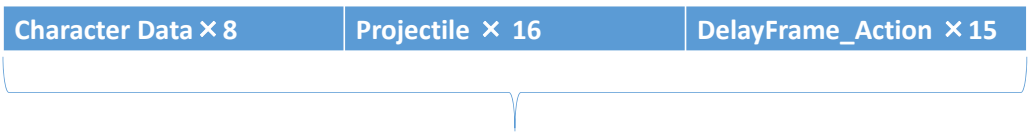
Input Representation and Network Architecture of LoadTorchWeightAI

Team FightingICE

March 8, 2017

Input (1/6): Representation

- Input
  - Character Data × 2 (self & opponent) × 4 frames
  - Projectiles × 2 (two most recent ones) × 2 (self & opponent) × 4 frames
  - Actions during DelayFrame × 15 frames



1376 units

Input (2/6): Character Data

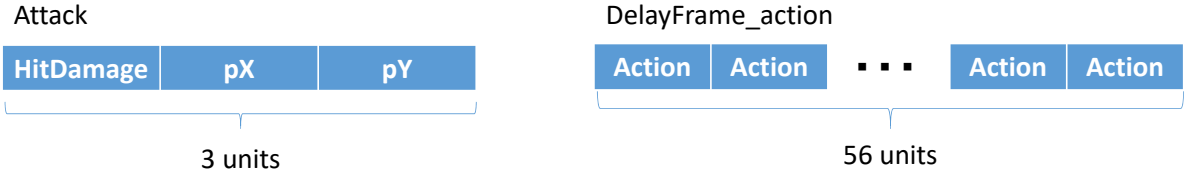
Character Data		
Xc	The x coordinate of the character	0～1
Yc	The y coordinate of the character	0～1
Energy	The energy of the character	0～1*
HP	The Hit Point of the character	0～1*
Action	The tuple of 56 bits, each indicating whether the action is being executed. (1 if the action is being executed, 0 otherwise)	0 or 1
RemainFrame	The number of remaining frames until the current action is completed	0～1

(*normalized by the maximum value in the training data, so this value might be more than 1 in practice.)

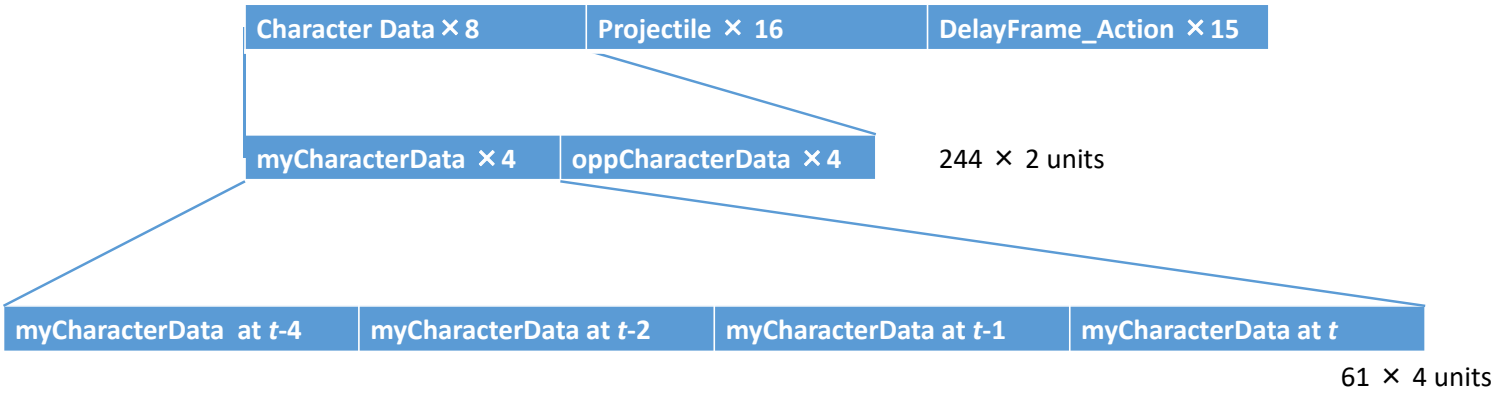
Input (3/6): Projectile & DelayFrame_action

Projectile		
HitDamage	The value of the damage when the attack (this projectile) hits the other character	0～1
pX	The x coordinate of the projectile	0～1
pY	The y coordinate of the projectile	0～1

DelayFrame_action		
Action	The action by this character during each of the previous 15 frames (Delay Frame) using the same representation as Action in the previous slide	0 or 1

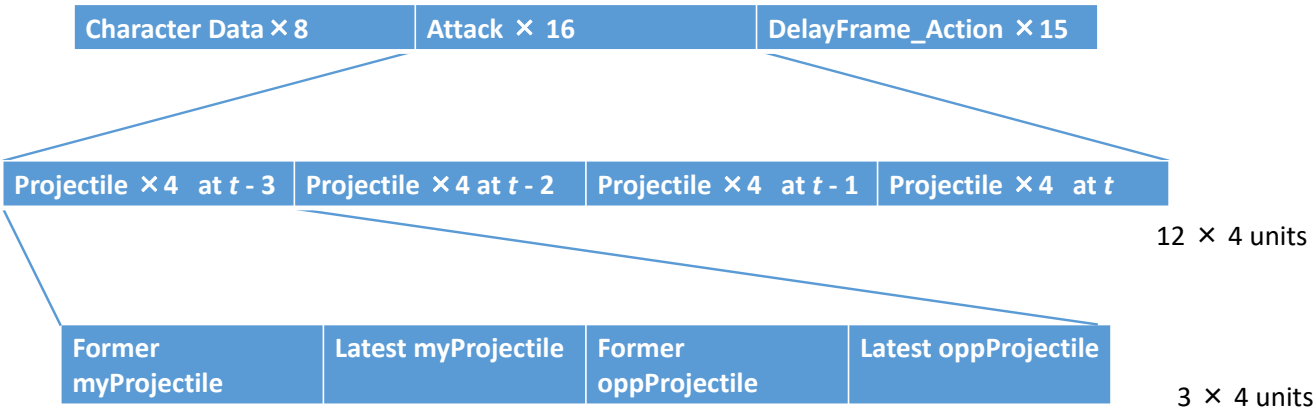


Input (4/6): Sequence of **CharacterData



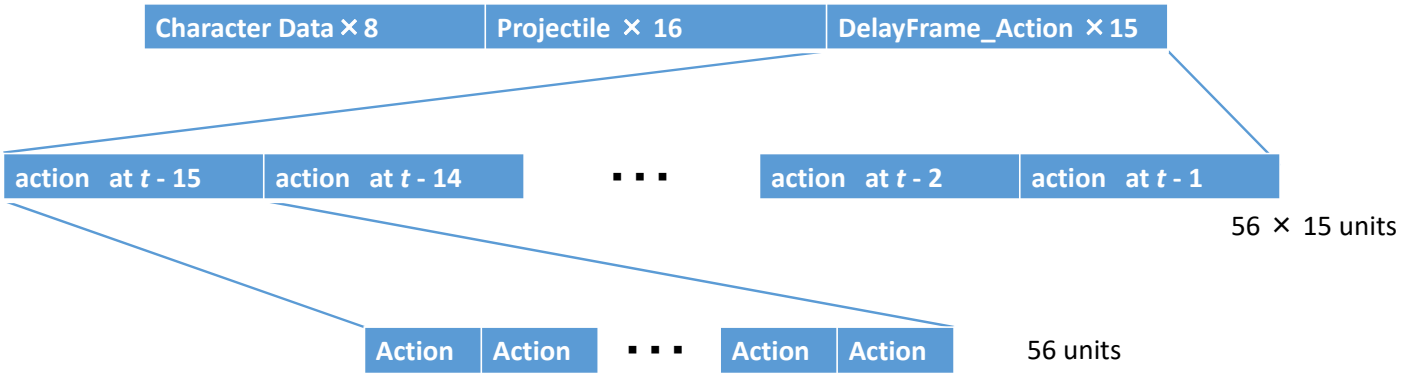
(the current frame is *t*)

Input (5/6): Sequence of Projectiles



(the current frame is *t*)

Input (6/6): Sequence of DelayFrame_Actions



(the current frame is *t*)