

# Creative Bandit

Jerry Zhu

Dept. of Computer Sciences, Univ. of Wisconsin-Madison

# Creative block

- Writers, musicians, artists...
- Narrow scope of this talk:  
Think as many properties as you can about [penguins](#).

# Unblock creativity by “nudging”

- Demo

Type a property that is true of **penguin**.

There is no hint yet!

You still have **free HINTS**! If the hints are not helpful, click '**Get Hints**' button one more time!

Type a property that is true of **penguin**.

Hint: penguin

You still have **free HINTS!** If the hints are not helpful, click '**Get Hints**' button one more time!

Type a property that is true of **penguin**.

Hint: ate, consumes, eat, buys, sleeps

You still have **free HINTS!** If the hints are not helpful, click '**Get Hints**' button one more time!

Submit

Get Hints

Type a property that is true of **penguin**.

Hint: ate, consumes, eat, buys, sleeps

You still have **free HINTS!** If the hints are not helpful, click '**Get Hints**' button one more time!

Sleeps while standing up

Submit

Get Hints

Type a property that is true of **penguin**.

Hint: cellphone, safer, smallpox, rush, clarifies

You still have **free HINTS!** If the hints are not helpful, click '**Get Hints**' button one more time!

Submit

Get Hints



Type a property that is true of **penguin**.

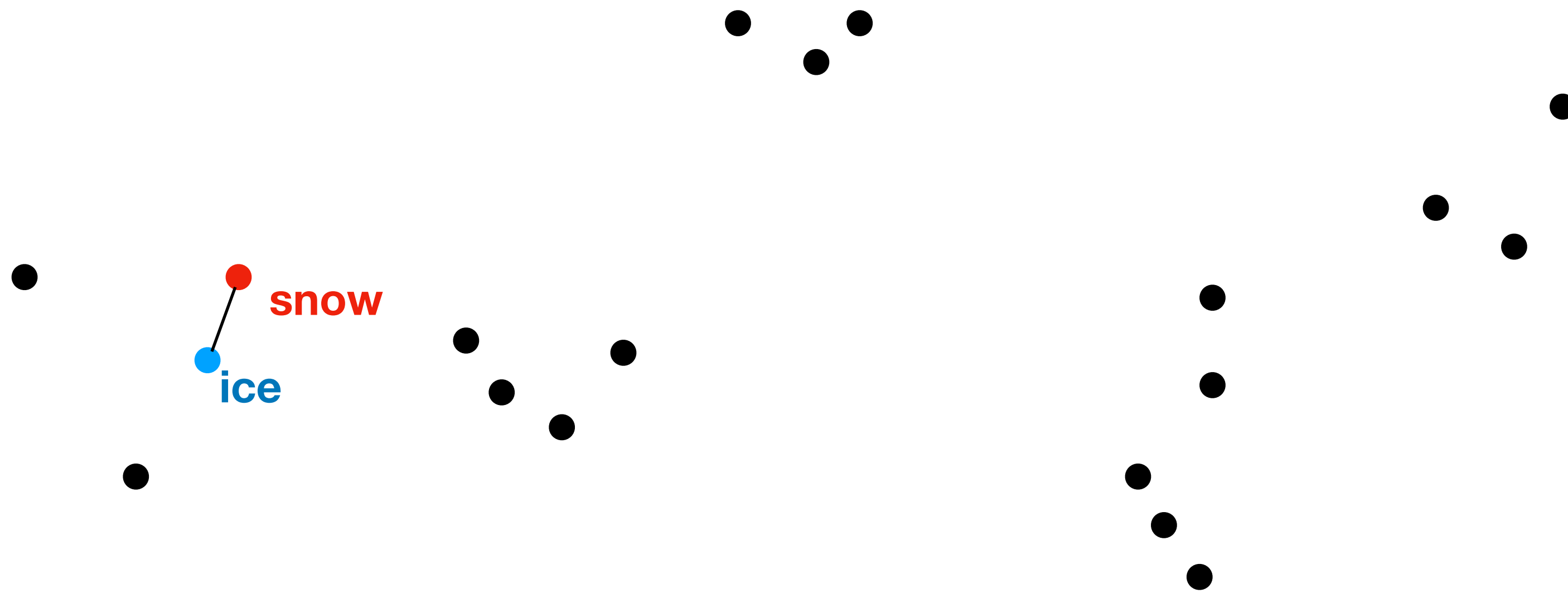
Hint: cellphone, safer, smallpox, rush, clarifies

You still have ~~free HINTS!~~ If the hints are not helpful, click '**Get Hints**' button one more time!

# There are many types of nudges

- (Not considered here): cartoons, music, hot shower...
- Nearby words to what you have said (in semantic embedding space)

$$w^* = \arg \min_w \text{dist}(\text{ice}, w)$$

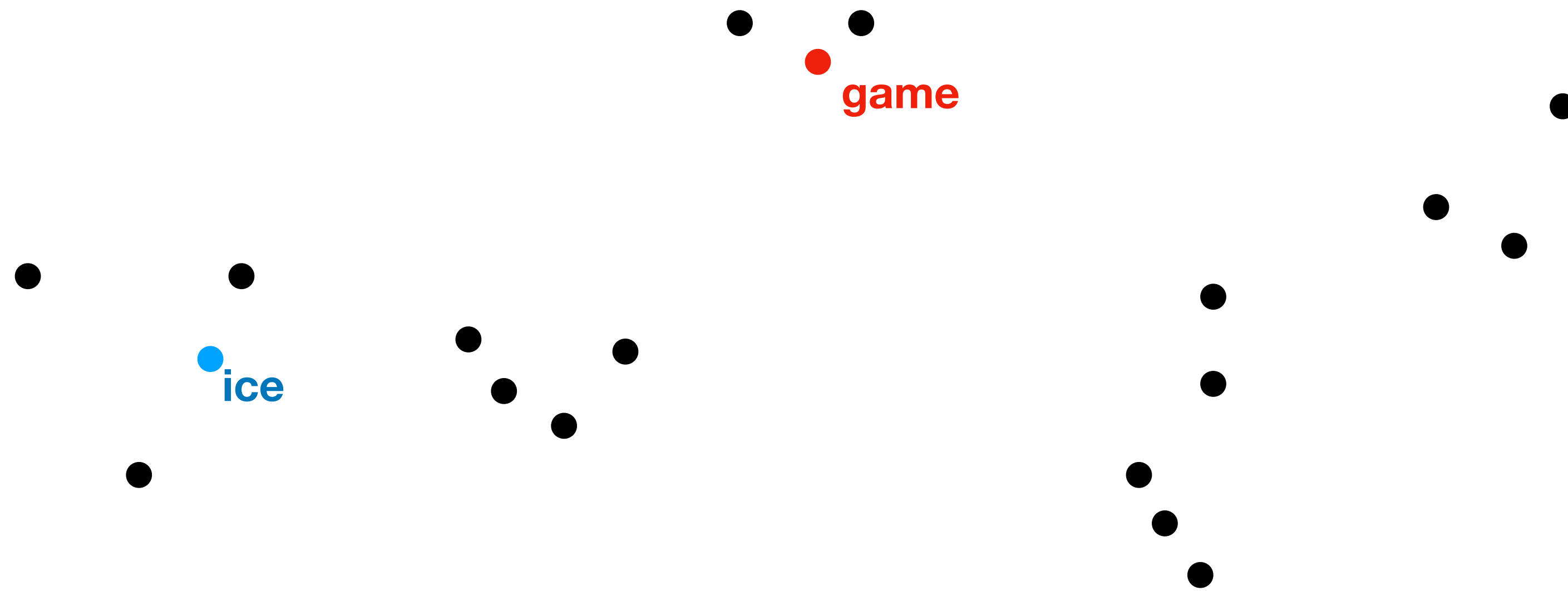


snow, hockey, water, icy, frozen

# There are many types of nudges

- Random words (sampled according to English word frequency)

$$w^* \sim P(\text{vocabulary})$$



game, button, freighter, disagreed, fait

# There are many types of nudges

- Large language model (eg. GPT) generation

“A property of penguin: **lives on ice**

Another property of penguin: **cannot fly**

Another property of penguin: **eats fish**

Another property of penguin:” **has a beak**

**Another property of penguin: has feathers**

**Another property of penguin: lays eggs**

# AI has a dilemma

- Different nudges may have different effectiveness on you
- Can AI give you the most effective nudge type?
  - AI has to try a nudge type sufficiently often to estimate its effectiveness
  - But if AI tries too much, you could have suffered creative block more than you should
- Fundamental dilemma: exploration vs. exploitation

# Multi-armed bandit

- $k$  nudge types =  $k$  arms
- The  $t$ -th time ( $t = 1 \dots T$ ) you request a nudge, AI chooses nudge type  $a_t \in \{1 \dots k\}$
- Upon receiving the nudge, you might:
  - be inspired to write some sentences (loss to AI is  $\ell_{t,a_t} = 0$ ); or
  - unable to write any sentences ( $\ell_{t,a_t} = 1$ )

# Multi-armed bandit

- If AI always chooses nudge type  $i$ , cumulative loss  $\sum_{t=1}^T \ell_{t,i}$
- The best nudge type in hindsight  $a^* = \arg \min_{i=1\dots k} \sum_{t=1}^T \ell_{t,i}$
- Can AI deliver nudge sequence  $a_1, \dots, a_T$  whose cumulative loss  $\sum_{t=1}^T \ell_{t,a_t}$  approaches that of the best nudge type?

# The EXP3 Algorithm

1. Initialize  $w_{11} = \dots = w_{1k} = 1$
2. For  $t = 1 \dots T$
3. 
$$p_{ti} = \frac{w_{ti}}{\sum_{j=1}^k w_{tj}}, \quad \forall i = 1 \dots k$$
4. Deliver nudge  $a_t \sim \text{multi}(p_{t1}, \dots, p_{tk})$ , receive loss  $\ell_{t,a_t}$
5. 
$$w_{t+1,a_t} = w_{t,a_t} \exp\left(-\eta \frac{\ell_{t,a_t}}{p_{t,a_t}}\right)$$
 where  $\eta = \sqrt{\frac{2 \log k}{Tk}}$



# EXP3 regret guarantee

$$\mathbb{E} \left[ \sum_{t=1}^T \ell_{t,a_t} \right] \leq \sum_{t=1}^T \ell_{t,a^*} + \sqrt{2Tk \log k}$$

- In words, if AI follows the EXP3 algorithm, it will be almost as good as always choosing the best nudge in hindsight.
- That's our system
- Human experiments on-going

# Things bandit cannot do (future work)

- Stateful modeling of humans
  - Reinforcement learning
- Explaining why any arm works (or not)
- Automatically coming up with new arms
  - Can potentially do composition of arms