Dreaming to Prove

Kristóf Szabó, Zsolt Zombori

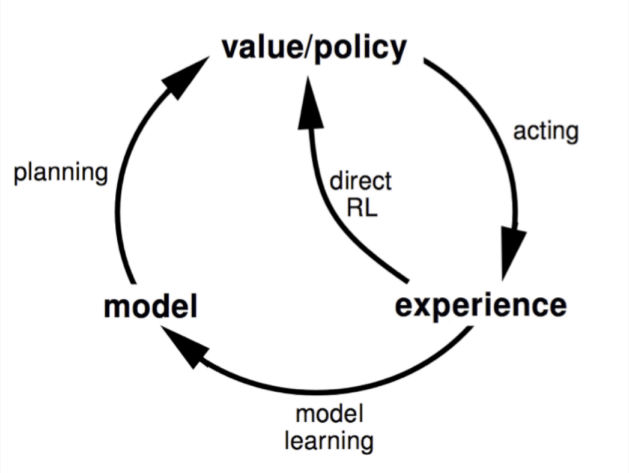
Acknowledgements

This work was supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00002), the Hungarian National Excellence Grant 2018-1.2.1-NKP-00008 and by the Hungarian Ministry of Innovation and Technology NRDI Office within the framework of the Artificial Intelligence National Laboratory Program.

PROJECT FINANCED
FROM THE NRDI FUND

Dreaming to prove

The main concept is that the **agent** not only needs to learn the best **policy** but is also asked to accurately imitate the **dynamics** of theorem proving.



Dreamer

Parts of the dreamer algorithm

- Encoder/decoder pair
- Dynamics model
- Value and policy model (Actor critic)

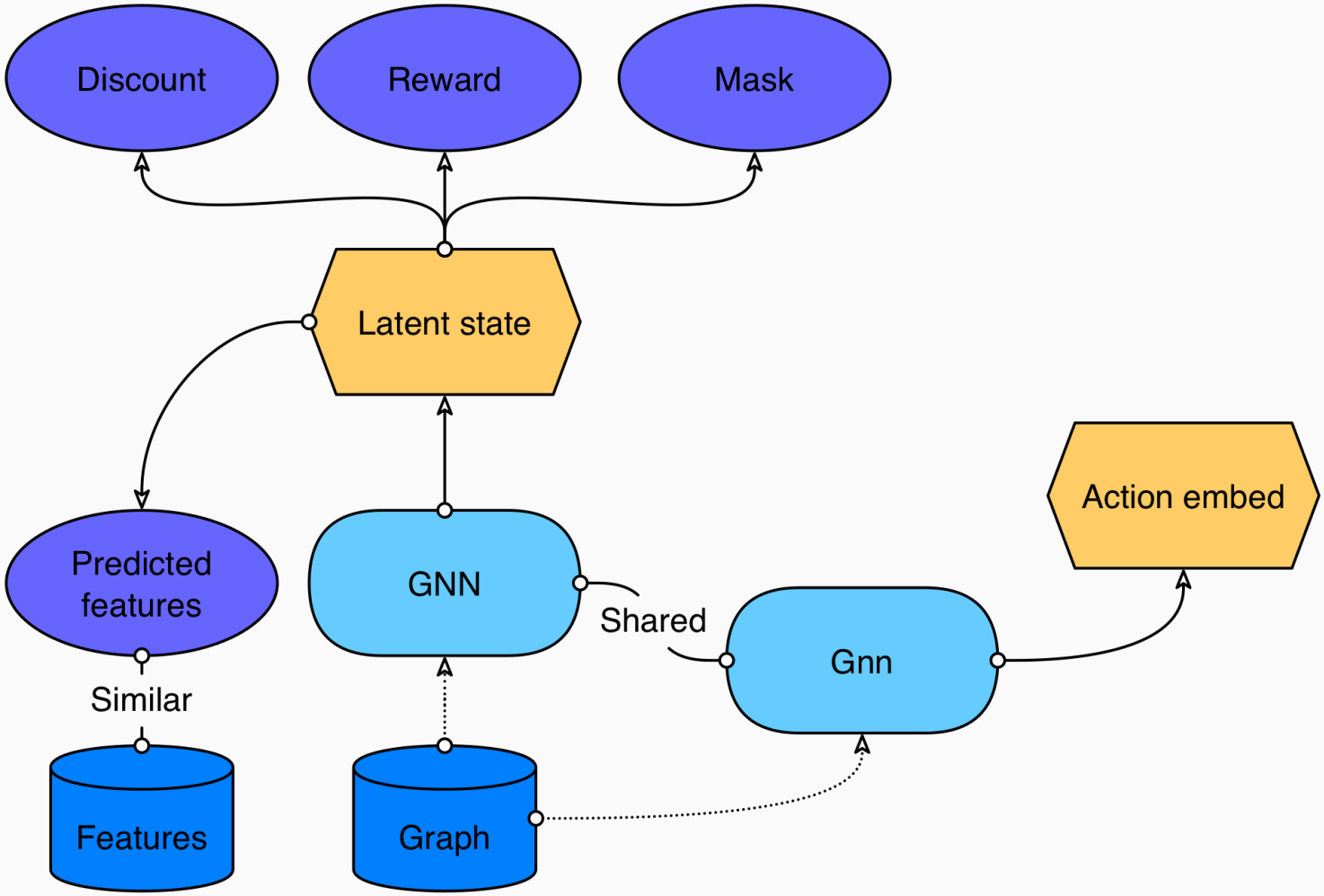
Since it learns the dynamics of the environment, the policy can be trained without the environment.

**Mastering Atari with Discrete World Models [Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, Jimmy Ba]*

Intuitions and motivations

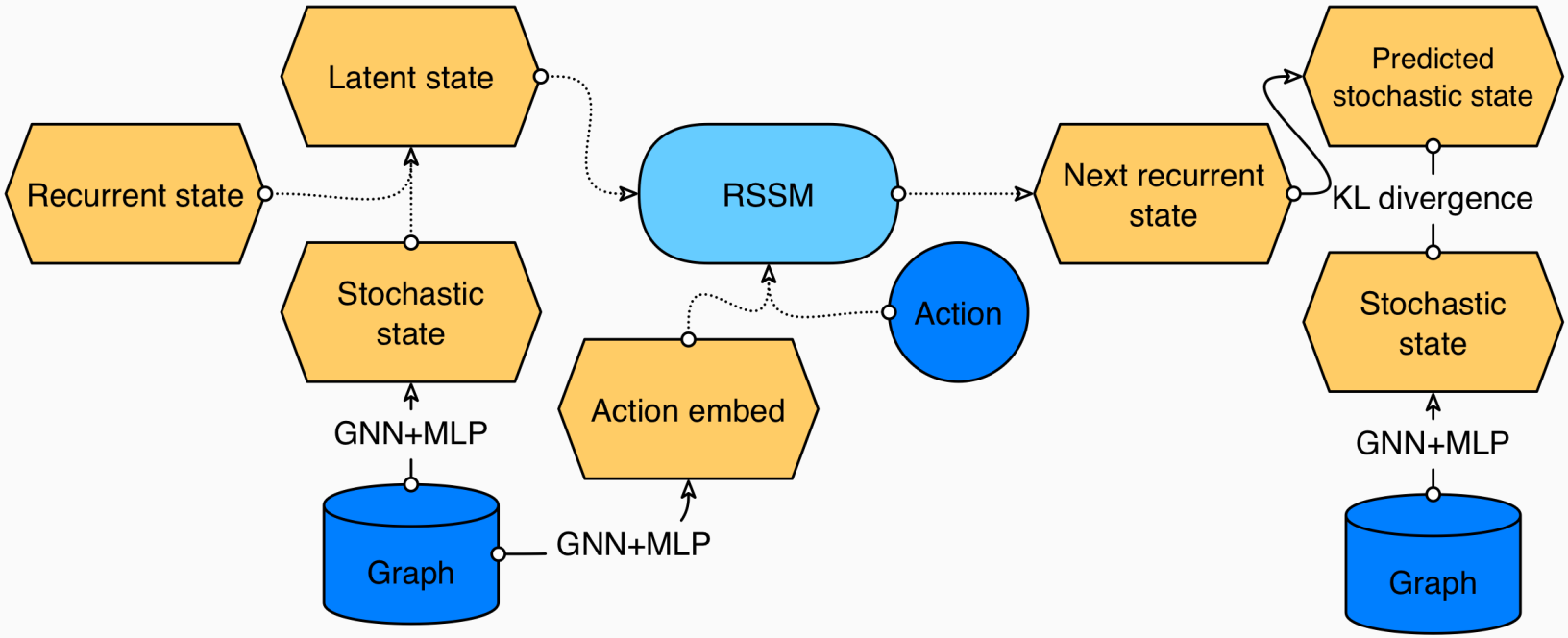
- Repeatedly querying the environment is **time**-consuming.
  1. We save time due to the smaller amount of interaction required with the **environment**.
  2. Training the policy without the **agent** decoding the state at each step accelerates the training process.
- We are not fully dependent on reward signals.
  1. In theorem proving, positive rewards are rare.
  2. By introducing several objectives, we can expect the model to generalise better to new situations.

Encoder/Decoder

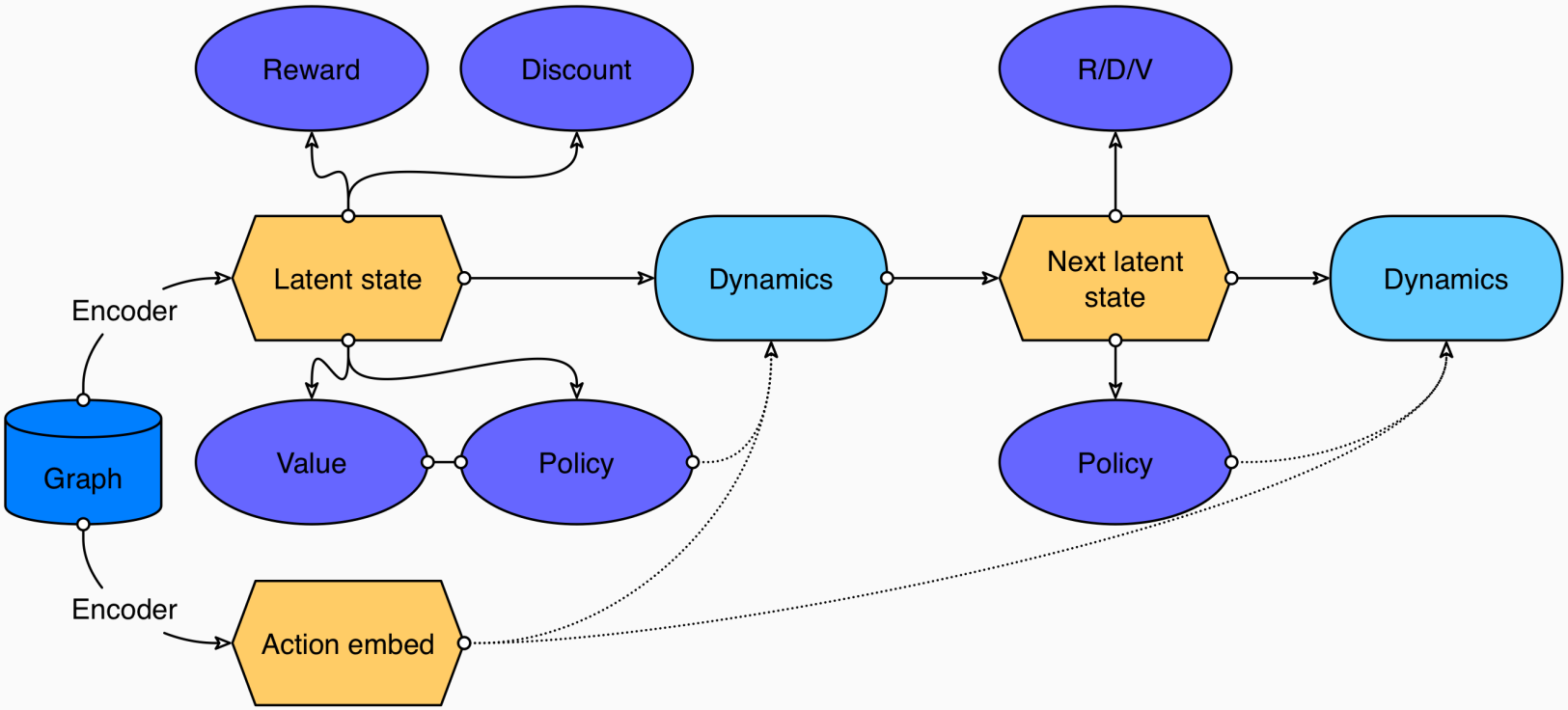


*Property invariant embedding for automated reasoning. [Miroslav Olsák, Cezary Kaliszyk, Josef Urban]

Dynamics


Latent state
Recurrent state
RSSM
Next recurrent state
Predicted stochastic state
KL divergence
Stochastic state
Action
Action embed
Stochastic state
GNN+MLP
GNN+MLP
Graph
GNN+MLP
Graph

Policy


Reward
Discount
R/D/V
Latent state
Dynamics
Next latent state
Dynamics
Encoder
Graph
Value
Policy
Policy
Encoder
Action embed

Technicals

- Sampling method specialized for batching, reward balancing.
- The mask loss teaches the model to avoid invalid steps. Originally, each invalid action had to be queried.

Thank You for Listening