the top-down approach.

- Inspired by the famous Pac-Man game we model: a Deontological Pac-Man agent and a Utilitarian Pac-Man agent.
Research Goal and Approach

To computationally model and compare the moral theories of Deontology and Utilitarianism in a common game setting using the top-down approach

- Inspired by the famous Pac-Man game we model: a Deontological Pac-Man agent and a Utilitarian Pac-Man agent

- Top-down Approach: This approach explicitly captures the theories it aims to represent into a set of rules
Pac-Man World Settings

- Elements: 16 pac-dots, 1 big pac-dot, 1 fruit
- Points scheme:

<table>
<thead>
<tr>
<th>Action</th>
<th>Points awarded</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pac-Man eats pac-dots</td>
<td>$10 \times 17 = 170$ points</td>
</tr>
<tr>
<td>Pac-Man rescues the fruit</td>
<td>200 points</td>
</tr>
<tr>
<td>Pac-Man kills the ghost</td>
<td>200 points</td>
</tr>
<tr>
<td>Pac-Man clears the level</td>
<td>500 points</td>
</tr>
</tbody>
</table>

Points required to clear the level = 370
Agency and Patiency for actions

- When Pac-Man has not eaten a big pac-dot, the ghost can kill Pac-Man

Agent: Ghost  Patient: Pac-Man
Agency and Patiency for actions

• When Pac-Man has not eaten a big pac-dot, the ghost can kill Pac-Man
  Agent: Ghost  Patient: Pac-Man

• When Pac-Man has eaten a big pac-dot, it can kill the ghost
  Agent: Pac-Man  Patient: Ghost
Agency and Patiency for actions

- When Pac-Man has not eaten a big pac-dot, the ghost can kill Pac-Man
  
  Agent: Ghost
  Patient: Pac-Man

- When Pac-Man has eaten a big pac-dot, it can kill the ghost
  
  Agent: Pac-Man
  Patient: Ghost

- If ghost eats (“traps”) the fruit first
  
  Agent: Ghost
  Patient: fruit
Agency and Patiency for actions

- When Pac-Man has not eaten a big pac-dot, the ghost can kill Pac-Man
  Agent: Ghost  Patient: Pac-Man

- When Pac-Man has eaten a big pac-dot, it can kill the ghost
  Agent: Pac-Man  Patient: Ghost

- If ghost eats ("traps") the fruit first
  Agent: Ghost  Patient: fruit

- If Pac-Man eats ("rescues") the fruit first
  Agent: Pac-Man  Patient: fruit
Standard Game Rules

Rules:

1. If there are pac-dots available, then Pac-Man should eat the pac-dots
exist (PacDot) ⇒ eat (PacDot)

2. If there exists ghost and it is not in scared state, then Pac-Man should escape
exist (Ghost) ∧ ¬eat (BigPacDot) ⇒ escape(Ghost)

Using Breadth First Search Algorithm (BFS)
Standard Game Rules

Rules:
1. exist (PacDot) ⇒ eat (PacDot)
2. exist (Ghost) ∧ ¬eat (BigPacDot) ⇒ escape (Ghost)
Deontological Pac-Man Agent

Rules:
3a. If the fruit is still available, Pac-Man has the duty of rescuing the fruit
   \[ \text{exist (Fruit)} \Rightarrow \text{rescue (Fruit)} \]

4a. If Pac-Man eats the big pac-dot, it should escape ghost to not kill it
   \[ \text{eat (BigPacDot)} \Rightarrow \text{escape (Ghost)} \land \neg \text{kill (Ghost)} \]
Deontological Pac-Man Agent

Rules:

3a. exist (Fruit) ⇒ rescue (Fruit)

4a. eat (BigPacDot) ⇒ escape (Ghost) ∧ ¬kill (Ghost)
# Utilitarian Pac-Man Agent

As Utilitarianism emphasizes on “the greatest good for the greatest number”, we set happiness and pain scores for every action.

<table>
<thead>
<tr>
<th>Action</th>
<th>Happiness score</th>
<th>Pain score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pac-Man clearing the level</td>
<td>+870</td>
<td></td>
</tr>
<tr>
<td>Pac-Man unable to clear the level</td>
<td></td>
<td>-870</td>
</tr>
<tr>
<td>Pac-Man rescuing the fruit</td>
<td>+200</td>
<td></td>
</tr>
<tr>
<td>Pac-Man failed to rescue the fruit</td>
<td></td>
<td>-200</td>
</tr>
<tr>
<td>Pac-Man killing the ghost</td>
<td></td>
<td>-200</td>
</tr>
<tr>
<td>Ghost trapping the fruit</td>
<td></td>
<td>-200</td>
</tr>
<tr>
<td>Ghost failed to trap the fruit</td>
<td>+200</td>
<td></td>
</tr>
</tbody>
</table>
Utilitarian Pac-Man Agent

Rules:

3b. If Pac-Man rescues fruit and eats the big pac-dot, it should escape ghost to not kill it

\[\text{rescue(Fruit)} \land \text{eat(BigPacDot)} \Rightarrow \text{escape(Ghost)} \land \neg\text{kill(Ghost)}\]

4b. If Pac-Man fails to rescue fruit, it should eat the big pac-dot and kill ghost

\[\neg\text{rescue(Fruit)} \Rightarrow \text{eat(BigPacDot)} \land \text{kill(Ghost)}\]
Utilitarian Pac-Man Agent

Rules:
3b. rescue(Fruit) ∧ eat(BigPacDot) ⇒ escape(Ghost) ∧ ¬kill(Ghost)
4b. ¬rescue(Fruit) ⇒ eat(BigPacDot) ∧ kill(Ghost)
Utilitarian Pac-Man Agent

Rules:
3b. rescue(Fruit) ∧ eat(BigPacDot) ⇒ escape(Ghost) ∧ ¬kill(Ghost)
4b. ¬rescue(Fruit) ⇒ eat(BigPacDot) ∧ kill(Ghost)
Scenario 1:
If the ghost traps the fruit first, what would the Pac-Man agent do? Would it kill the ghost?
Scenario 1:
If the ghost traps the fruit first, what would the Pac-Man agent do? Would it kill the ghost?

- The deontological Pac-Man agent will not kill the ghost as it is wrong to kill. Thus, it would not be able to clear the level.
- The utilitarian Pac-Man agent realizes that the only way to clear the level is to kill the ghost. As the amount of happiness in clearing the level outweighs the pain of killing the ghost, the utilitarian Pac-Man agent would kill the ghost.
Scenario 2:
If ghost blocks Pac-Man and the fruit, would the Pac-Man agent kill the ghost to rescue the fruit?
Scenario 2:
If ghost blocks Pac-Man and the fruit, would the Pac-Man agent kill the ghost to rescue the fruit?

- The deontological Pac-Man agent would not kill the ghost even to protect the fruit as it is wrong to kill.
- The utilitarian Pac-Man agent should kill the ghost as there is greater happiness in rescuing the fruit and preventing it from getting trapped by the ghost than the pain in killing the ghost.
Conclusions

● While the deontological agent is sometimes unable to clear the level, the utilitarian agent always manages to clear the level.

● The deontological agent may sometimes have to face conflict between succeeding and sticking to it’s value of always doing the right thing.

● Taking the consequences into account can help the deontological agent realize that sometimes it maybe necessary to kill the ghost.
Future Work

- **Integrate long-term planning** into the BFS and moral algorithms to prevent the deontological agent from getting stuck, or help the utilitarian evaluate when it could be worth killing the ghost in advance to prevent a greater harm.
- The **theory of dyadic morality** cannot be fully expressed by our current model.
- **Combine** deontology and utilitarianism on the lines of dyadic morality.
- Translate the deontological and utilitarian approaches to more complex settings like Mario Bros. where there are greater number of entities (agents and patients).
Thank you for your attention