Using Slates in VW

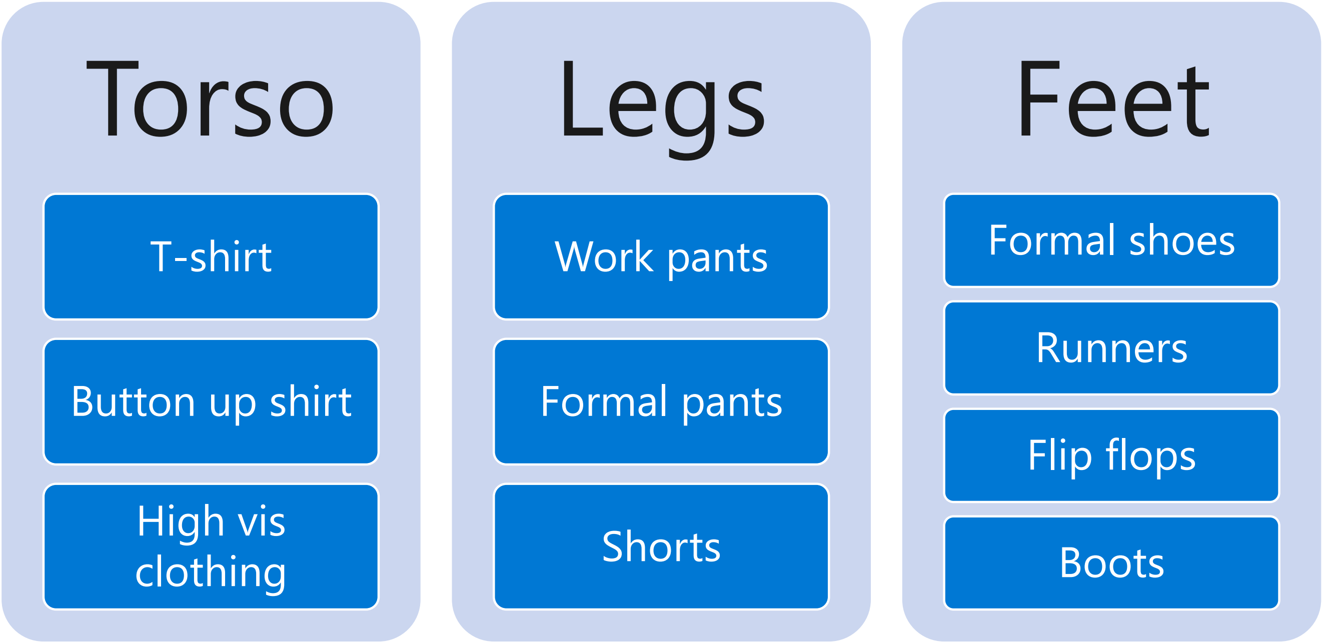
Adith Swaminathan & Jack Gerrits
Microsoft Research

Installing

```
pip install vowpalwabbit
```

Scenario

- Outfit optimization for job interview
- Slots are different clothing types
- Actions are the individual pieces of clothing for each slot
- We’re either applying for a corporate job or trade



Torso

- T-shirt
- Button up shirt
- High vis clothing

Legs

- Work pants
- Formal pants
- Shorts

Feet

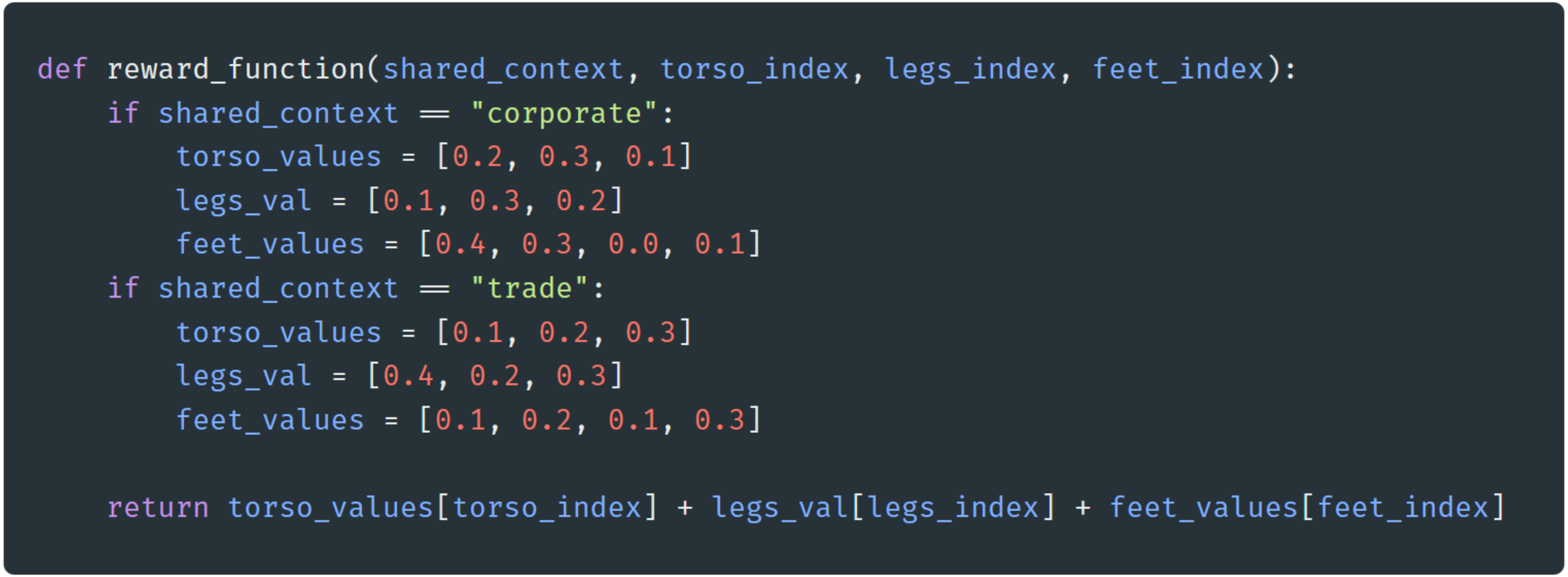
- Formal shoes
- Runners
- Flip flops
- Boots

Follow along here

aka.ms/slates_example_nb

Reward function

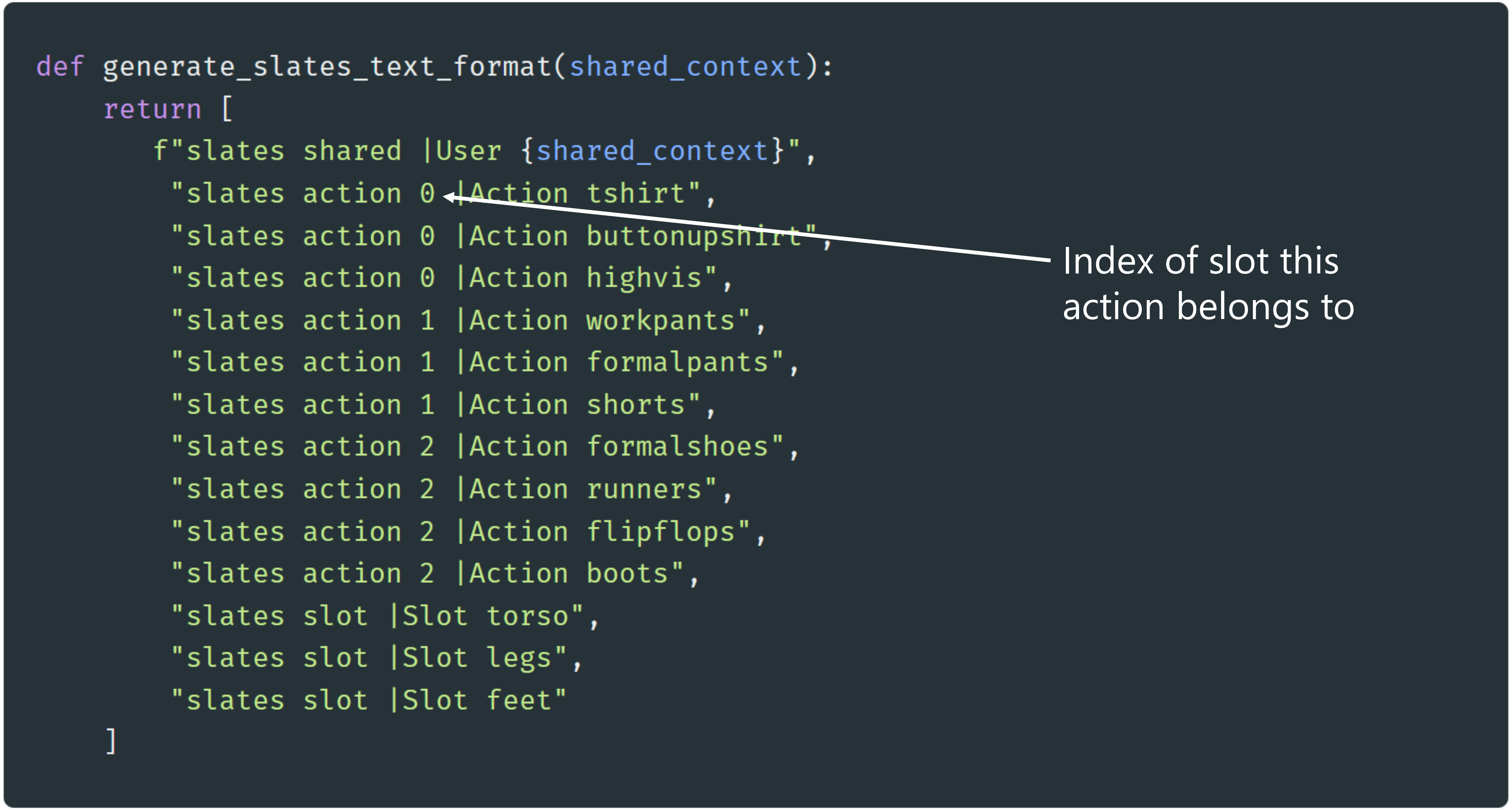
- Two contexts – corporate and trade
- In example noise is added around reward values



```
def reward_function(shared_context, torso_index, legs_index, feet_index):
    if shared_context == "corporate":
        torso_values = [0.2, 0.3, 0.1]
        legs_val = [0.1, 0.3, 0.2]
        feet_values = [0.4, 0.3, 0.0, 0.1]
    if shared_context == "trade":
        torso_values = [0.1, 0.2, 0.3]
        legs_val = [0.4, 0.2, 0.3]
        feet_values = [0.1, 0.2, 0.1, 0.3]

    return torso_values[torso_index] + legs_val[legs_index] + feet_values[feet_index]
```

aka.ms/slates_example_nb

Input format

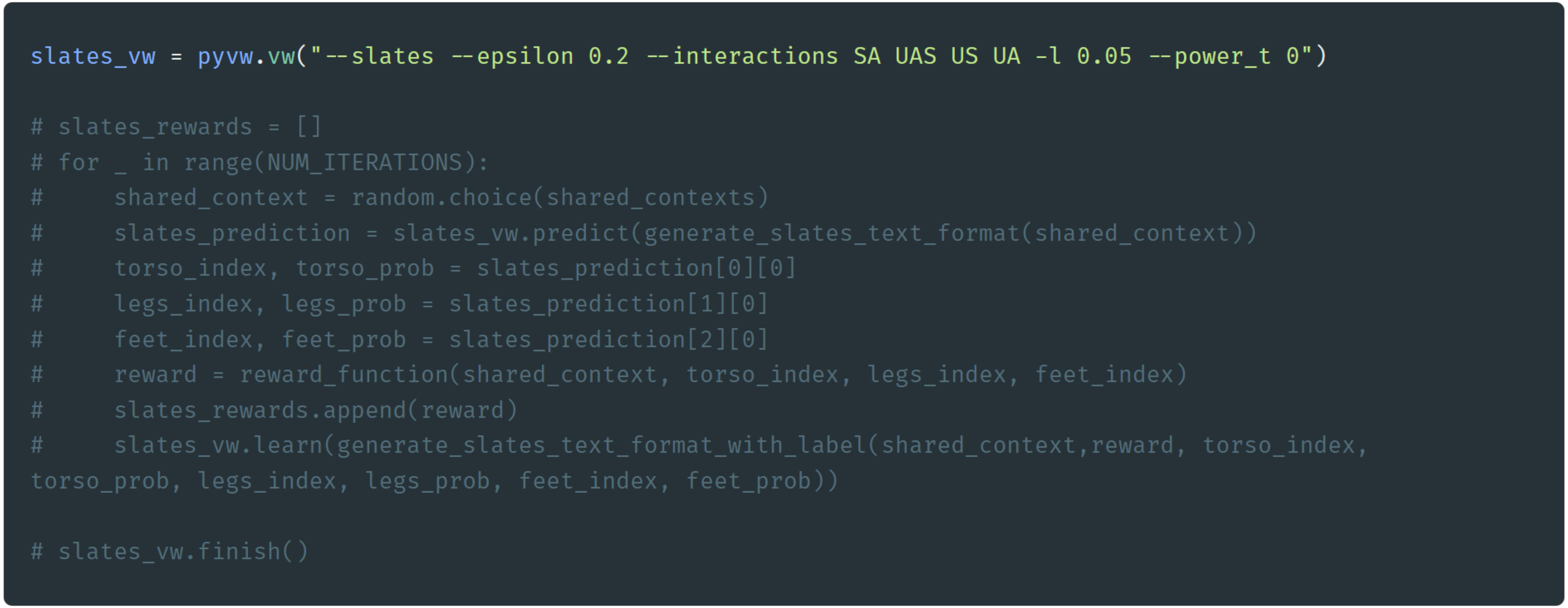


```
def generate_slates_text_format(shared_context):
    return [
       f"slates shared |User {shared_context}",
        "slates action 0 |Action tshirt",
        "slates action 0 |Action buttonupshirt",
        "slates action 0 |Action highvis",
        "slates action 1 |Action workpants",
        "slates action 1 |Action formalpants",
        "slates action 1 |Action shorts",
        "slates action 2 |Action formalshoes",
        "slates action 2 |Action runners",
        "slates action 2 |Action flipflops",
        "slates action 2 |Action boots",
        "slates slot |Slot torso",
        "slates slot |Slot legs",
        "slates slot |Slot feet"
    ]
```

Index of slot this action belongs to

aka.ms/slates_example_nb

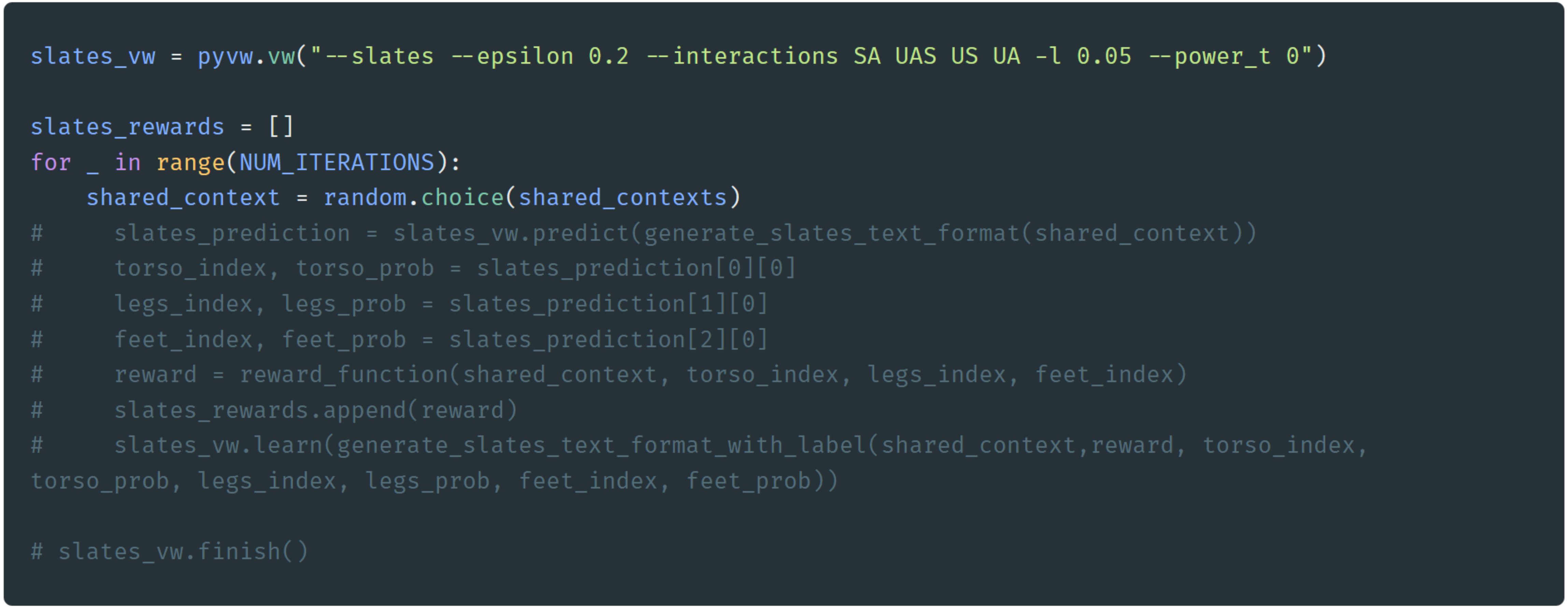
Simulator



```
slates_vw = pyvw.vw("--slates --epsilon 0.2 --interactions SA UAS US UA -l 0.05 --power_t 0")

# slates_rewards = []
# for _ in range(NUM_ITERATIONS):
#     shared_context = random.choice(shared_contexts)
#     slates_prediction = slates_vw.predict(generate_slates_text_format(shared_context))
#     torso_index, torso_prob = slates_prediction[0][0]
#     legs_index, legs_prob = slates_prediction[1][0]
#     feet_index, feet_prob = slates_prediction[2][0]
#     reward = reward_function(shared_context, torso_index, legs_index, feet_index)
#     slates_rewards.append(reward)
#     slates_vw.learn(generate_slates_text_format_with_label(shared_context,reward, torso_index,
torso_prob, legs_index, legs_prob, feet_index, feet_prob))

# slates_vw.finish()
```

aka.ms/slates_example_nb

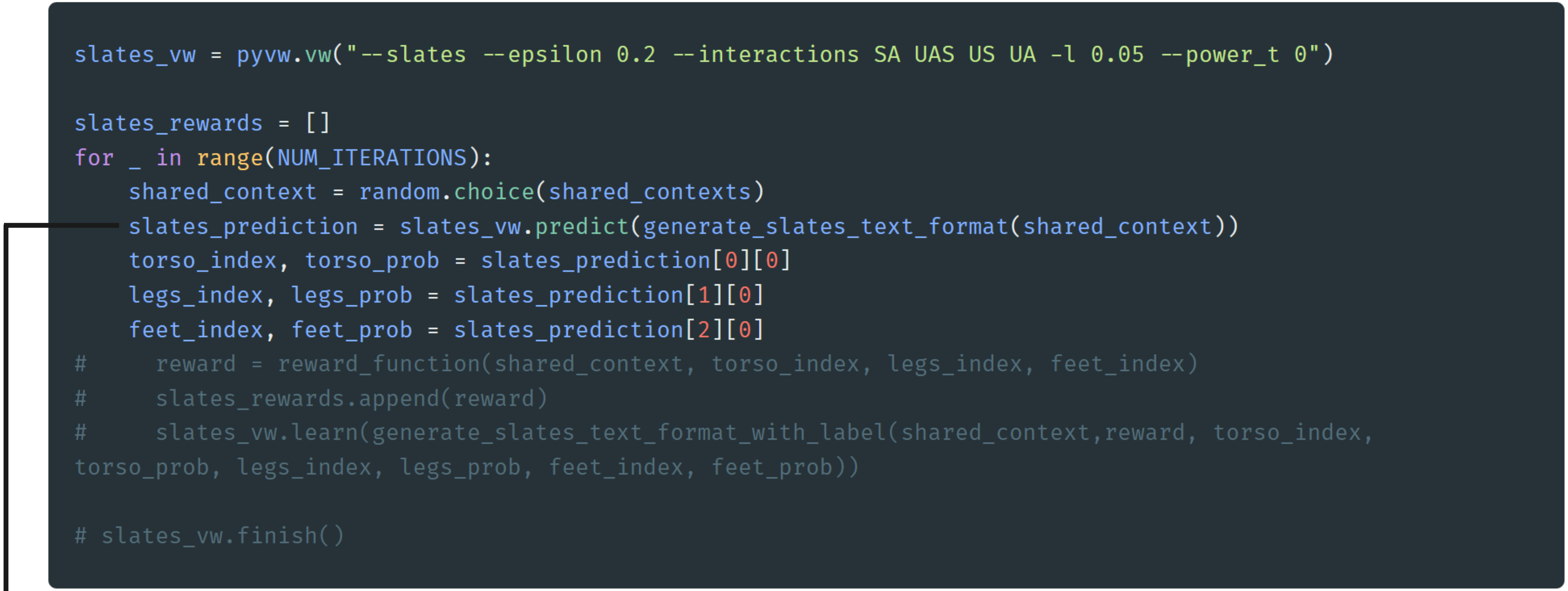
Simulator



```
slates_vw = pyvw.vw("--slates --epsilon 0.2 --interactions SA UAS US UA -l 0.05 --power_t 0")

slates_rewards = []
for _ in range(NUM_ITERATIONS):
    shared_context = random.choice(shared_contexts)
#     slates_prediction = slates_vw.predict(generate_slates_text_format(shared_context))
#     torso_index, torso_prob = slates_prediction[0][0]
#     legs_index, legs_prob = slates_prediction[1][0]
#     feet_index, feet_prob = slates_prediction[2][0]
#     reward = reward_function(shared_context, torso_index, legs_index, feet_index)
#     slates_rewards.append(reward)
#     slates_vw.learn(generate_slates_text_format_with_label(shared_context,reward, torso_index,
torso_prob, legs_index, legs_prob, feet_index, feet_prob))

# slates_vw.finish()
```

aka.ms/slates_example_nb

Simulator



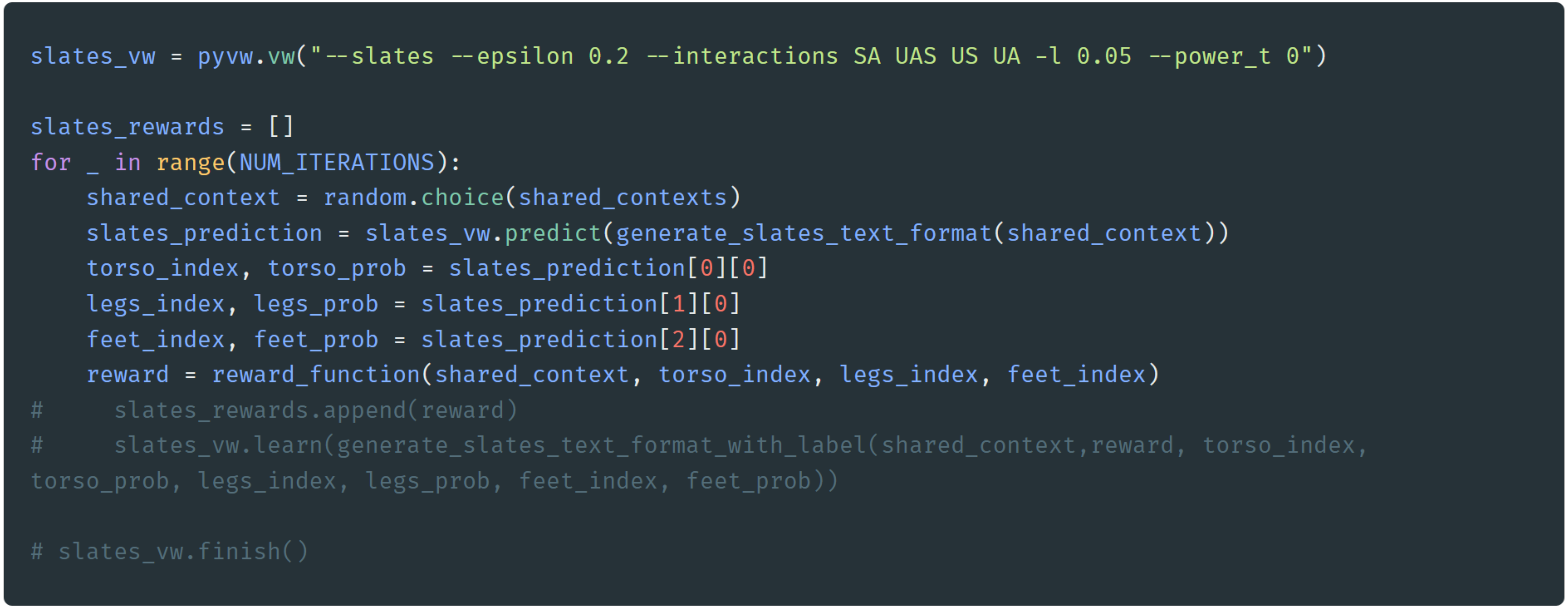
```python
slates_vw = pyvw.vw("--slates --epsilon 0.2 --interactions SA UAS US UA -l 0.05 --power_t 0")

slates_rewards = []
for _ in range(NUM_ITERATIONS):
    shared_context = random.choice(shared_contexts)
    slates_prediction = slates_vw.predict(generate_slates_text_format(shared_context))
    torso_index, torso_prob = slates_prediction[0][0]
    legs_index, legs_prob = slates_prediction[1][0]
    feet_index, feet_prob = slates_prediction[2][0]
#     reward = reward_function(shared_context, torso_index, legs_index, feet_index)
#     slates_rewards.append(reward)
#     slates_vw.learn(generate_slates_text_format_with_label(shared_context,reward, torso_index,
torso_prob, legs_index, legs_prob, feet_index, feet_prob))

# slates_vw.finish()
```

```
[[(0, 0.8675), (1, 0.067), (2, 0.067)],
 [(1, 0.067), (0, 0.867), (2, 0.067)],
 [(0, 0.85), (1, 0.05), (2, 0.05), (3, 0.05)]]
```

T-shirt
Formal pants
Formal shoes

aka.ms/slates_example_nb

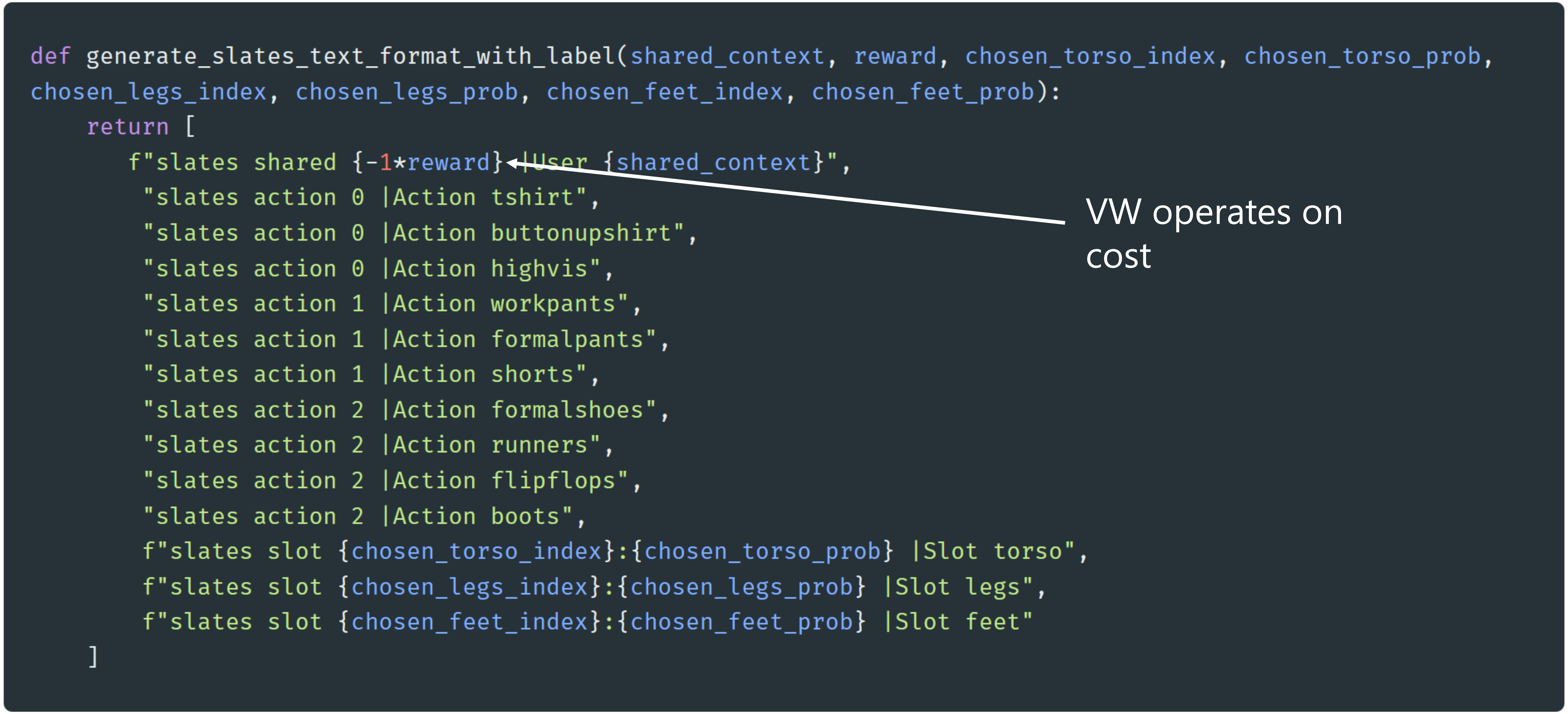
Simulator



```
slates_vw = pyvw.vw("--slates --epsilon 0.2 --interactions SA UAS US UA -l 0.05 --power_t 0")

slates_rewards = []
for _ in range(NUM_ITERATIONS):
    shared_context = random.choice(shared_contexts)
    slates_prediction = slates_vw.predict(generate_slates_text_format(shared_context))
    torso_index, torso_prob = slates_prediction[0][0]
    legs_index, legs_prob = slates_prediction[1][0]
    feet_index, feet_prob = slates_prediction[2][0]
    reward = reward_function(shared_context, torso_index, legs_index, feet_index)
#     slates_rewards.append(reward)
#     slates_vw.learn(generate_slates_text_format_with_label(shared_context,reward, torso_index,
torso_prob, legs_index, legs_prob, feet_index, feet_prob))

# slates_vw.finish()
```

aka.ms/slates_example_nb

Labels

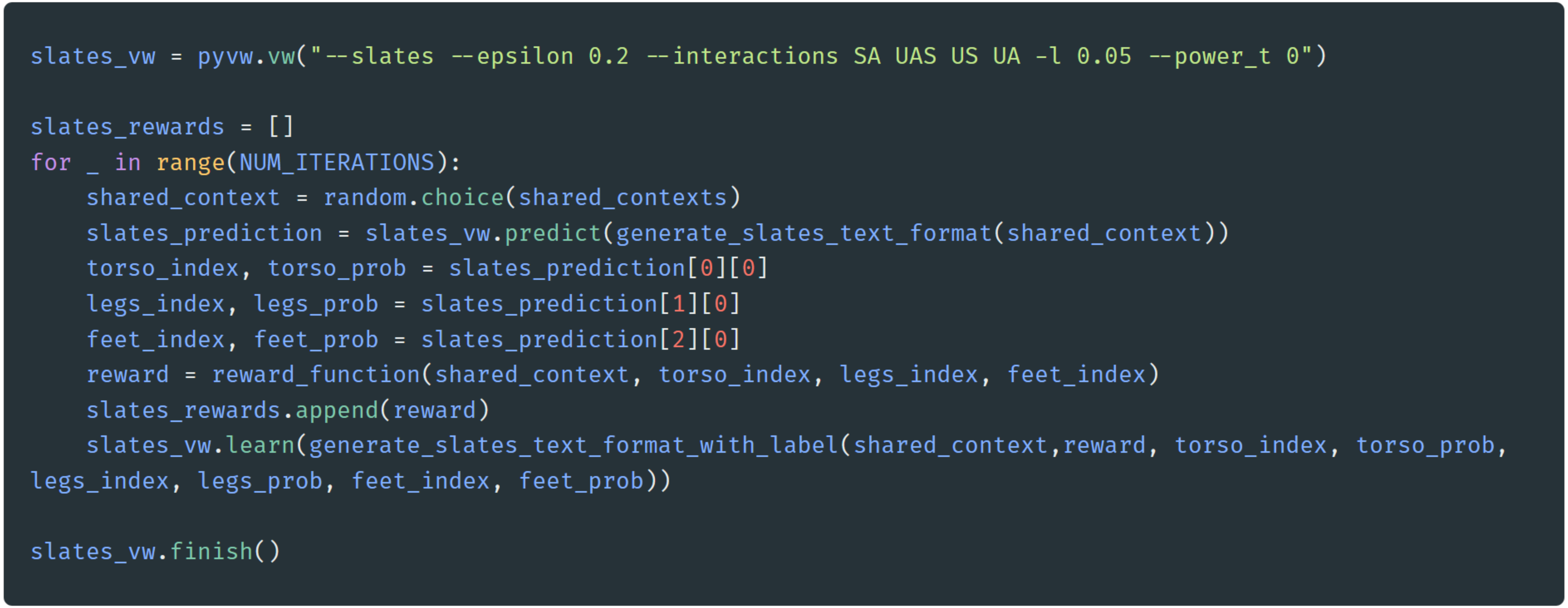


```
def generate_slates_text_format_with_label(shared_context, reward, chosen_torso_index, chosen_torso_prob,
chosen_legs_index, chosen_legs_prob, chosen_feet_index, chosen_feet_prob):
    return [
       f"slates shared {-1*reward} |User {shared_context}",
        "slates action 0 |Action tshirt",
        "slates action 0 |Action buttonupshirt",
        "slates action 0 |Action highvis",
        "slates action 1 |Action workpants",
        "slates action 1 |Action formalpants",
        "slates action 1 |Action shorts",
        "slates action 2 |Action formalshoes",
        "slates action 2 |Action runners",
        "slates action 2 |Action flipflops",
        "slates action 2 |Action boots",
        f"slates slot {chosen_torso_index}:{chosen_torso_prob} |Slot torso",
        f"slates slot {chosen_legs_index}:{chosen_legs_prob} |Slot legs",
        f"slates slot {chosen_feet_index}:{chosen_feet_prob} |Slot feet"
    ]
```

VW operates on cost

aka.ms/slates_example_nb

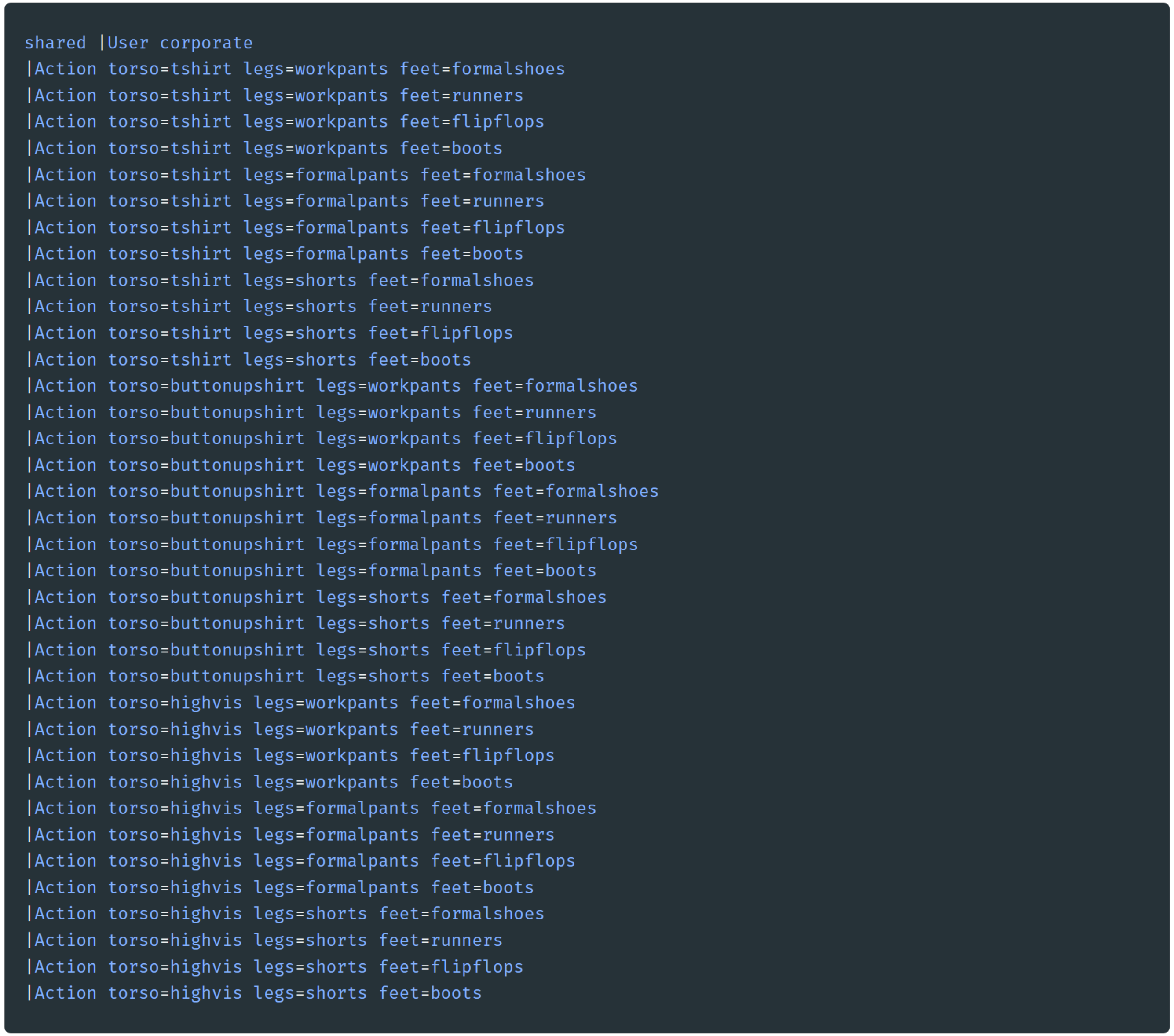
Simulator



```
slates_vw = pyvw.vw("--slates --epsilon 0.2 --interactions SA UAS US UA -l 0.05 --power_t 0")

slates_rewards = []
for _ in range(NUM_ITERATIONS):
    shared_context = random.choice(shared_contexts)
    slates_prediction = slates_vw.predict(generate_slates_text_format(shared_context))
    torso_index, torso_prob = slates_prediction[0][0]
    legs_index, legs_prob = slates_prediction[1][0]
    feet_index, feet_prob = slates_prediction[2][0]
    reward = reward_function(shared_context, torso_index, legs_index, feet_index)
    slates_rewards.append(reward)
    slates_vw.learn(generate_slates_text_format_with_label(shared_context,reward, torso_index, torso_prob,
legs_index, legs_prob, feet_index, feet_prob))

slates_vw.finish()
```

aka.ms/slates_example_nb

CB Equivalent



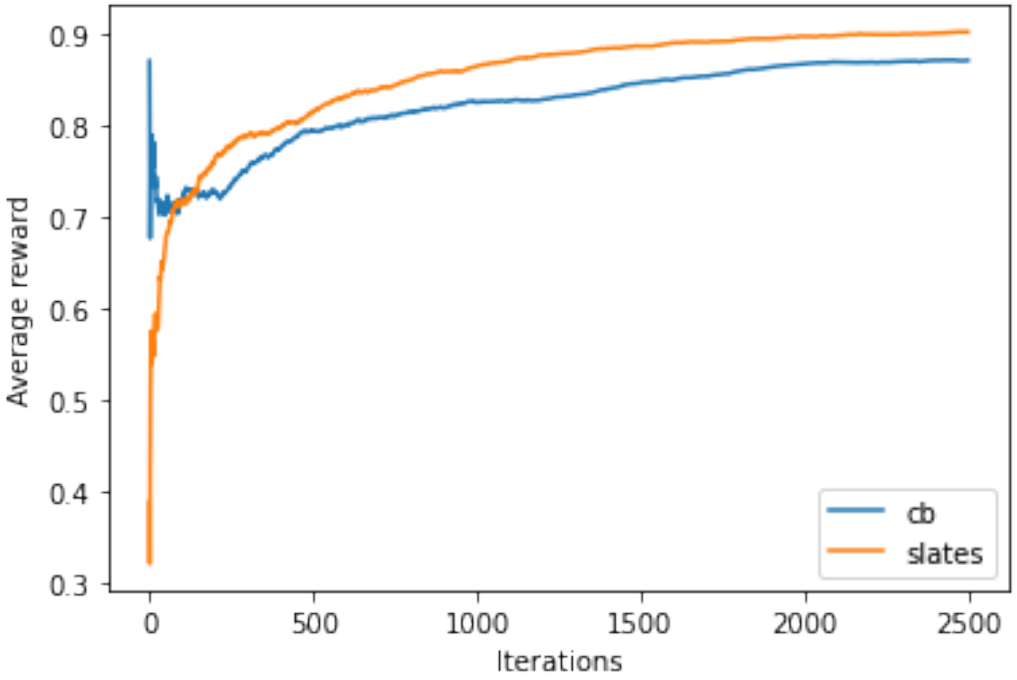
```
shared |User corporate
|Action torso=tshirt legs=workpants feet=formalshoes
|Action torso=tshirt legs=workpants feet=runners
|Action torso=tshirt legs=workpants feet=flipflops
|Action torso=tshirt legs=workpants feet=boots
|Action torso=tshirt legs=formalpants feet=formalshoes
|Action torso=tshirt legs=formalpants feet=runners
|Action torso=tshirt legs=formalpants feet=flipflops
|Action torso=tshirt legs=formalpants feet=boots
|Action torso=tshirt legs=shorts feet=formalshoes
|Action torso=tshirt legs=shorts feet=runners
|Action torso=tshirt legs=shorts feet=flipflops
|Action torso=tshirt legs=shorts feet=boots
|Action torso=buttonupshirt legs=workpants feet=formalshoes
|Action torso=buttonupshirt legs=workpants feet=runners
|Action torso=buttonupshirt legs=workpants feet=flipflops
|Action torso=buttonupshirt legs=workpants feet=boots
|Action torso=buttonupshirt legs=formalpants feet=formalshoes
|Action torso=buttonupshirt legs=formalpants feet=runners
|Action torso=buttonupshirt legs=formalpants feet=flipflops
|Action torso=buttonupshirt legs=formalpants feet=boots
|Action torso=buttonupshirt legs=shorts feet=formalshoes
|Action torso=buttonupshirt legs=shorts feet=runners
|Action torso=buttonupshirt legs=shorts feet=flipflops
|Action torso=buttonupshirt legs=shorts feet=boots
|Action torso=highvis legs=workpants feet=formalshoes
|Action torso=highvis legs=workpants feet=runners
|Action torso=highvis legs=workpants feet=flipflops
|Action torso=highvis legs=workpants feet=boots
|Action torso=highvis legs=formalpants feet=formalshoes
|Action torso=highvis legs=formalpants feet=runners
|Action torso=highvis legs=formalpants feet=flipflops
|Action torso=highvis legs=formalpants feet=boots
|Action torso=highvis legs=shorts feet=formalshoes
|Action torso=highvis legs=shorts feet=runners
|Action torso=highvis legs=shorts feet=flipflops
|Action torso=highvis legs=shorts feet=boots
```

$$3 * 3 * 4 + 1 = 37$$

aka.ms/slates_example_nb

Performance

- Equivalent between CB and slates
  - Because the CB version is combinatorially large it will quickly become difficult to solve



aka.ms/slates_example_nb

Slates

- Slates allows you to more clearly and efficiently express multi-slot optimization problems
- Potential future extensions
  - Ranking problems
  - Dependent action sets
  - Batch mode
- Slates is in VW 8.9
- Paper: https://arxiv.org/abs/1605.04812
- Wiki: https://github.com/VowpalWabbit/vowpal_wabbit/wiki/Slates

aka.ms/slates_example_nb