Decoding continued + Phrase-based translation

The IBM Models

The IBM Models

- Fertility probabilities.

The IBM Models

- Fertility probabilities.
- Word translation probabilities.

The IBM Models

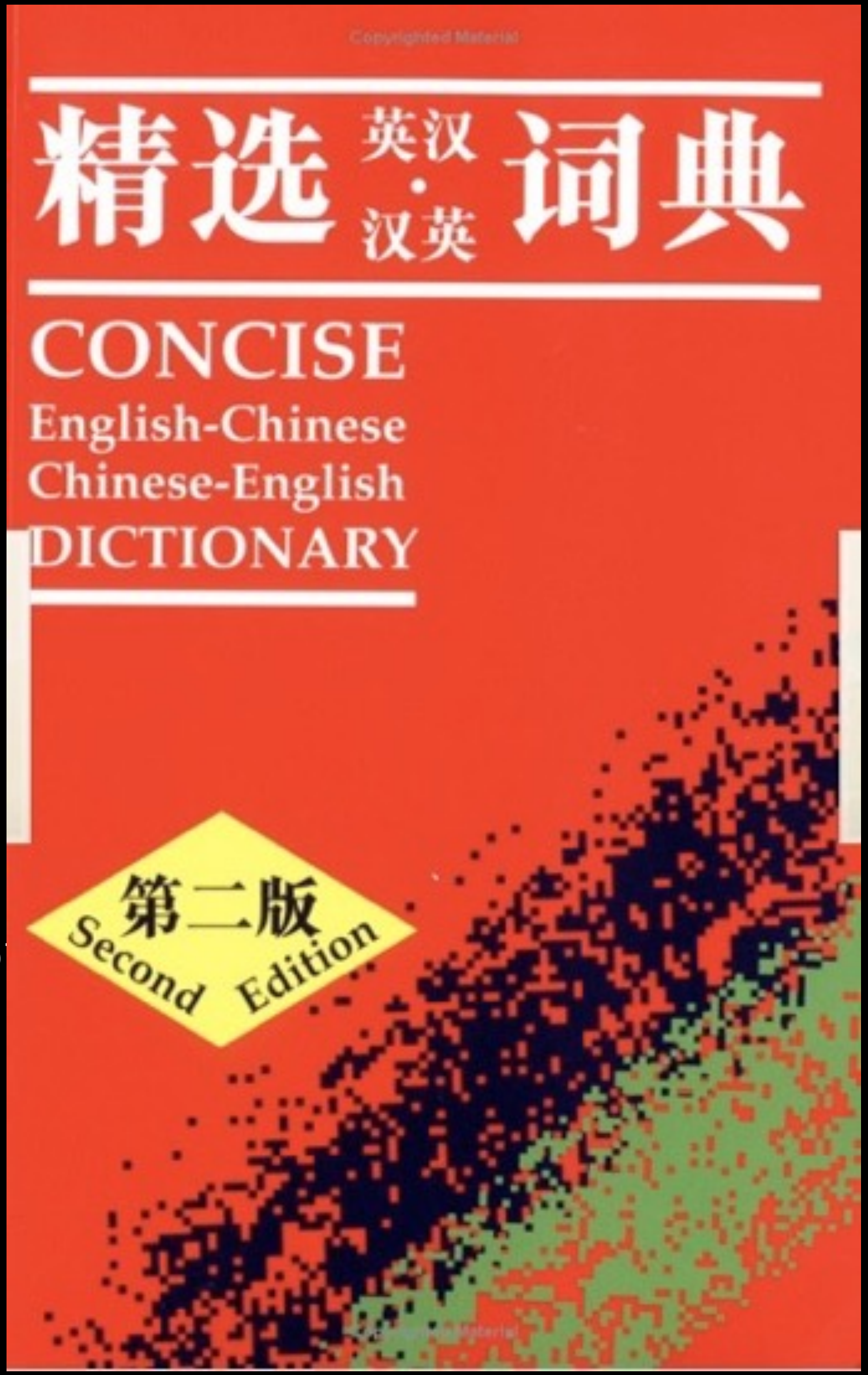
- Fertility probabilities.
- Word translation probabilities.
- Distortion probabilities.

The IBM Models

- Fertility probabilities.
- Word translation probabilities.
- Distortion probabilities.
- Some problems:
  - Weak reordering model -- output is not fluent.
  - Many decisions -- many things can go wrong.

The IBM Models

- Fertility probabilities.
- Word translation probabilities.
- Distortion probabilities.
- Some problems:
  - Weak reordering model -- outp
  - Many decisions -- many things



The IBM Models

- Fertility probabilities.
- Word translation probabilities.
- Distortion probabilities.
- Some problems:
  - Weak reordering model -- output is not fluent.
  - Many decisions -- many things can go wrong.

IBM Model 4

Although north wind howls , but sky still very clear .
虽然 北 风 呼啸 ， 但 天空 依然 十分 清澈 。 ε
⊥
⊥
虽然 北 风 呼啸 ， 天空 天空 依然 清澈 。 ε ε
However north wind strong , the sky remained clear . under the
However , the sky remained clear under the strong north wind .

Tradeoffs: Modeling v. Learning

	Lexical Translation	Local ordering dependency	Fertility	Convex	Tractable Exact Inference
IBM Model 1	✔	✘	✘	✔	✔
HMM	✔	✔	✘	✘	✔
IBM Model 4	✔	✔	✔	✘	✘

Tradeoffs: Modeling v. Learning

Lesson:
Trade exactness
for expressivity

	Lexical Translation	Local ordering dependency	Fertility	Convex	Tractable Exact Inference
IBM Model 1	✔	✘	✘	✔	✔
HMM	✔	✔	✘	✘	✔
IBM Model 4	✔	✔	✔	✘	✘

IBM Model 4

However , the sky remained clear under the strong north wind .

What are some things this model doesn’t account for?

What are some things this model doesn't account for?

Phrase-based Models

Although north wind howls , but sky still very clear .

虽然 北 风 呼啸 ，但 天空 依然 十分 清澈 。

Phrase-based Models

Although north wind howls , but sky still very clear .

虽然	北 风 呼啸	，但	天空 依然 十分 清澈	。

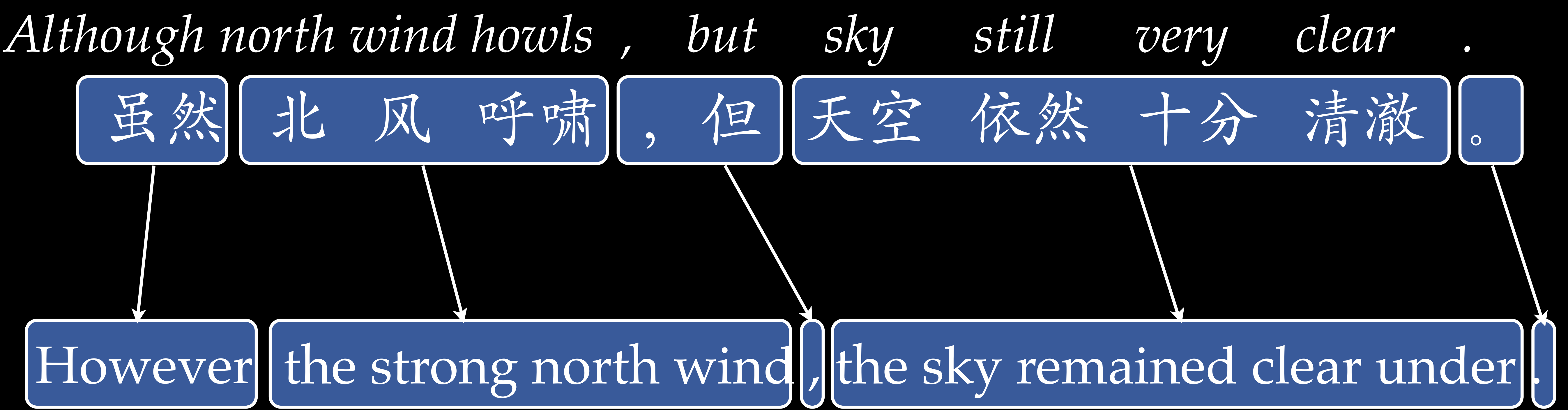
Phrase-based Models

Although north wind howls , but sky still very clear .

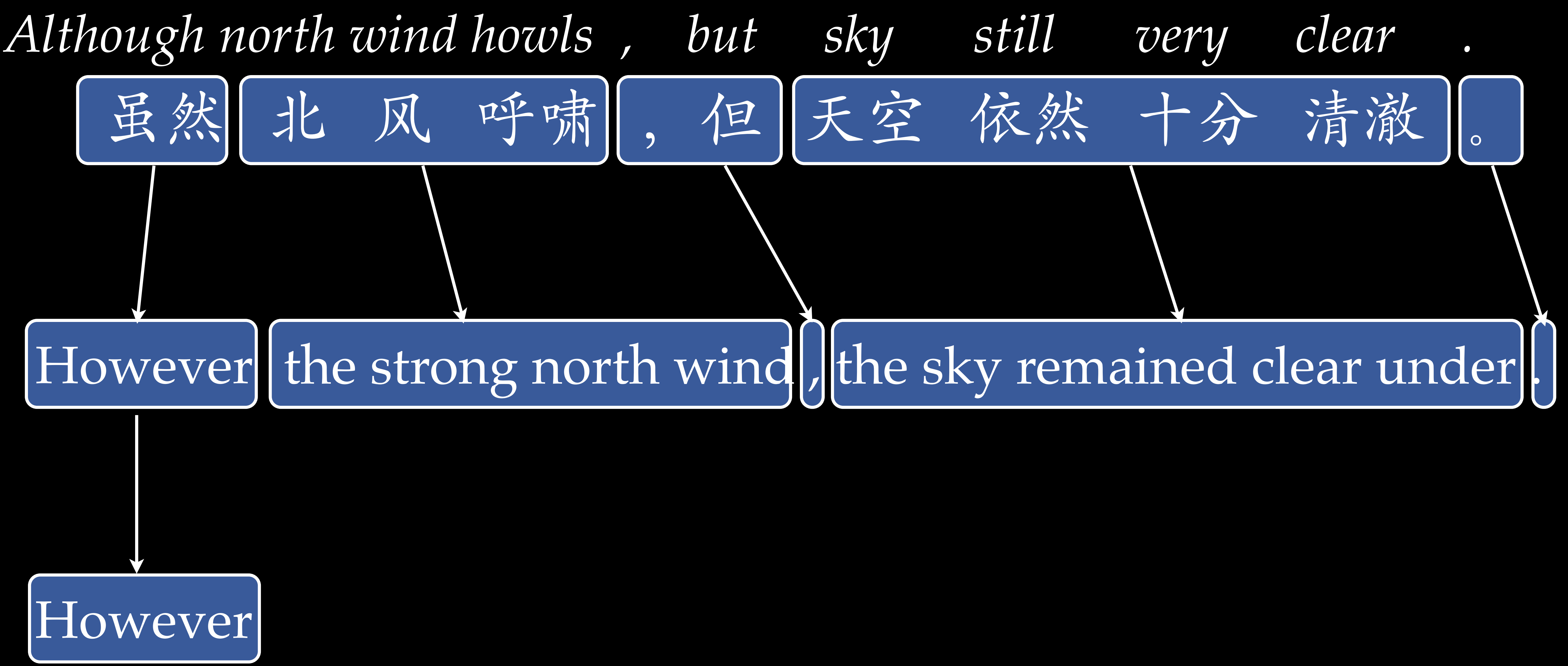
虽然	北 风 呼啸	，但	天空 依然 十分 清澈	。

However

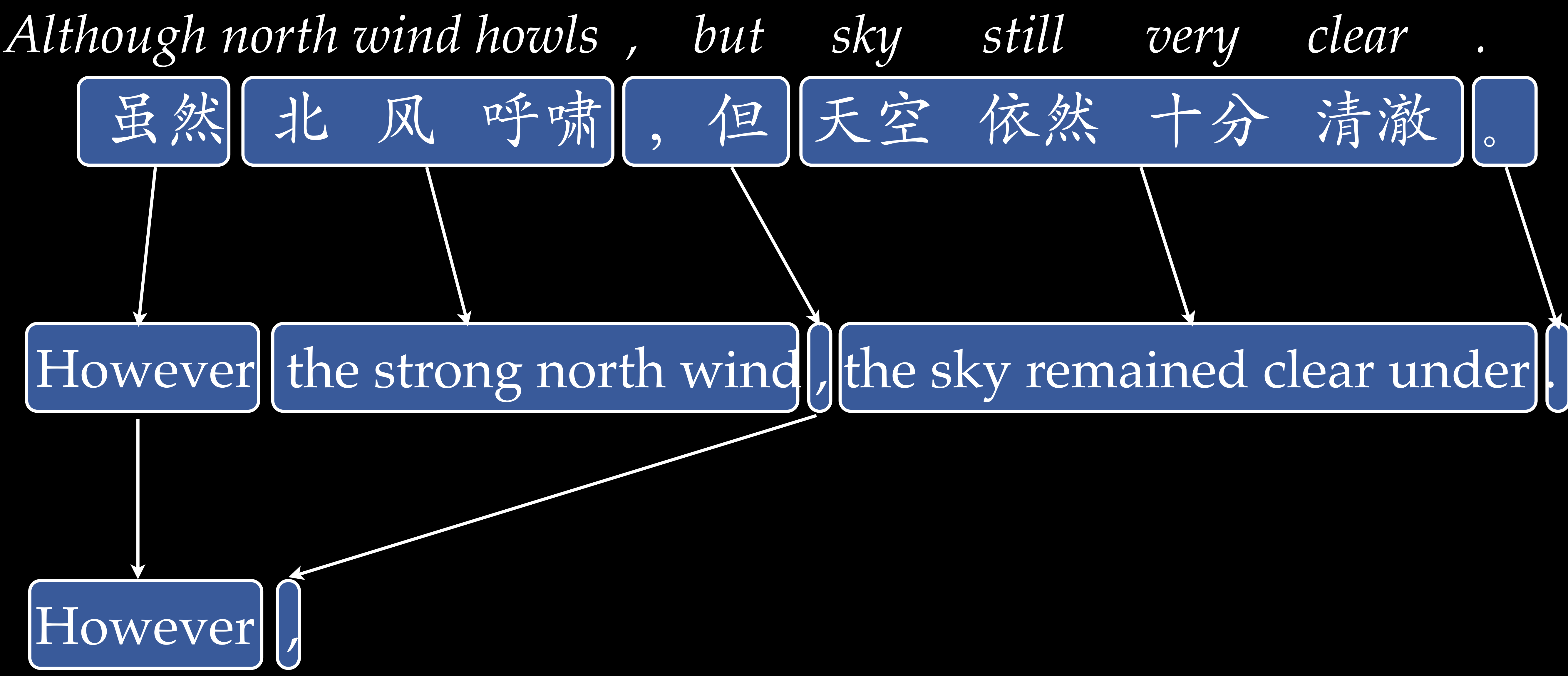
Phrase-based Models



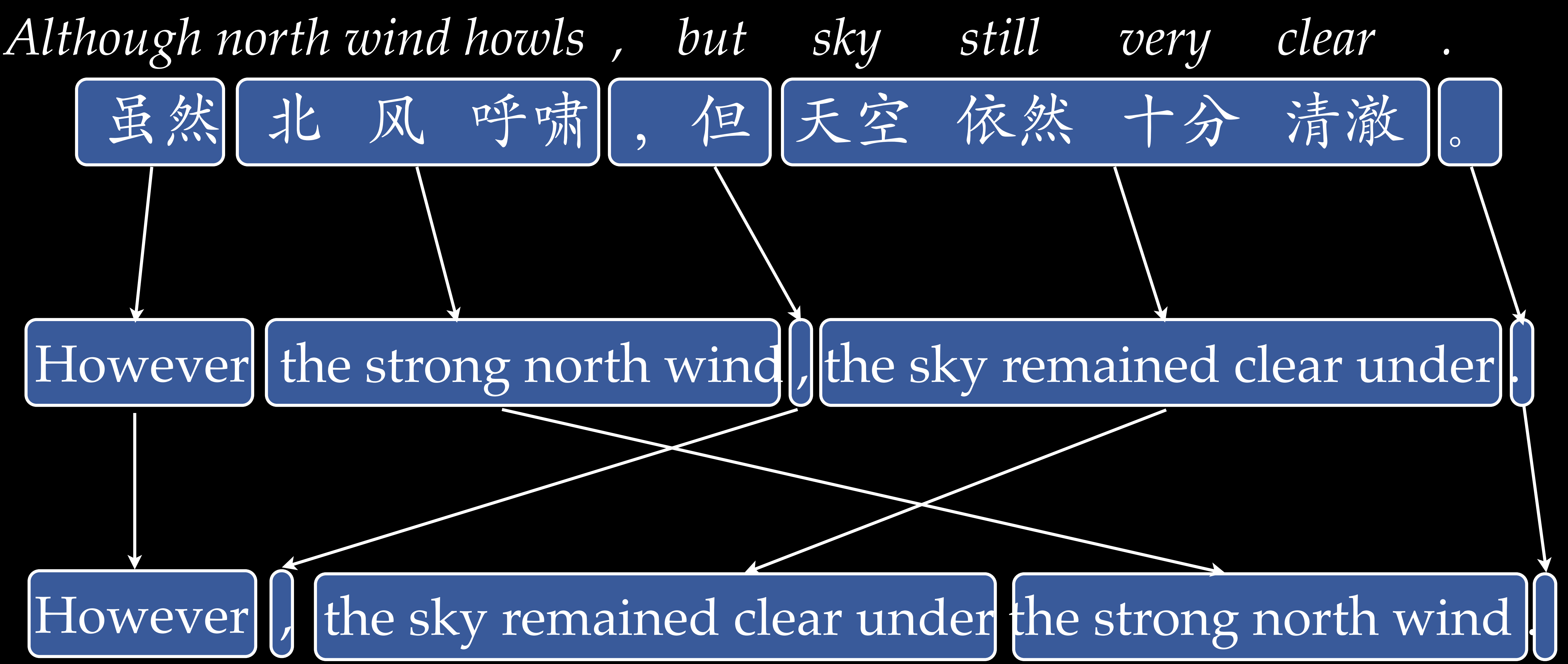
Phrase-based Models



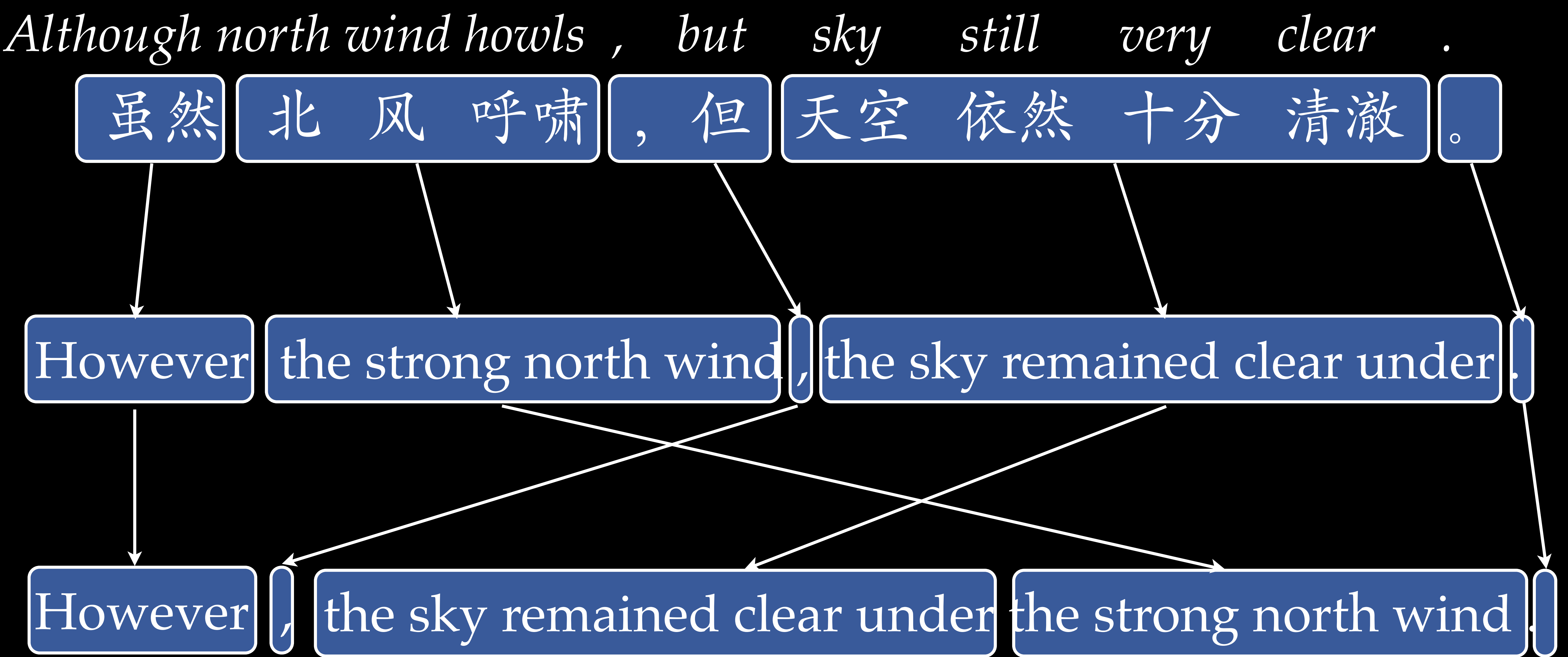
Phrase-based Models



Phrase-based Models



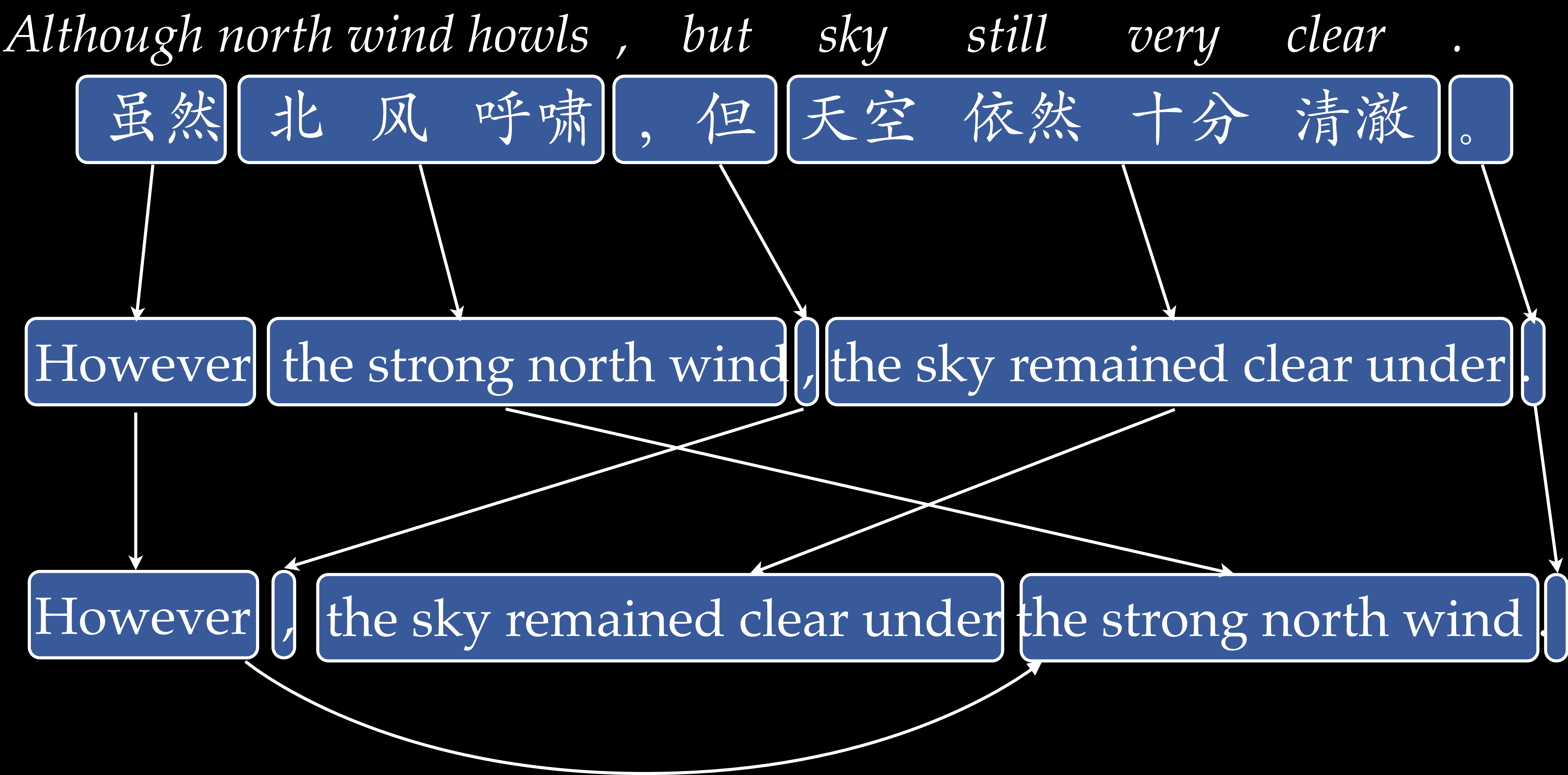
Phrase-based Models



$$p(English, alignment|Chinese) =$$

$$p(segmentation) \cdot p(translations) \cdot p(reorderings)$$

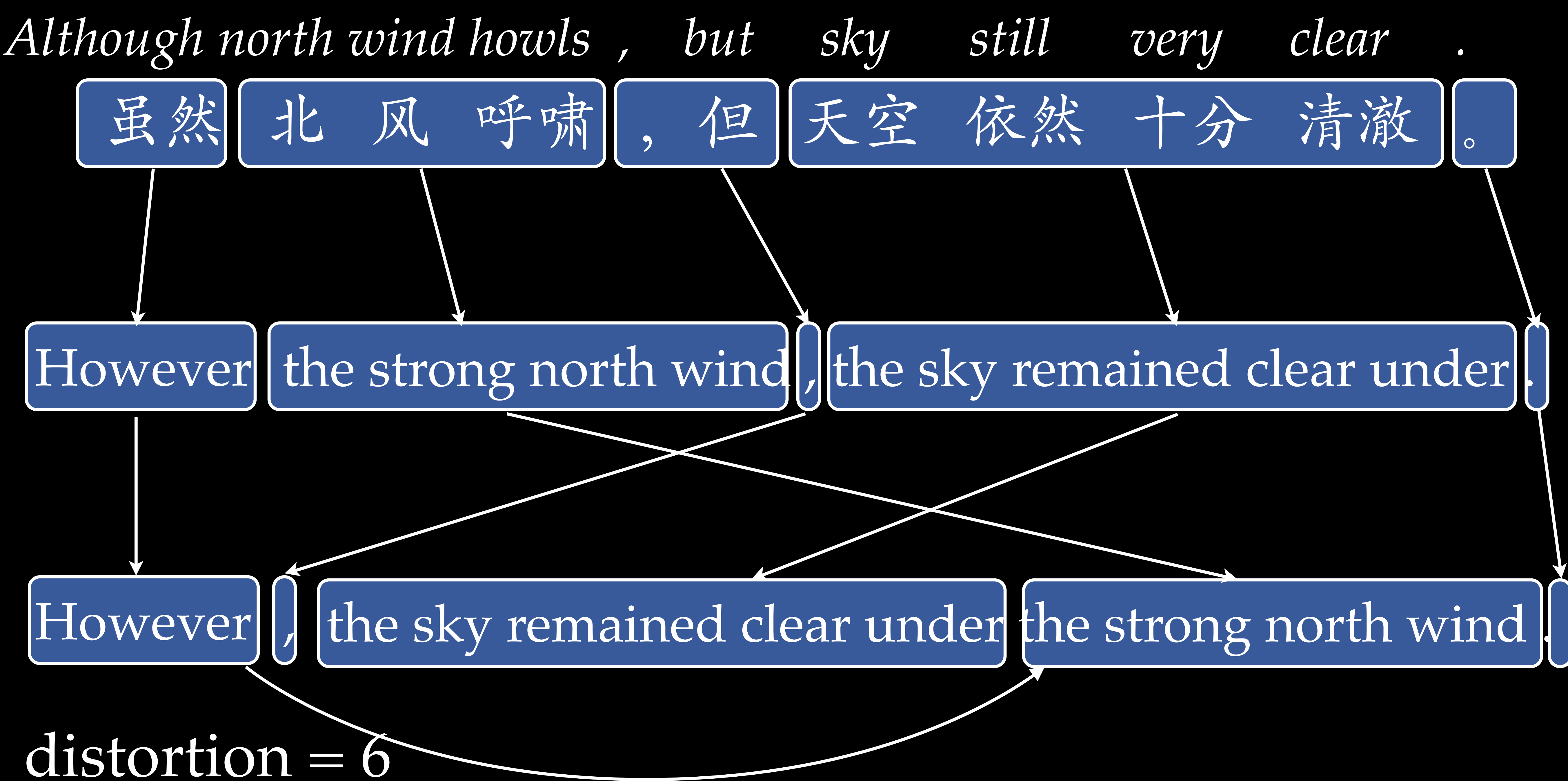
Phrase-based Models



$$p(English, alignment|Chinese) =$$

$$p(segmentation) \cdot p(translations) \cdot p(reorderings)$$

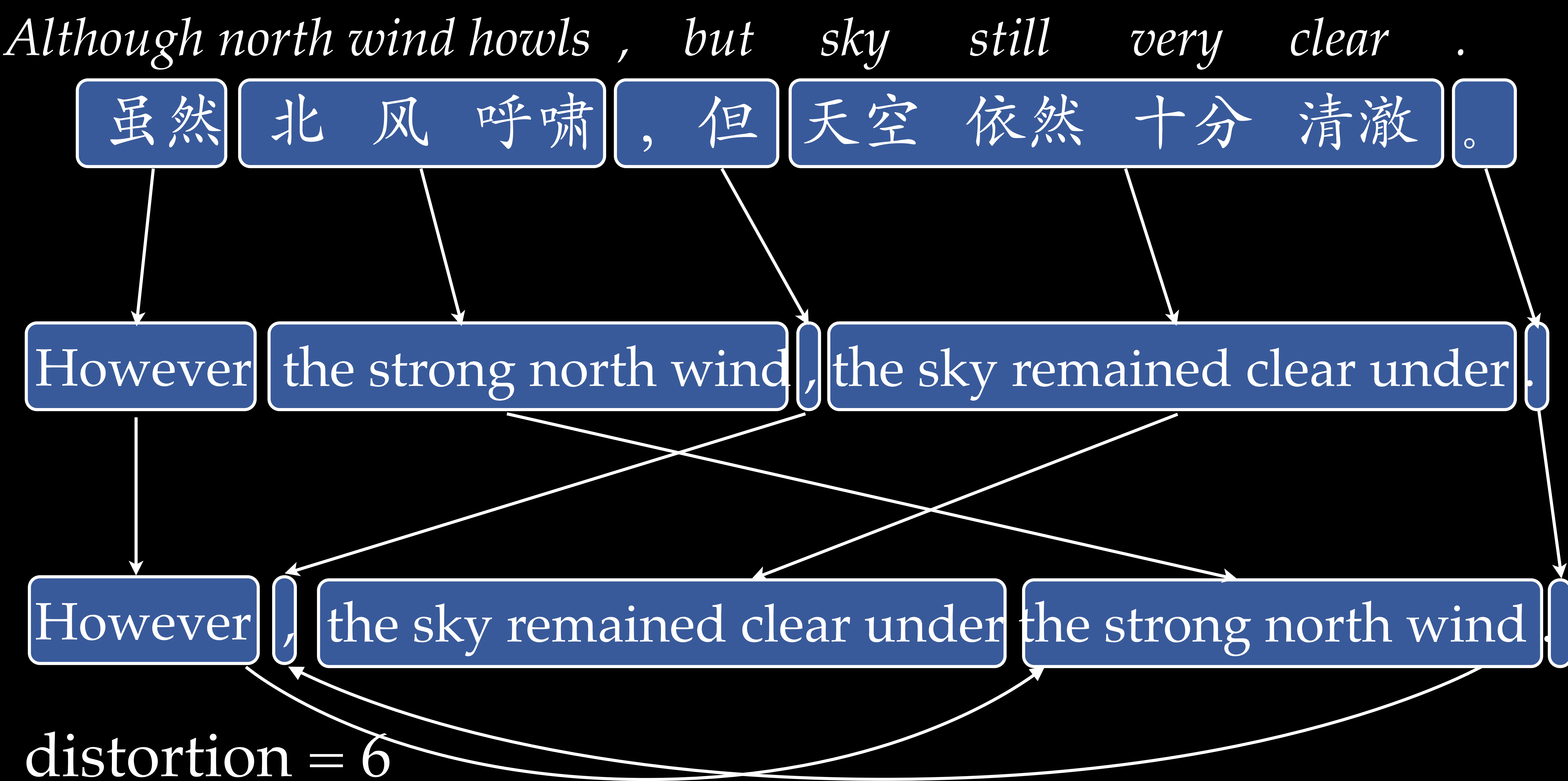
Phrase-based Models



$$p(English, alignment|Chinese) =$$

$$p(segmentation) \cdot p(translations) \cdot p(reorderings)$$

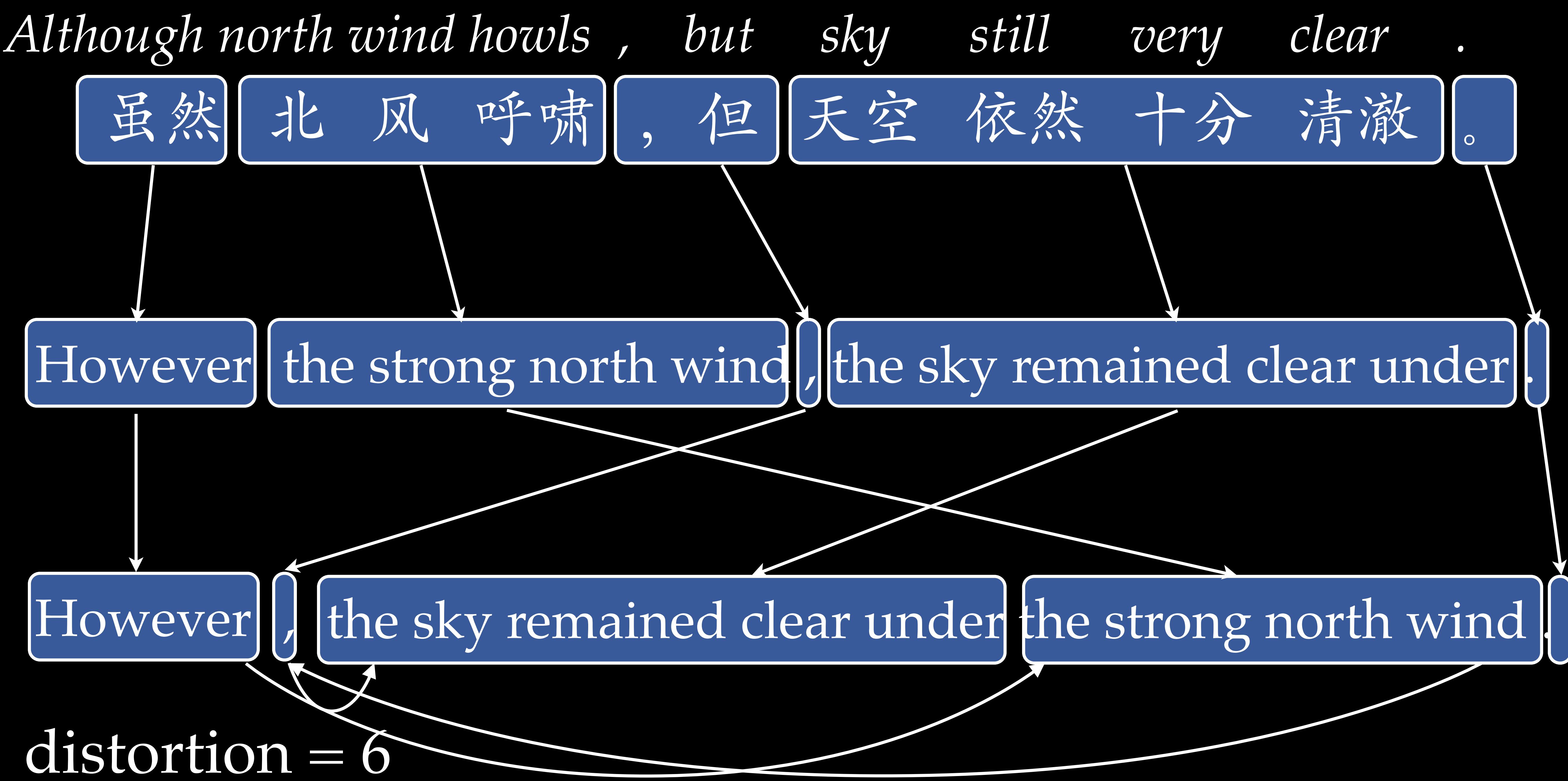
Phrase-based Models



$$p(English, alignment|Chinese) =$$

$$p(segmentation) \cdot p(translations) \cdot p(reorderings)$$

Phrase-based Models



distortion = 6

$$p(English, alignment|Chinese) =$$
$$p(segmentation) \cdot p(translations) \cdot p(reorderings)$$

Phrase-based Models

distortion = 6

$$p(English, alignment|Chinese) =$$

$$p(segmentation) \cdot p(translations) \cdot p(reorderings)$$

Phrase-based Models

Phrase-based Models

- Segmentation probabilities.

Phrase-based Models

- Segmentation probabilities.
- Phrase translation probabilities.

Phrase-based Models

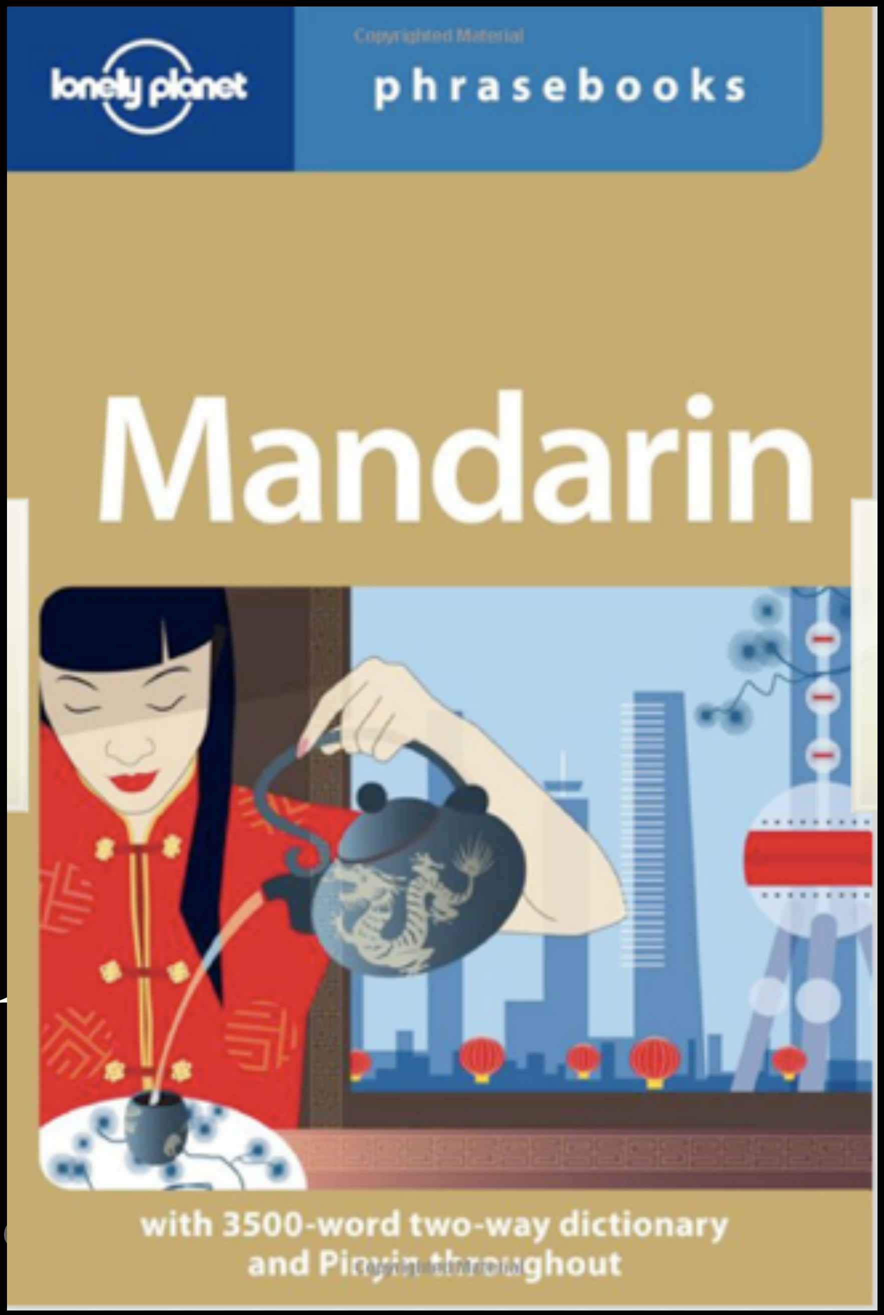
- Segmentation probabilities.
- Phrase translation probabilities.
- Distortion probabilities.

Phrase-based Models

- Segmentation probabilities.
- Phrase translation probabilities.
- Distortion probabilities.
- Some problems:
  - Weak reordering model -- output is not fluent.
  - Many decisions -- many things can go wrong.

Phrase-based Models

- Segmentation probabilities.
- Phrase translation probabilities.
- Distortion probabilities.
- Some problems:
  - Weak reordering model -- outpu
  - Many decisions -- many things c



Phrase-based Models

- Segmentation probabilities.
- Phrase translation probabilities.
- Distortion probabilities.
- Some problems:
  - Weak reordering model -- output is not fluent.
  - Many decisions -- many things can go wrong.

Phrase-based Models

- Segmentation probabilities: fixed (uniform)
- **Phrase translation probabilities.**
- Distortion probabilities: fixed (decaying)

Learning *p(Chinese | English)*

- Reminder: (nearly) every problem comes down to computing either:
  - Sums: MLE or EM (learning)
  - Maximum: most probable (decoding)

Recap: Expectation Maximization

- Arbitrarily select a set of parameters (say, uniform).
- Calculate ***expected counts*** of the unseen events.
- Choose new parameters to maximize likelihood, using expected counts as proxy for observed counts.
- Iterate.
- Guaranteed that likelihood is monotonically nondecreasing.

EM for Model 1

$$\frac{p(north|\text{北})}{\sum_{c \in Chinese\ words} p(north|c)}$$

EM for Phrase-Based

- Model parameters: *p(E phrase | F phrase)*
- All we need to do is compute expectations:

$$p(a_{i,i',j,j'} = 1 | F, E) = \frac{p(a_{i,i',j,j'} = 1, F | E)}{p(F|E)}$$

EM for Phrase-Based

- Model parameters: *p(E phrase | F phrase)*
- All we need to do is compute expectations:

$$p(a_{i,i',j,j'} = 1|F, E) = \frac{p(a_{i,i',j,j'} = 1, F|E)}{p(F|E)}$$

p(F,E) sums over all possible phrase alignments

EM for Phrase-Based

- Model parameters: *p(E phrase | F phrase)*
- All we need to do is compute expectations:

$$p(a_{i,i',j,j'} = 1 | F, E) = \frac{p(a_{i,i',j,j'} = 1, F|E)}{p(F|E)}$$

p(F,E) sums over all possible phrase alignments

...which are one-to-one by definition.

EM for Phrase-Based

Although north wind howls , but sky still very clear .

虽然 北 风 呼啸 ， 但 天空 依然 十分 清澈 。

However , the sky remained clear under the strong north wind .

$$p(a_{i,i',j,j'} = 1|F, E) = \frac{p(a_{i,i',j,j'} = 1, F|E)}{p(F|E)}$$

EM for Phrase-Based

Although north wind howls , but sky still very clear .

虽然 $\boxed{\text{北 风 呼啸}}$ ，但 天空 依然 十分 清澈 。

However , the sky remained clear under $\boxed{\text{the strong north wind}}$.

$$p(a_{i,i',j,j'} = 1|F, E) = \frac{p(a_{i,i',j,j'} = 1, F|E)}{p(F|E)}$$

Can we compute this quantity?

EM for Phrase-Based

Although north wind howls , but sky still very clear .

虽然 $\boxed{\text{北 风 呼啸}}$ ，但 天空 依然 十分 清澈 。

However , the sky remained clear under $\boxed{\text{the strong north wind}}$.

$$p(a_{i,i',j,j'} = 1|F, E) = \frac{p(a_{i,i',j,j'} = 1, F|E)}{p(F|E)}$$

Can we compute this quantity?

How many 1-to-1 alignments are there of the remaing 8 Chinese and 8 English words?

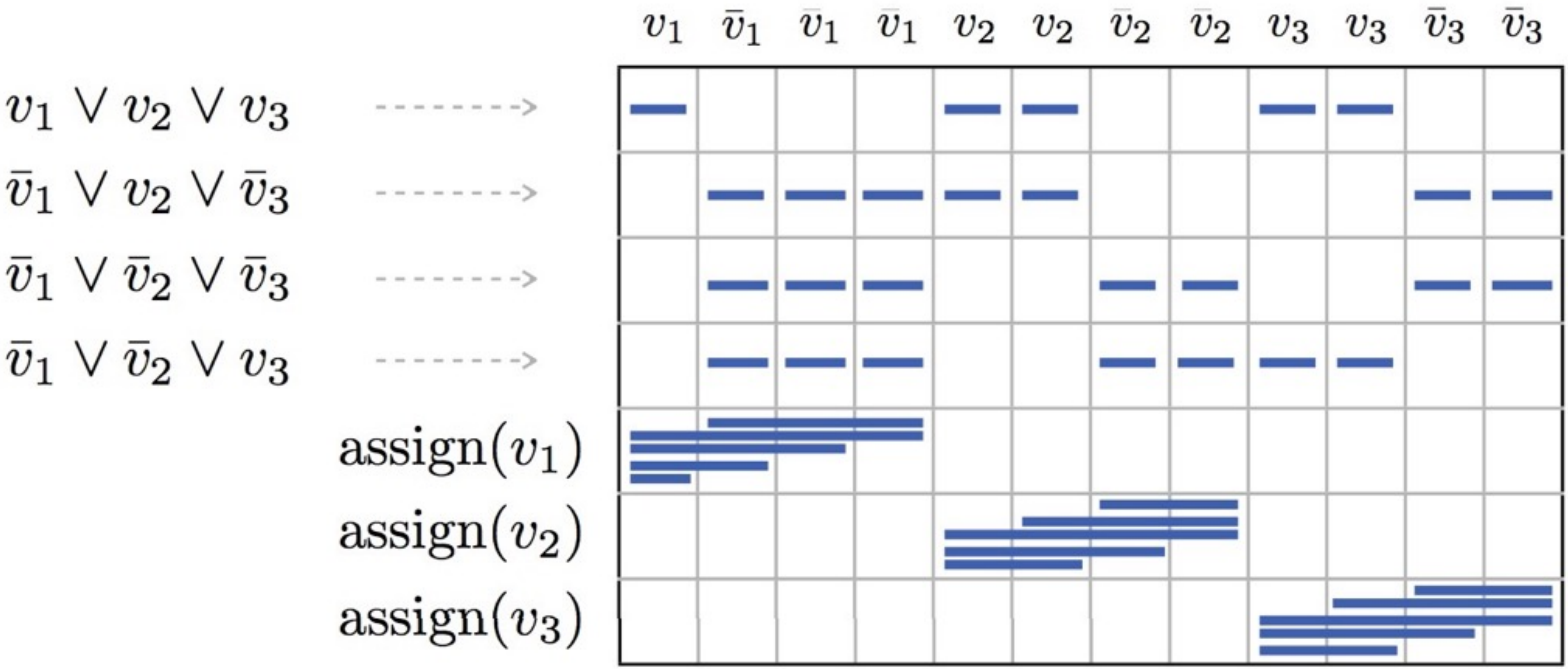
Recap: Expectation Maximization

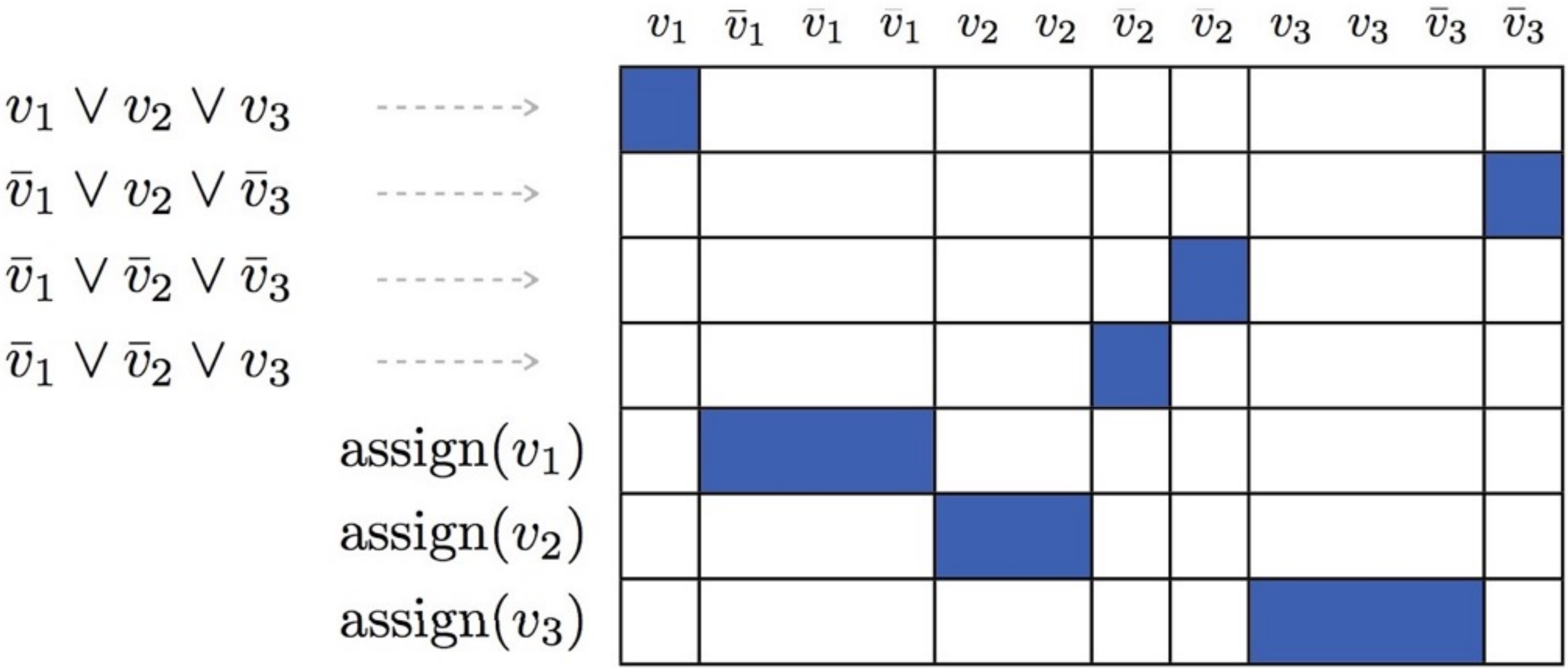
- Arbitrarily select a set of parameters (say, uniform).
- Calculate ***expected counts*** of the unseen events.
- Choose new parameters to maximize likelihood, using expected counts as proxy for observed counts.
- Iterate.
- Guaranteed that likelihood is monotonically nondecreasing.

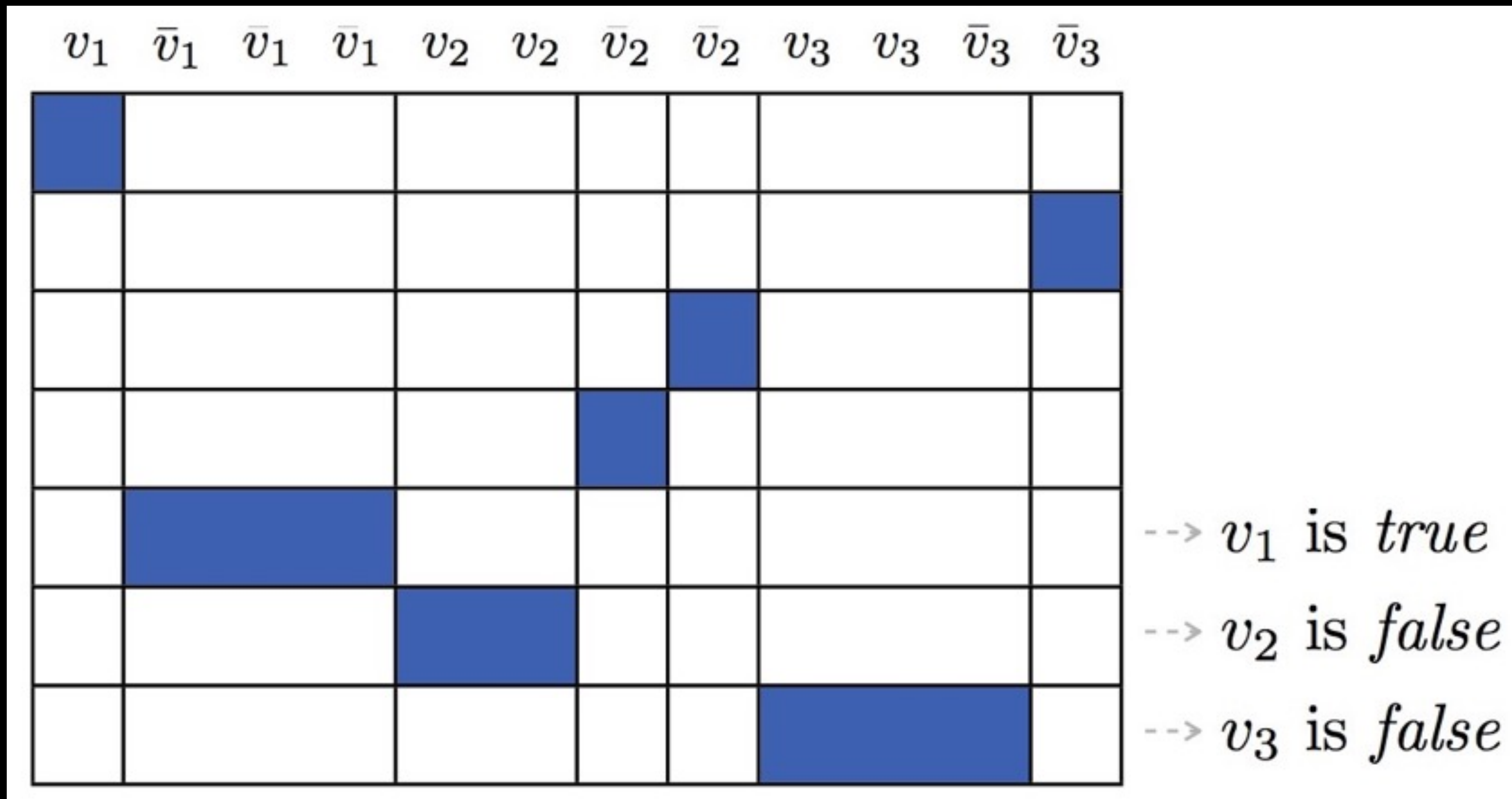
Recap: Expectation Maximization

- Arbitrarily select a set of parameters (say, uniform).
- Calculate ***expected counts*** of the unseen events.
- Choose new parameters to maximize likelihood, us nts.
- It
- G n

Computing expectations from a phrase-based model, given a sentence pair, is #P-Complete (by reduction to counting perfect matchings; DeNero & Klein, 2008)


v_1 $\bar{v}_1$ $\bar{v}_1$ $\bar{v}_1$ v_2 v_2 $\bar{v}_2$ $\bar{v}_2$ v_3 v_3 $\bar{v}_3$ $\bar{v}_3$
$v_1 \vee v_2 \vee v_3$
$\bar{v}_1 \vee v_2 \vee \bar{v}_3$
$\bar{v}_1 \vee \bar{v}_2 \vee \bar{v}_3$
$\bar{v}_1 \vee \bar{v}_2 \vee v_3$
$\mathrm{assign}(v_1)$
$\mathrm{assign}(v_2)$
$\mathrm{assign}(v_3)$


v_1 $\bar{v}_1$ $\bar{v}_1$ $\bar{v}_1$ v_2 v_2 $\bar{v}_2$ $\bar{v}_2$ v_3 v_3 $\bar{v}_3$ $\bar{v}_3$
$v_1 \vee v_2 \vee v_3$
$\bar{v}_1 \vee v_2 \vee \bar{v}_3$
$\bar{v}_1 \vee \bar{v}_2 \vee \bar{v}_3$
$\bar{v}_1 \vee \bar{v}_2 \vee v_3$
$\mathrm{assign}(v_1)$
$\mathrm{assign}(v_2)$
$\mathrm{assign}(v_3)$



v_1 $\bar{v}_1$ $\bar{v}_1$ $\bar{v}_1$ v_2 v_2 $\bar{v}_2$ $\bar{v}_2$ v_3 v_3 $\bar{v}_3$ $\bar{v}_3$
v_1 is *true*
v_2 is *false*
v_3 is *false*

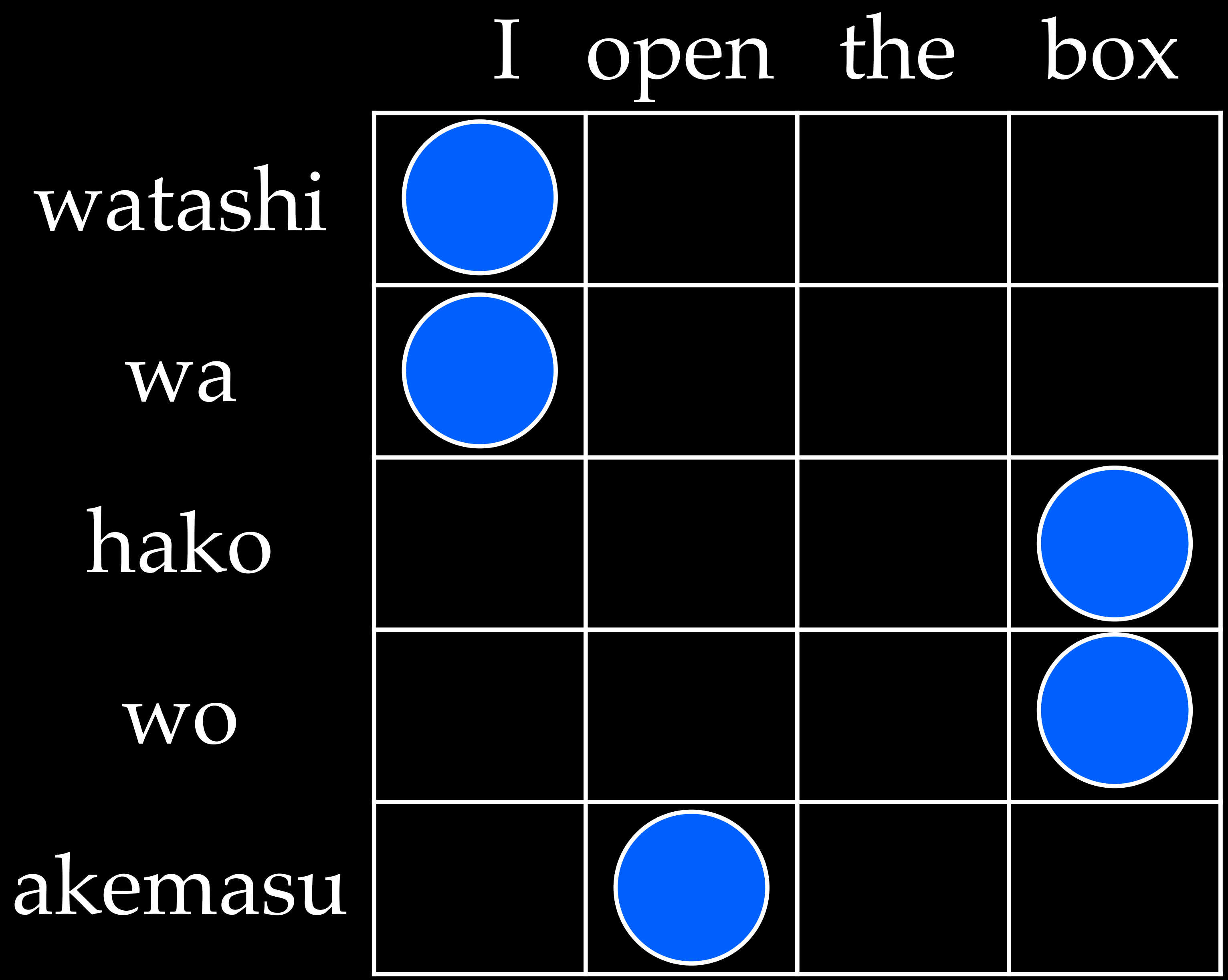
Now What?

- Option #1: approximate expectations
  - Restrict computation to some tractable subset of the alignment space (arbitrarily biased).
  - Markov chain Monte Carlo (slow).

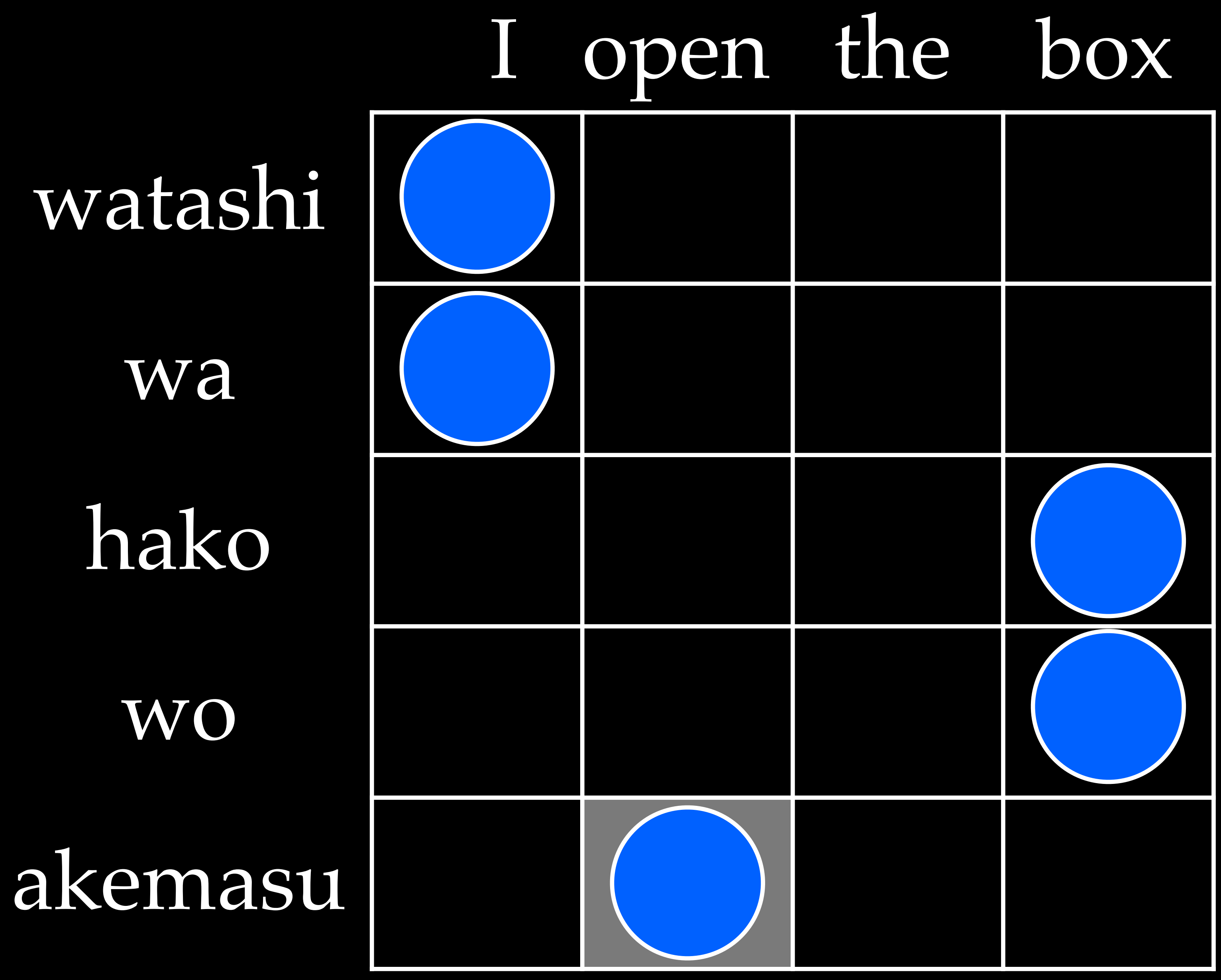
Now What?

- Change the problem definition
  - We already know how to learn word-to-word translation models efficiently.
  - Idea: learn word-to-word alignments, extract most probable alignment, then treat it as observed.
  - Learn phrase translations consistent with word alignments.
  - Decouples alignment from model learning -- is this a good thing?

Phrase Extraction

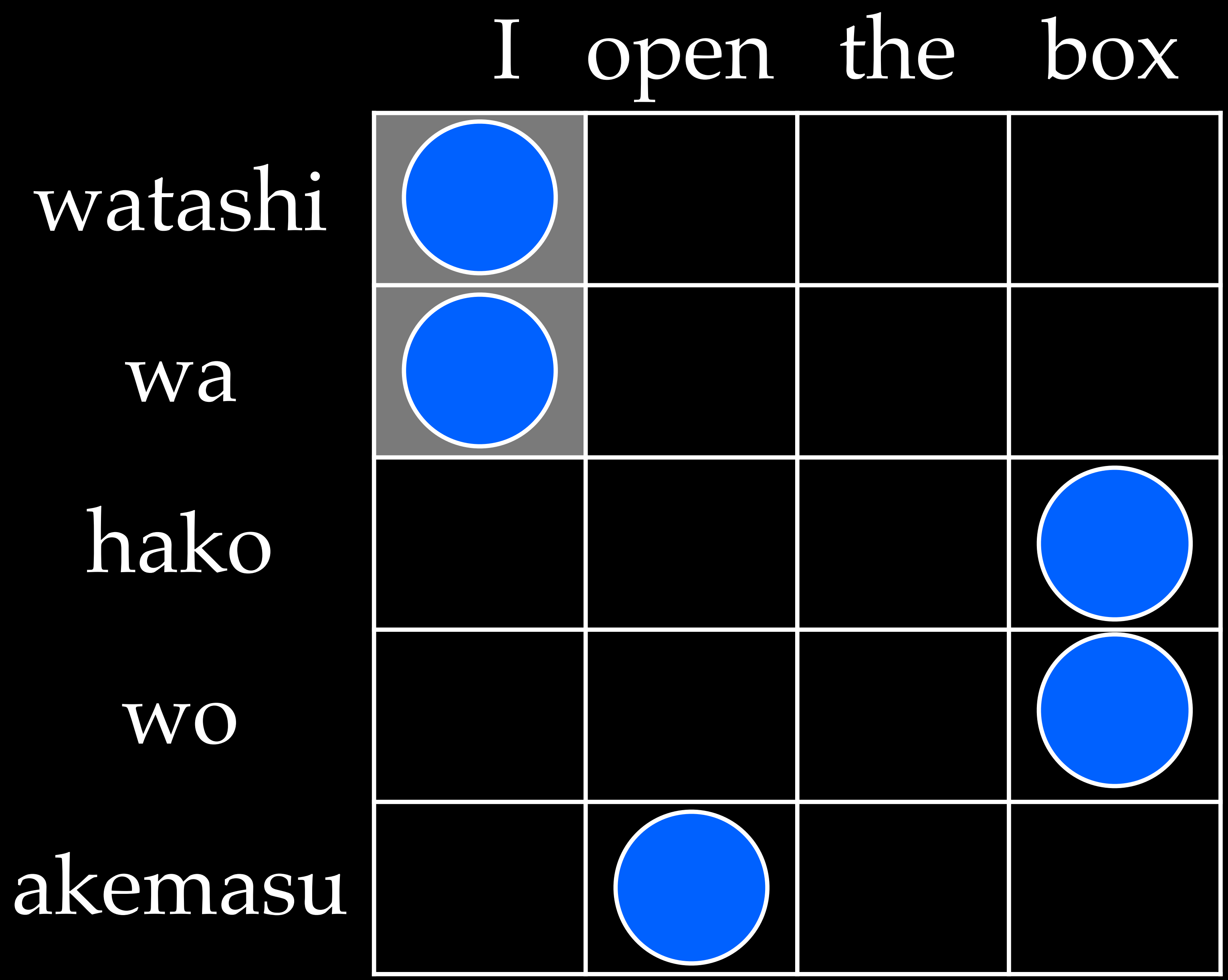


Phrase Extraction



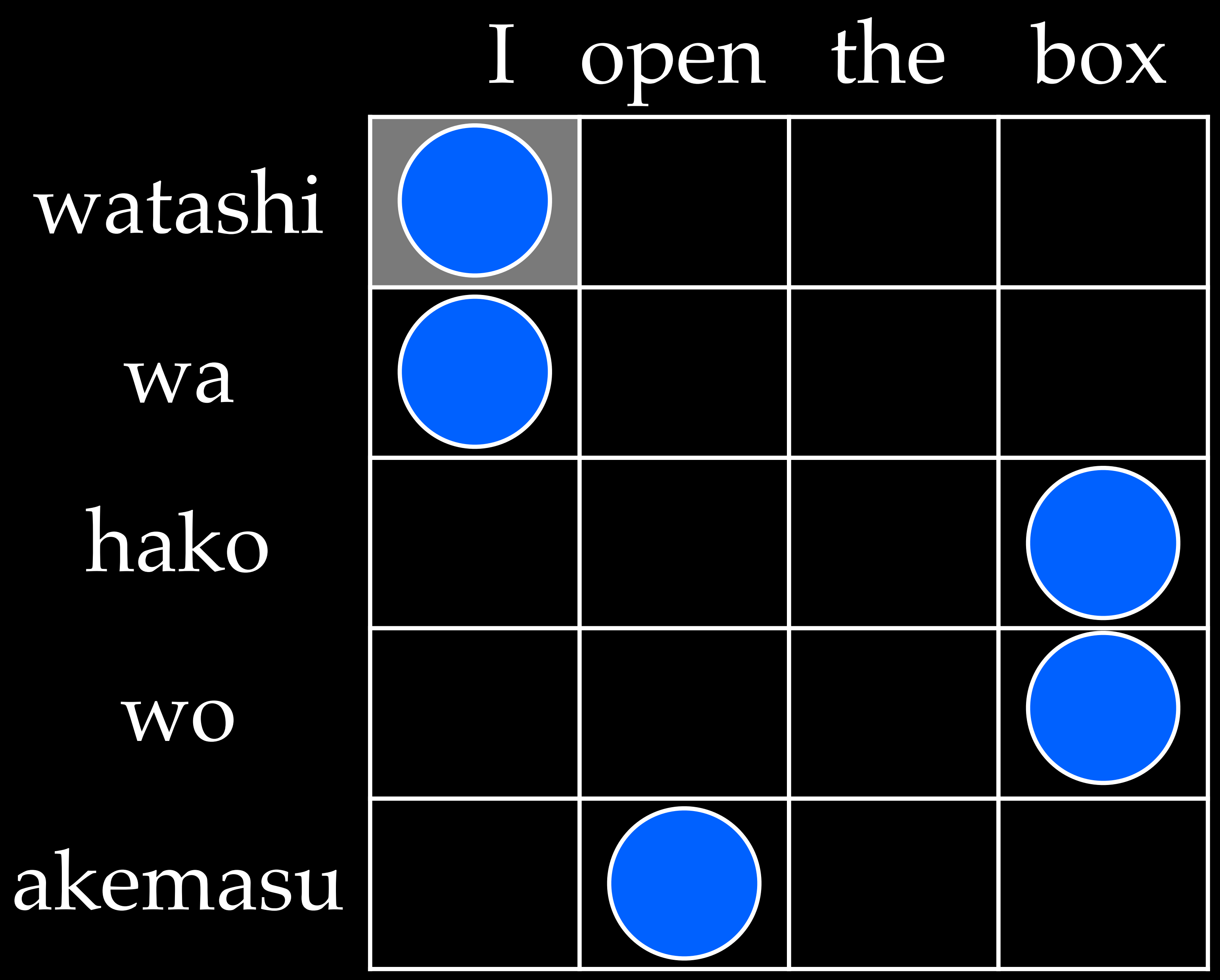
akemasu / open

Phrase Extraction



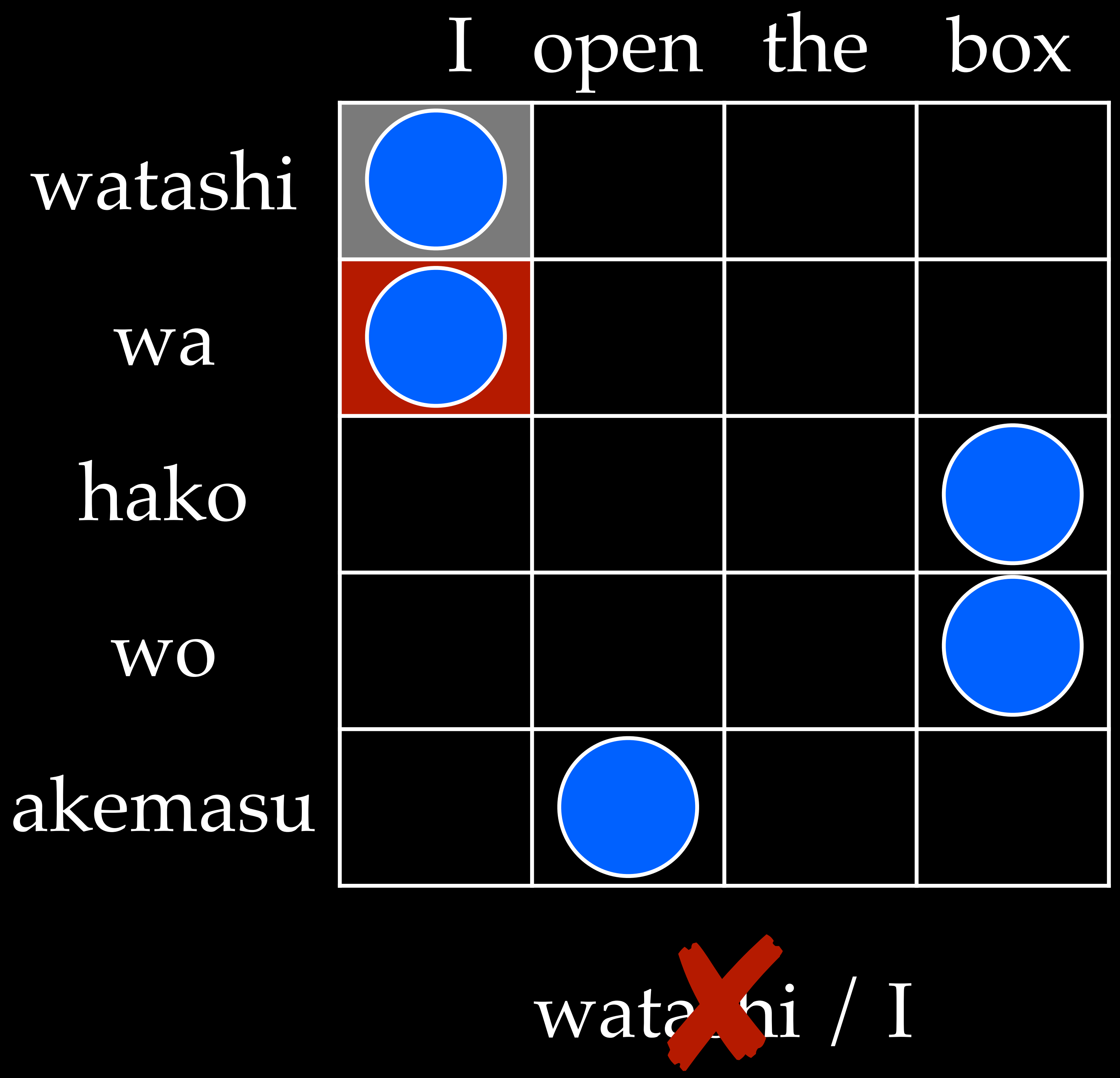
watashi wa / I

Phrase Extraction

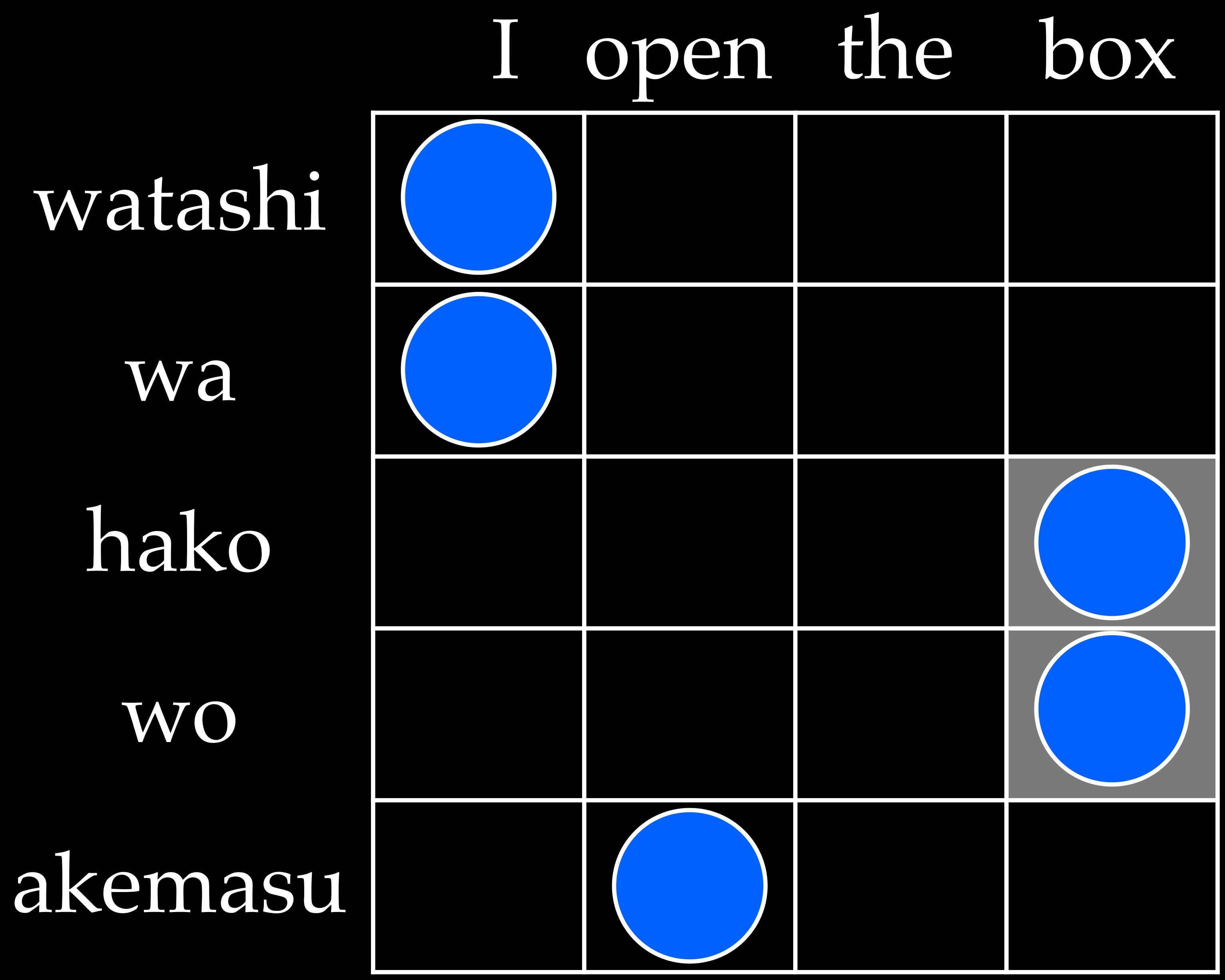


watashi / I

Phrase Extraction

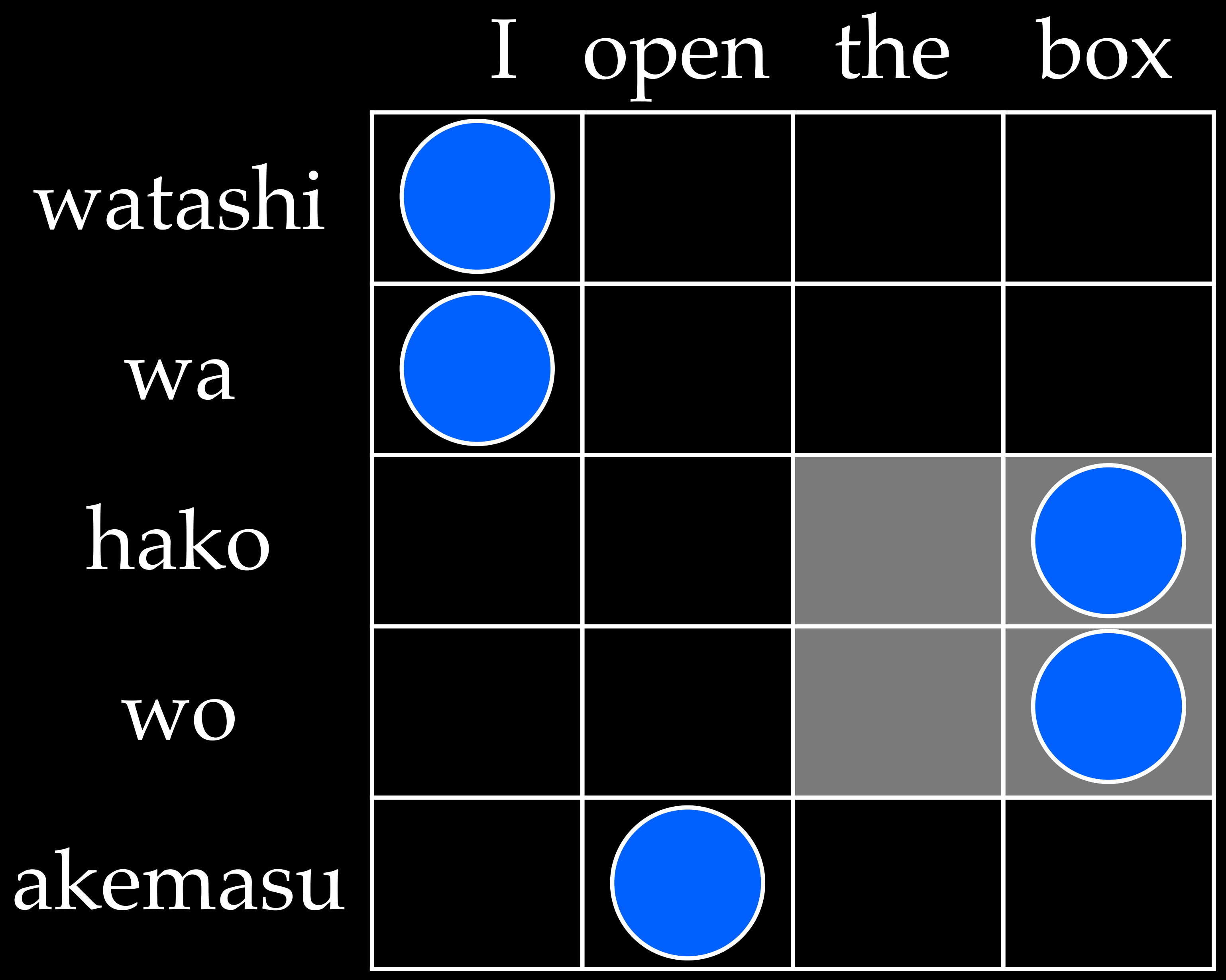


Phrase Extraction



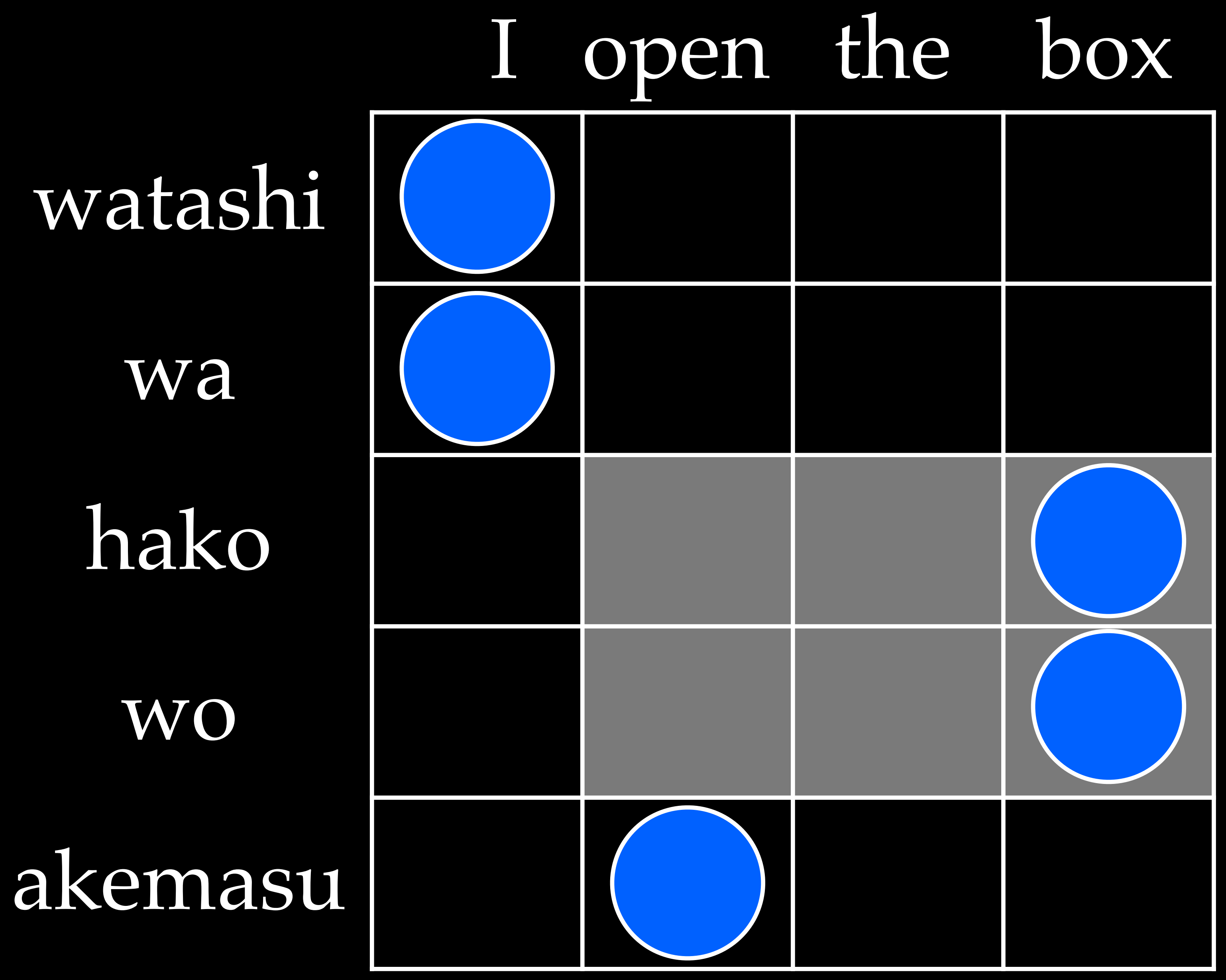
hako wo / box

Phrase Extraction



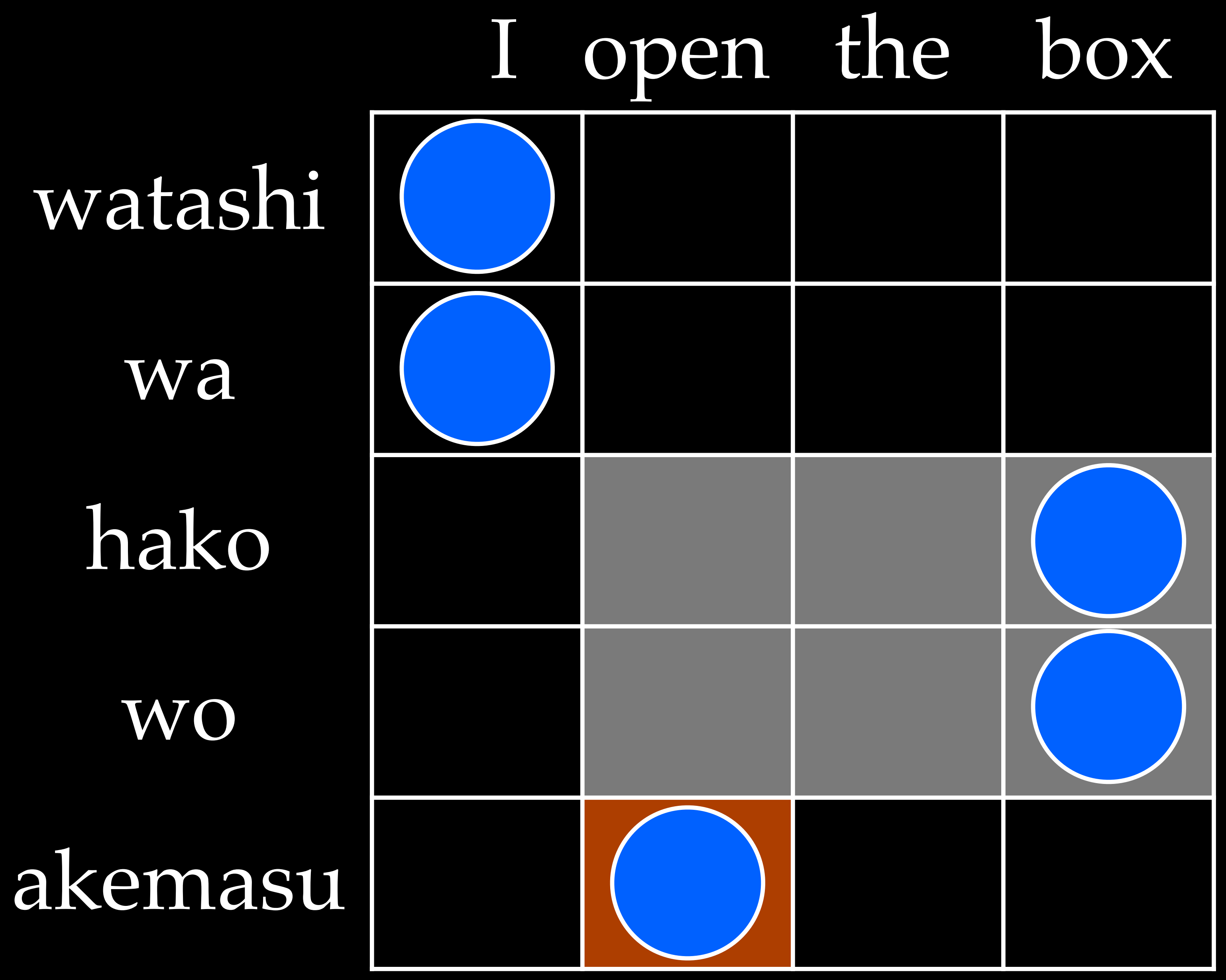
hako wo / the box

Phrase Extraction



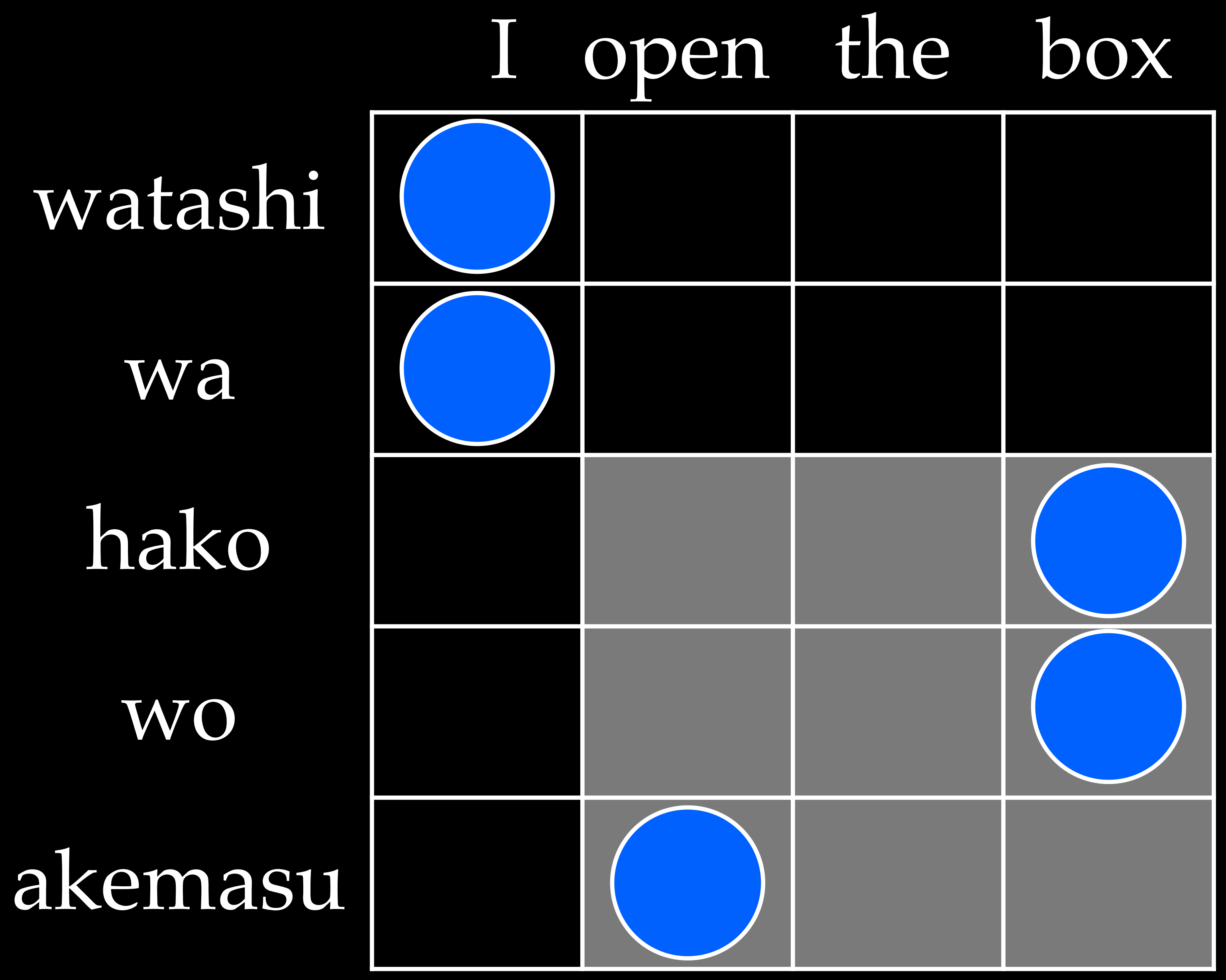
hako wo / open the box

Phrase Extraction



hako wo / ~~o~~pen the box

Phrase Extraction



hako wo akemasu / open the box

Phrasal Translation Estimation

Phrasal Translation Estimation

- Option #1 (EM over restricted space)
  - Align with a word-based model.
  - Compute expectations only over alignments consistent with the alignment grid.

Phrasal Translation Estimation

- Option #1 (EM over restricted space)
  - Align with a word-based model.
  - Compute expectations only over alignments consistent with the alignment grid.
- Option #2 (Non-global estimation)
  - View phrase pairs as observed, irrespective of context or overlap.

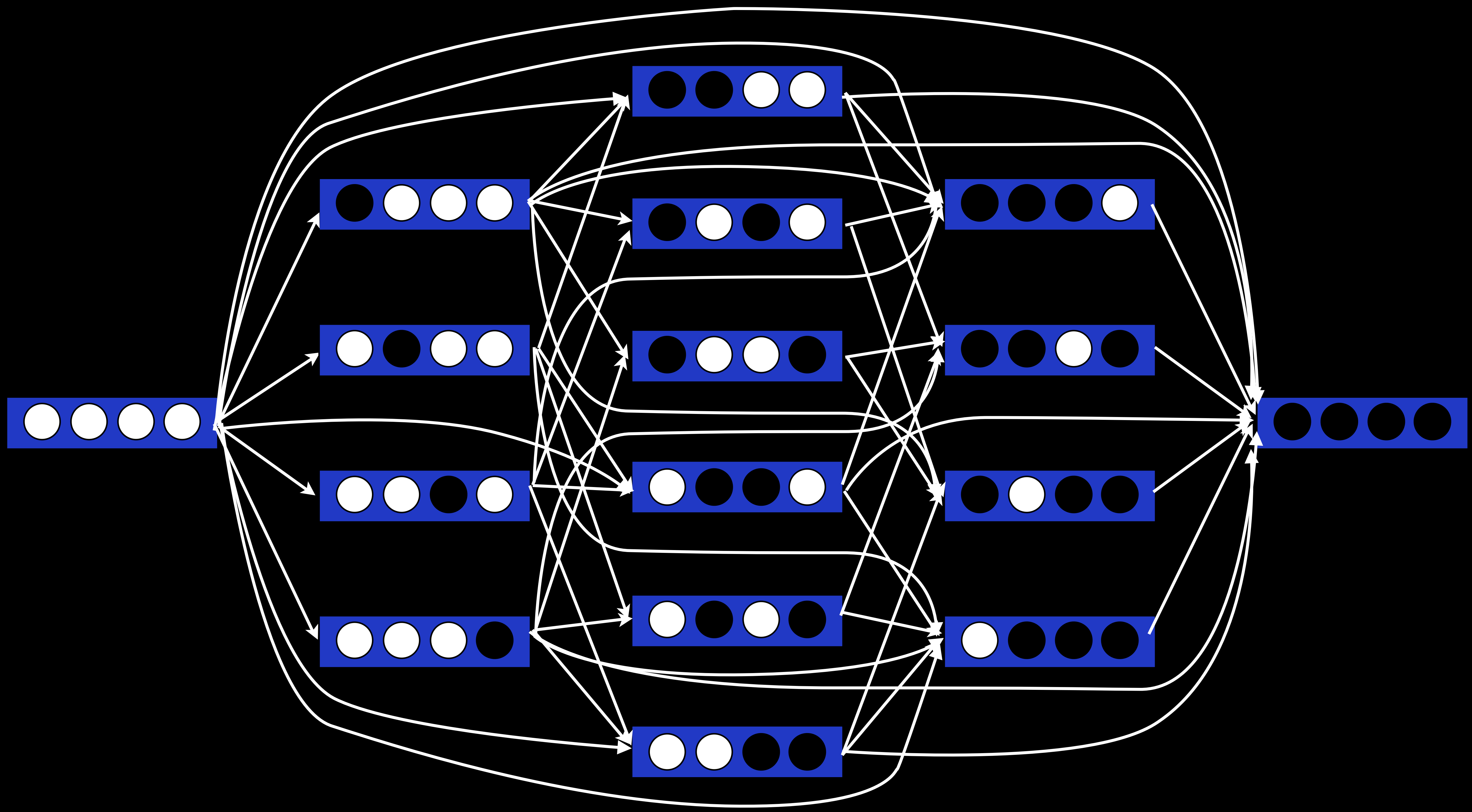
Decoding

We want to solve this problem:

$$e^* = \arg\max_e p(\mathbf{e}|\mathbf{f})$$

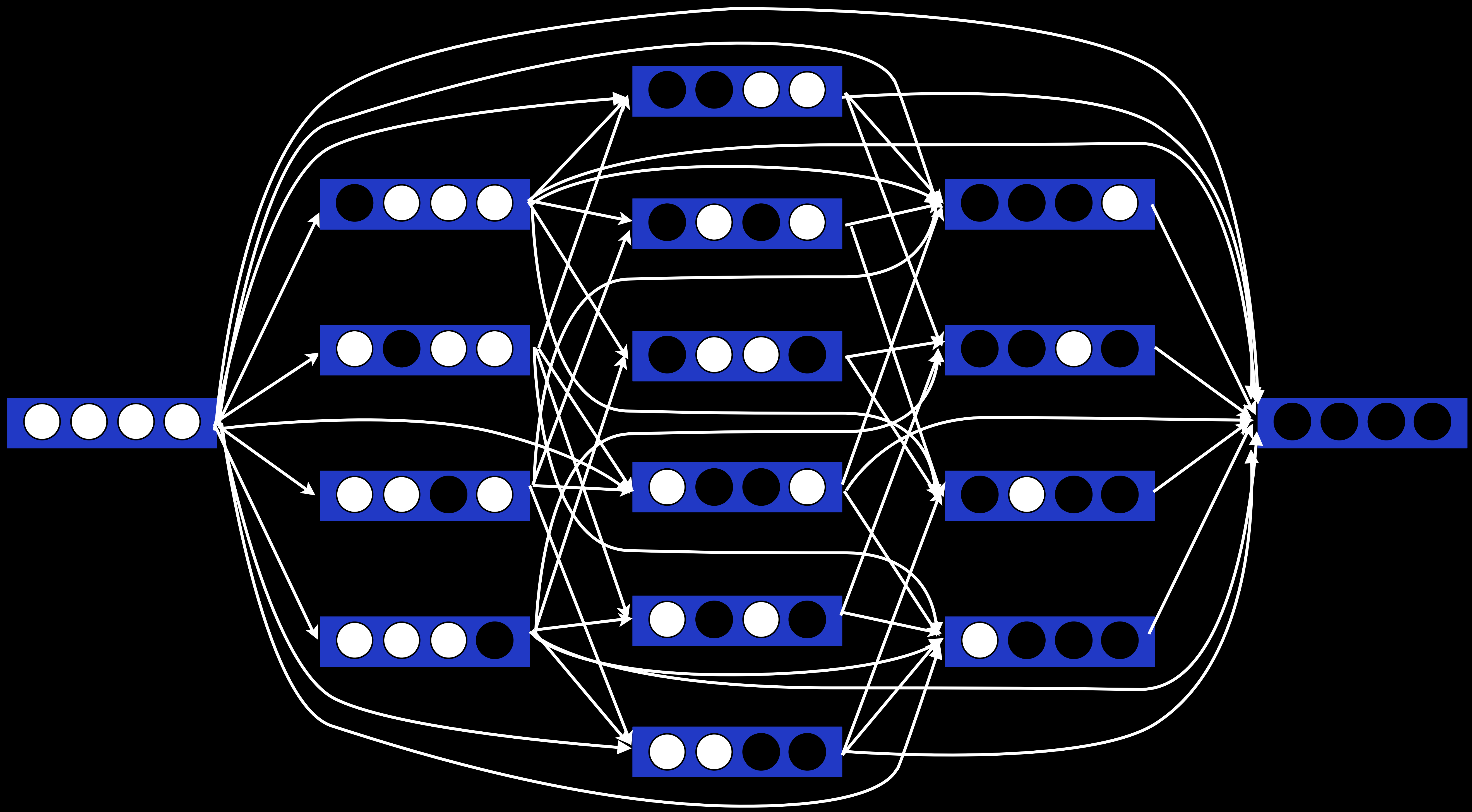
Does this problem look familiar?

Decoding similar to IBM4



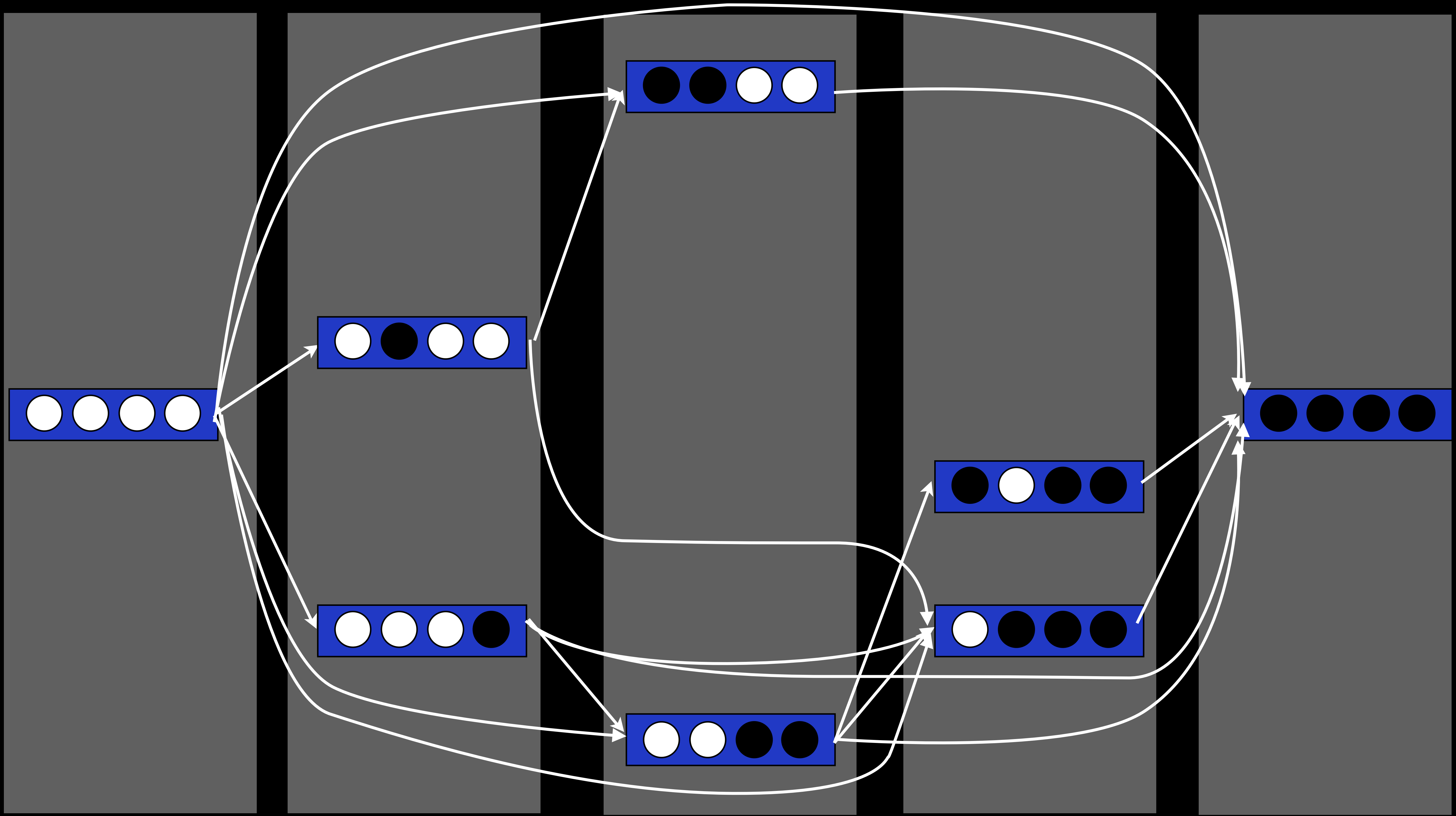
Dynamic Programming

Decoding similar to IBM4

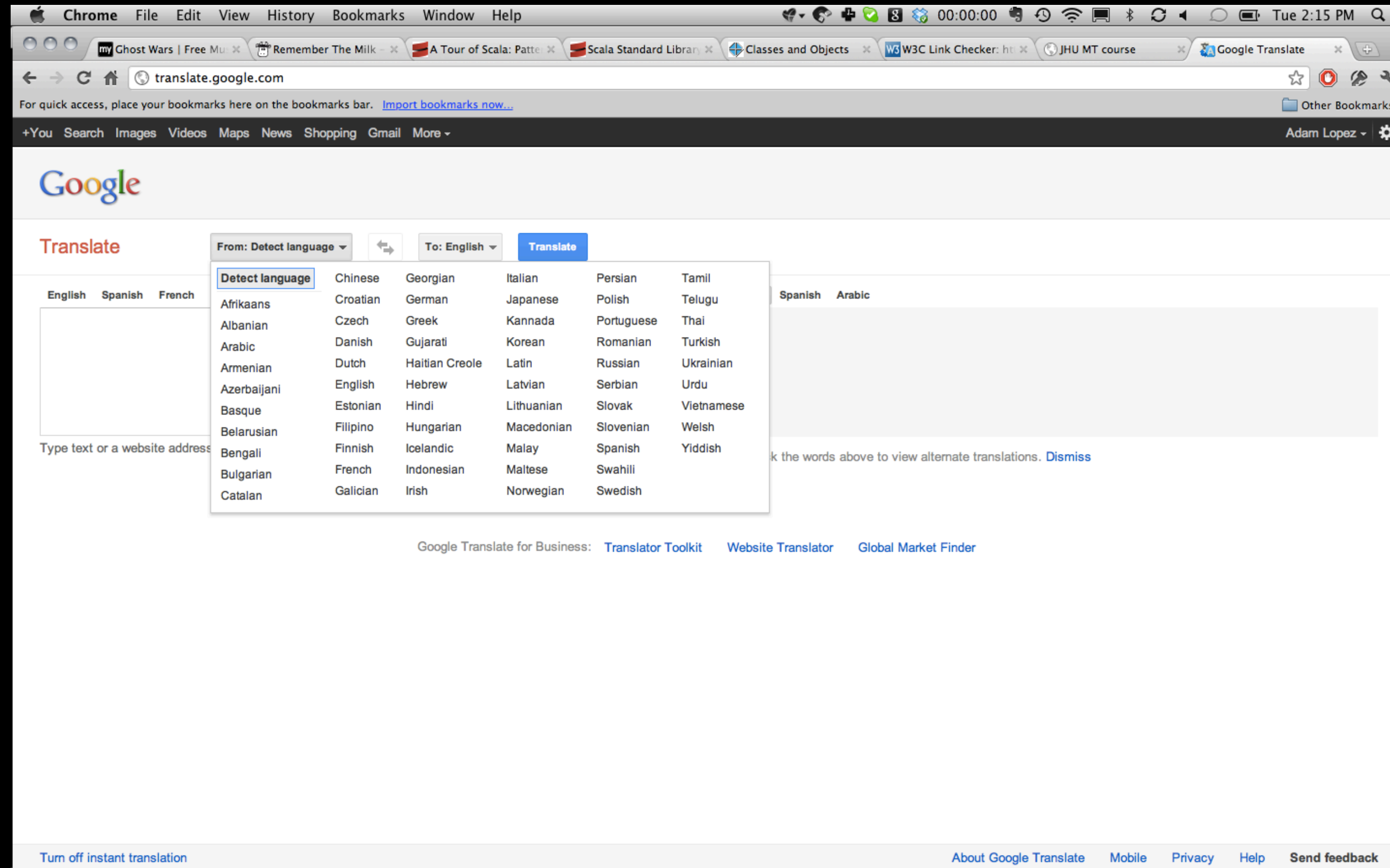


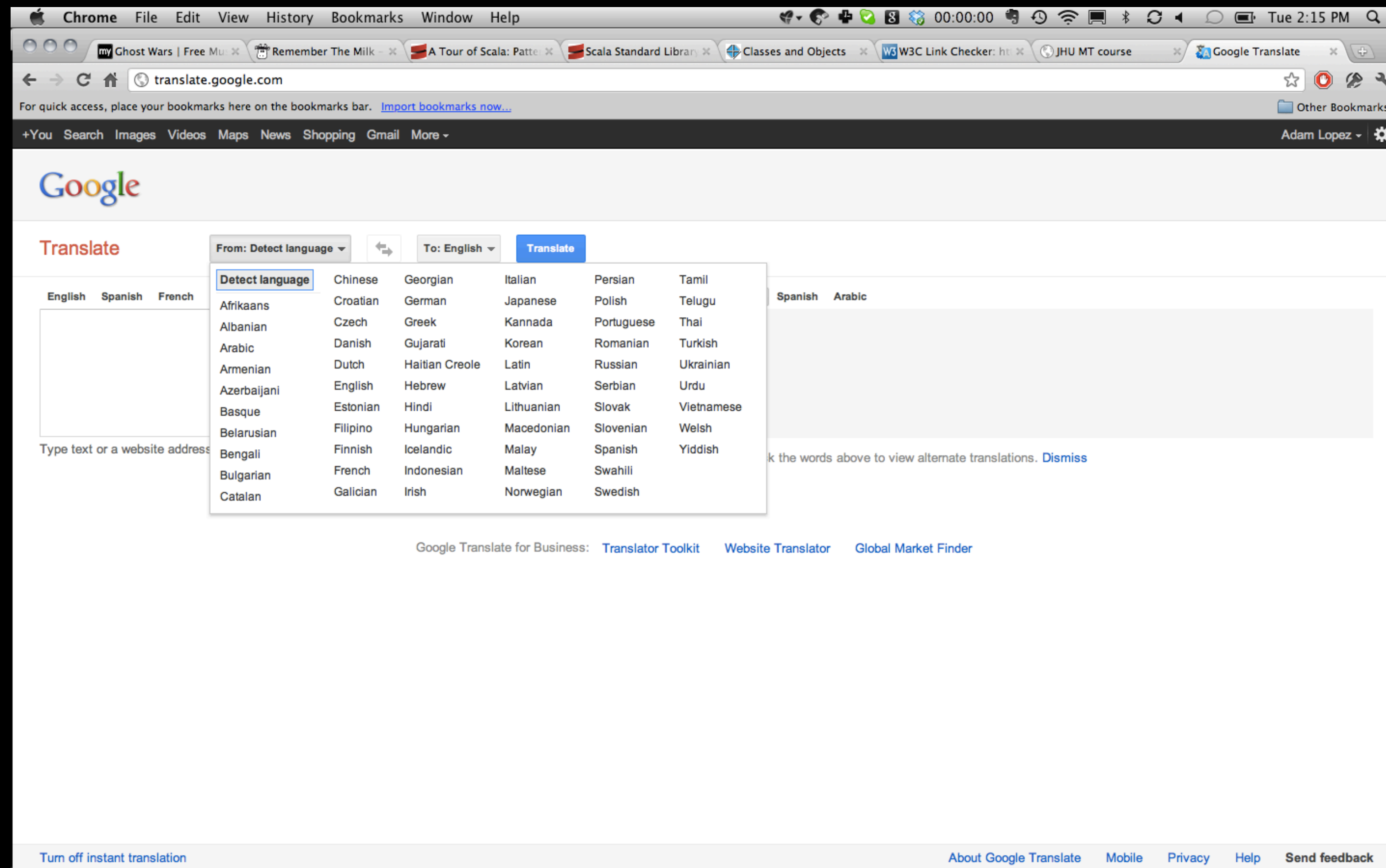
Dynamic Programming

Approximation: Pruning

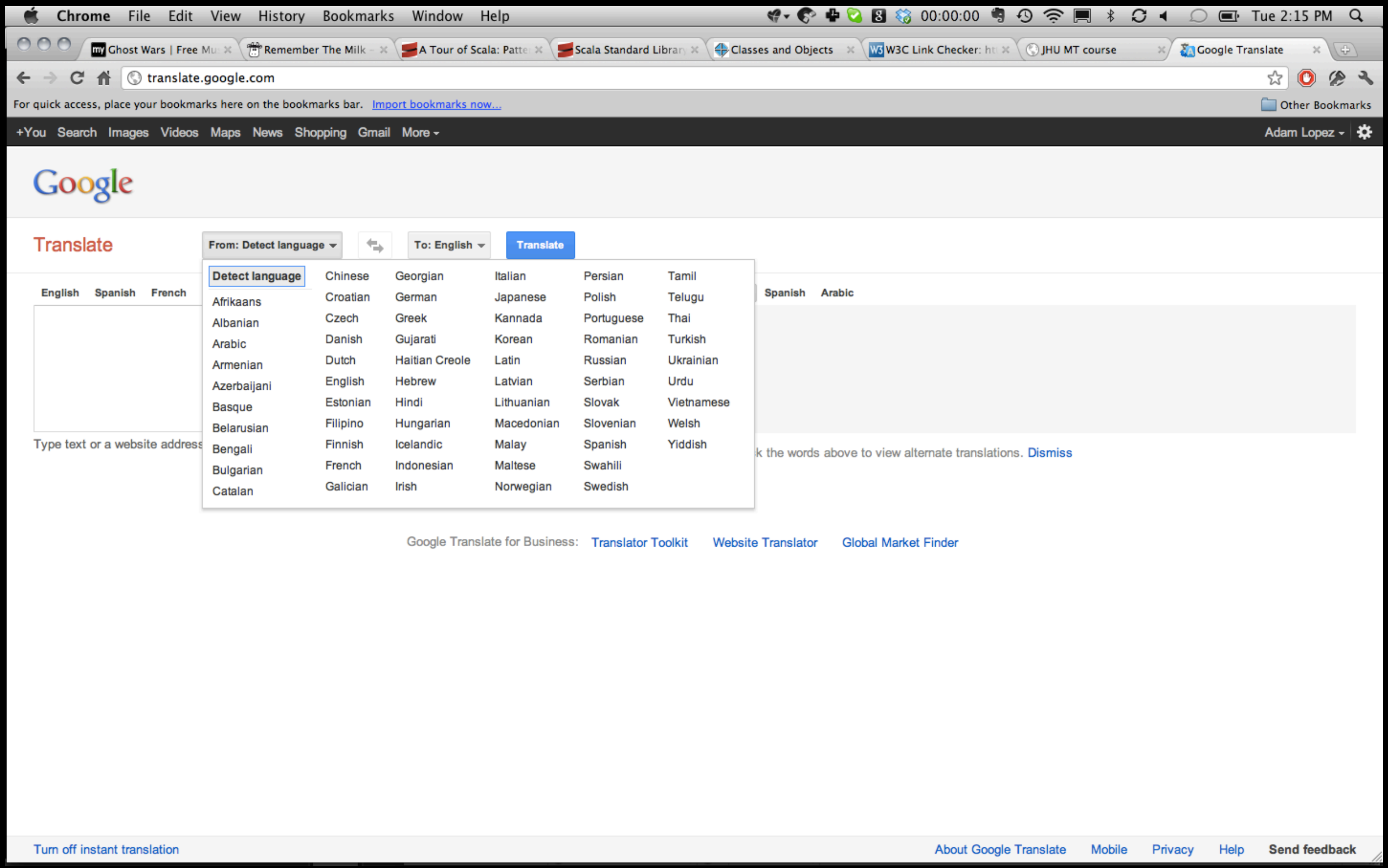


Similar approximations applicable

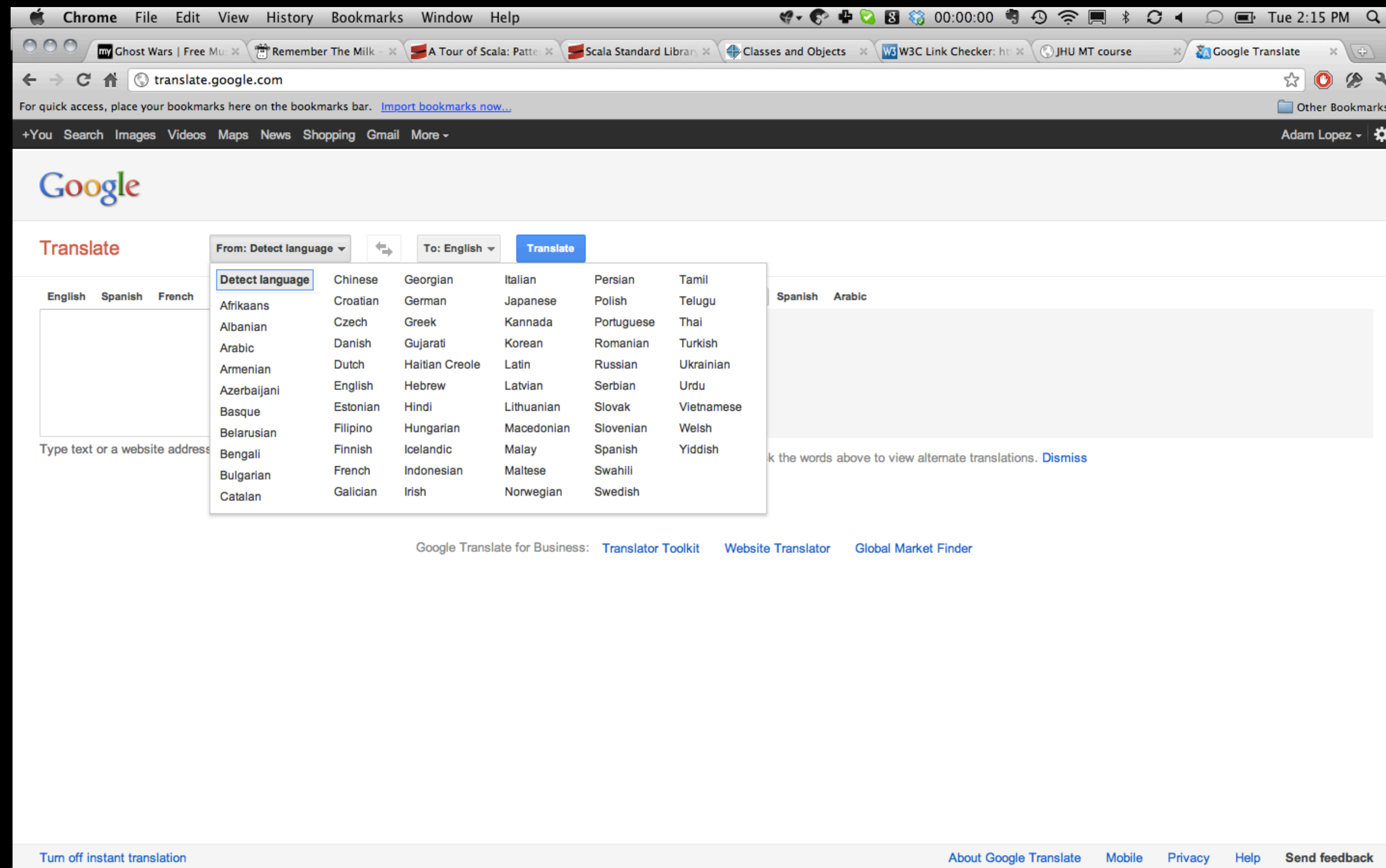

Chrome File Edit View History Bookmarks Window Help
Tue 2:15 PM
Ghost Wars | Free Mu
Remember The Milk
A Tour of Scala: Patte
Scala Standard Librar
Classes and Objects
W3C Link Checker: ht
JHU MT course
Google Translate
translate.google.com
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Other Bookmarks
+You Search Images Videos Maps News Shopping Gmail More
Adam Lopez
Google
Translate
From: Detect language
To: English
Translate
English Spanish French
Spanish Arabic
Detect language
Afrikaans
Albanian
Arabic
Armenian
Azerbaijani
Basque
Belarusian
Bengali
Bulgarian
Catalan
Chinese
Croatian
Czech
Danish
Dutch
English
Estonian
Filipino
Finnish
French
Galician
Georgian
German
Greek
Gujarati
Haitian Creole
Hebrew
Hindi
Hungarian
Icelandic
Indonesian
Irish
Italian
Japanese
Kannada
Korean
Latin
Latvian
Lithuanian
Macedonian
Malay
Maltese
Norwegian
Persian
Polish
Portuguese
Romanian
Russian
Serbian
Slovak
Slovenian
Spanish
Swahili
Swedish
Tamil
Telugu
Thai
Turkish
Ukrainian
Urdu
Vietnamese
Welsh
Yiddish
Type text or a website address
k the words above to view alternate translations. Dismiss
Google Translate for Business: Translator Toolkit Website Translator Global Market Finder
Turn off instant translation
About Google Translate Mobile Privacy Help Send feedback



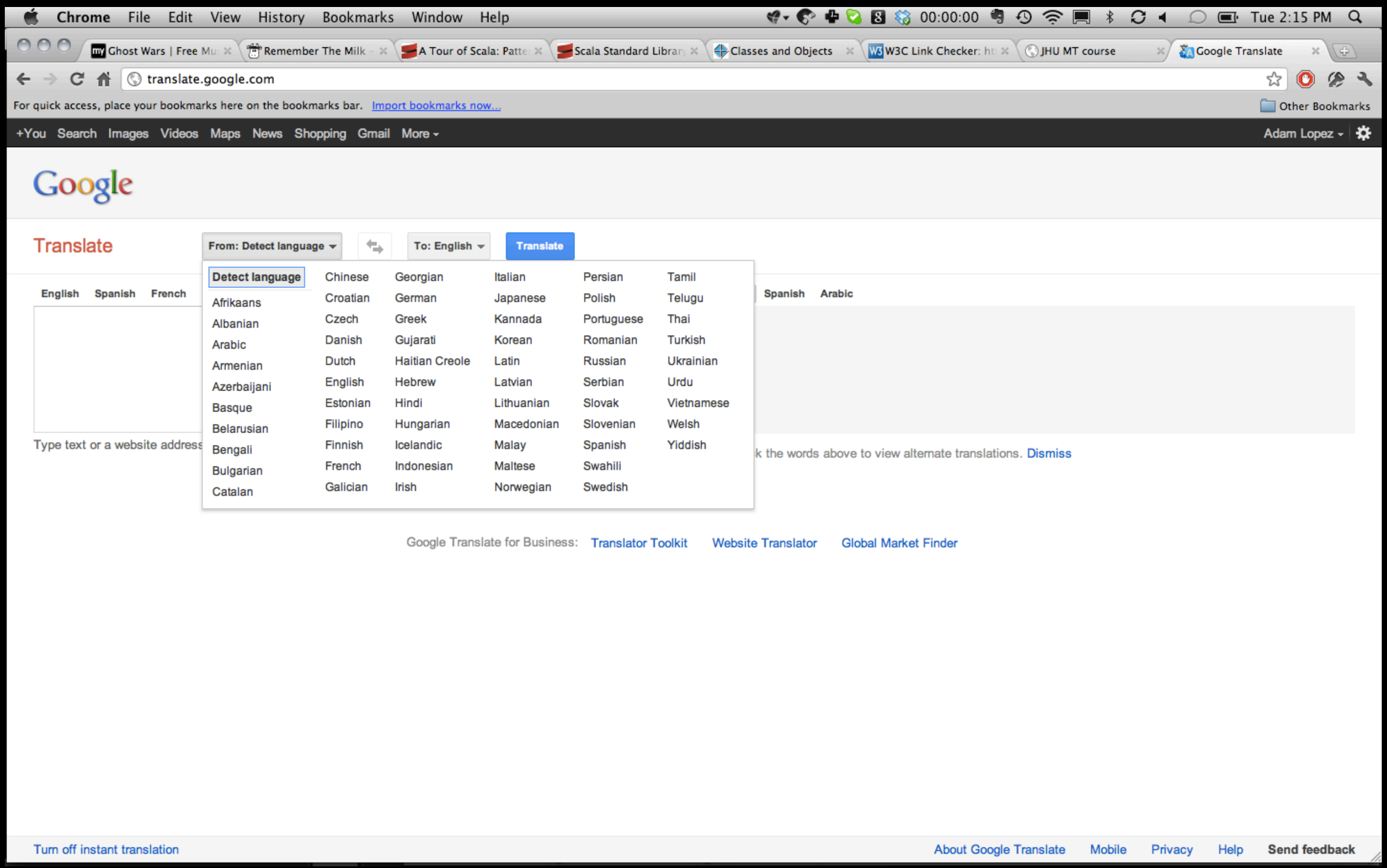
- Key ingredients in Google Translate (2006-2017):



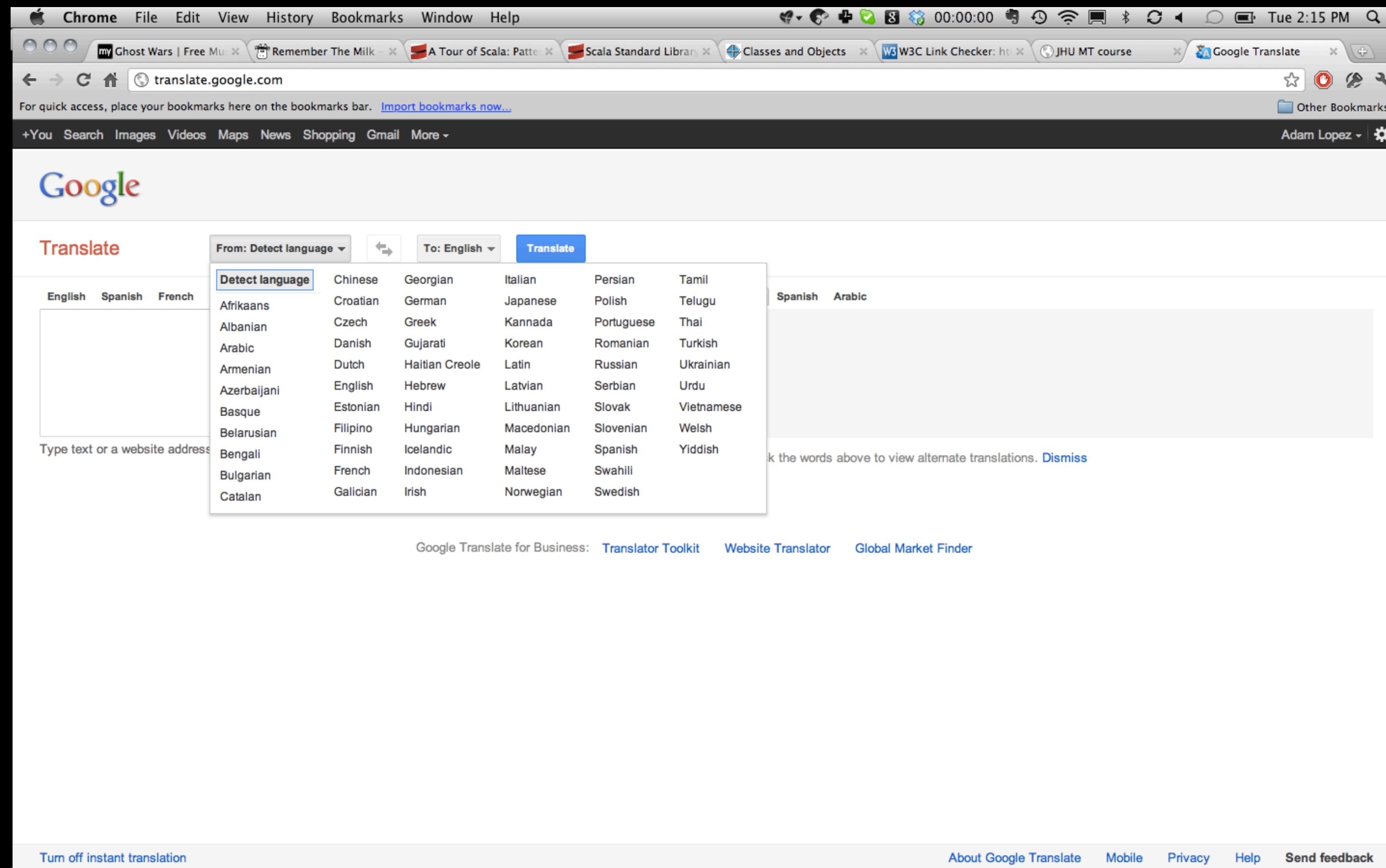
- Key ingredients in Google Translate (2006-2017):
  - Phrase-based translation models



- Key ingredients in Google Translate (2006-2017):
  - Phrase-based translation models
  - ... Learned heuristically from word alignments



- Key ingredients in Google Translate (2006-2017):
  - Phrase-based translation models
  - ... Learned heuristically from word alignments
  - ... Coupled with a huge language model



- Key ingredients in Google Translate (2006-2017):
  - Phrase-based translation models
  - ... Learned heuristically from word alignments
  - ... Coupled with a huge language model
  - ... And decoding with severe pruning heuristics

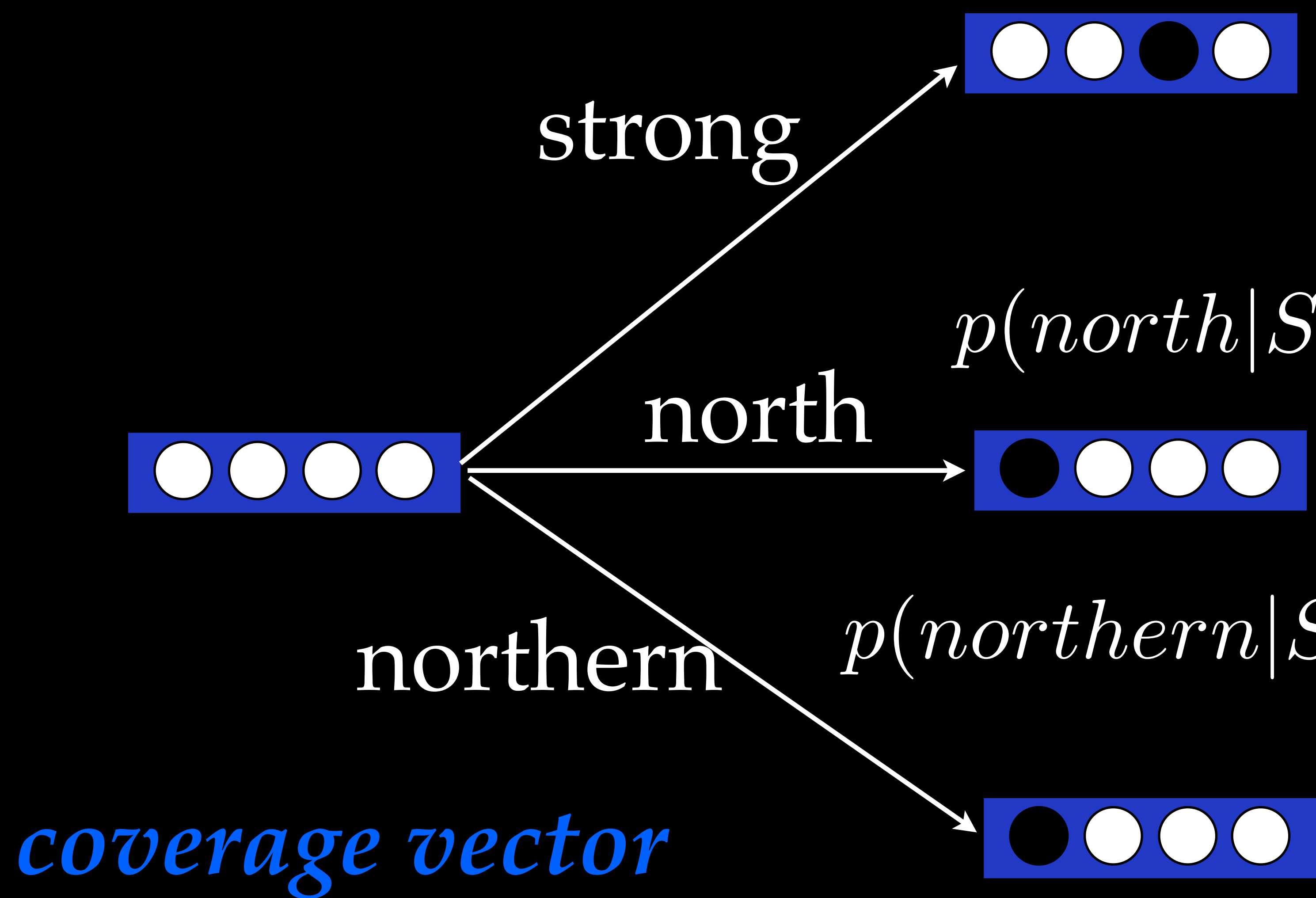
Dynamic programming

北 风 呼啸 。

There are $5^n n!$ target sentences.

But there are only $O(5n)$ ways to start them.

Dynamic programming

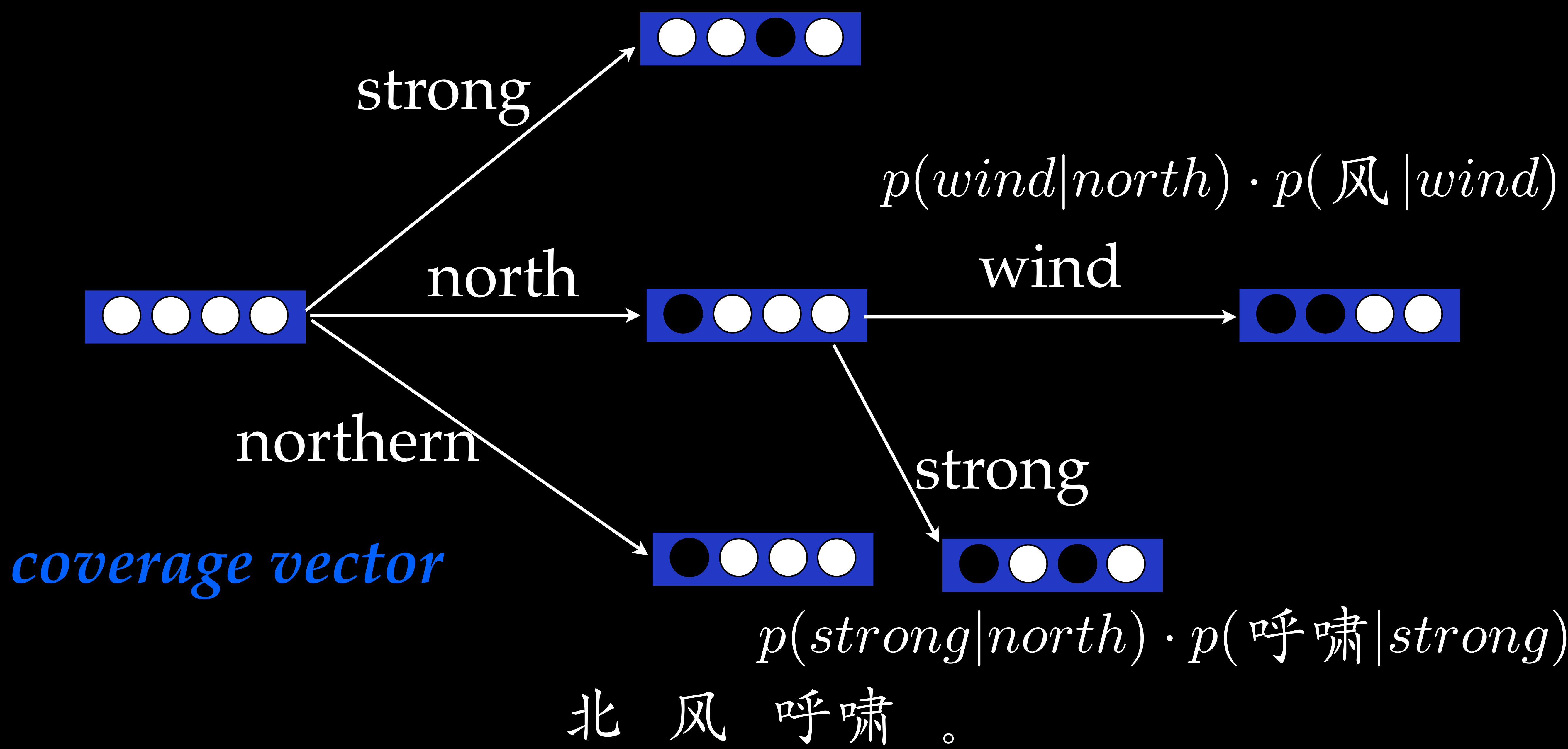


$$p(strong|START) \cdot p(呼啸|strong)$$

$$p(north|START) \cdot p(北|north)$$

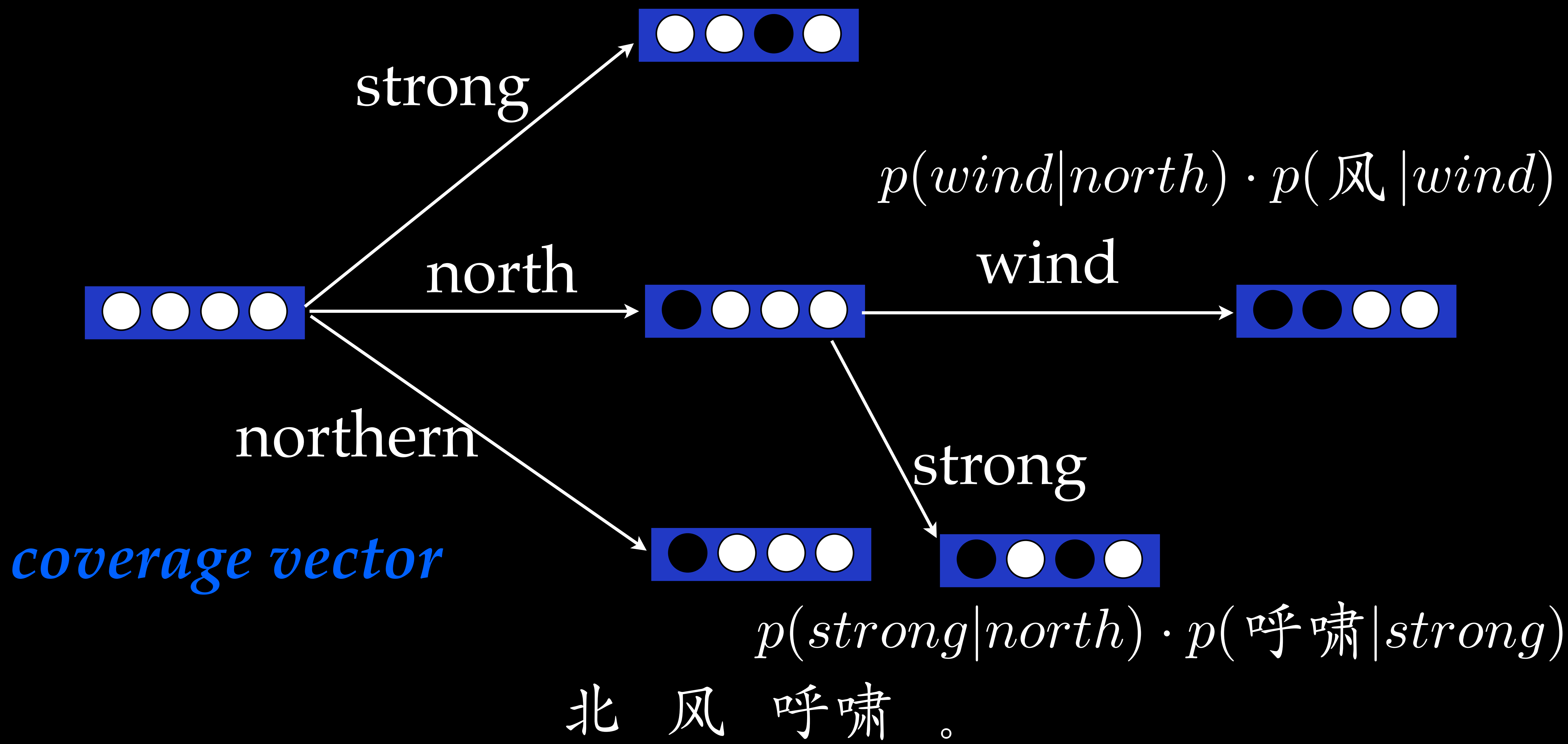
$$p(northern|START) \cdot p(北|northern)$$

Dynamic programming



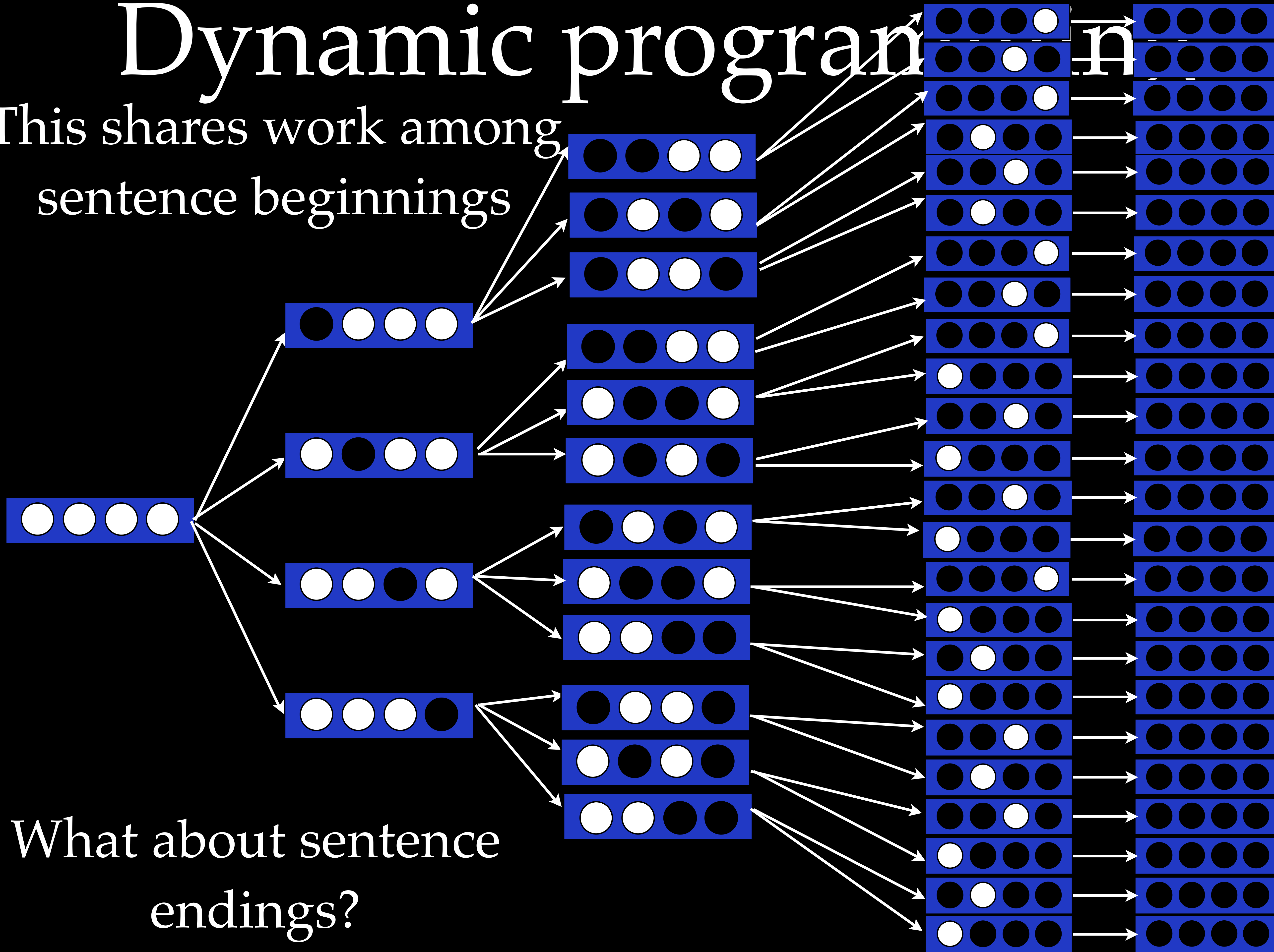
Dynamic programming

This shares work among sentence beginnings



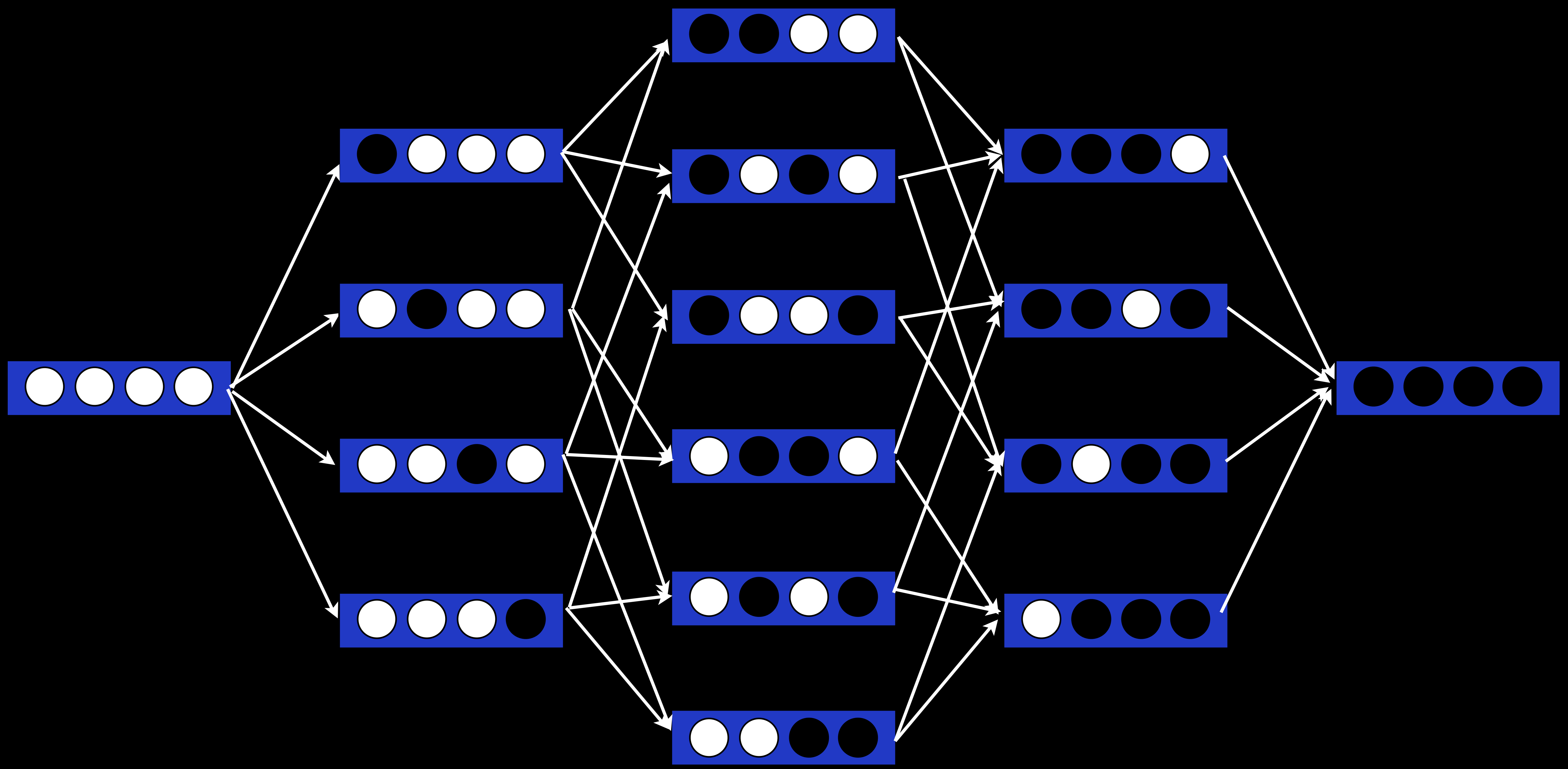
Dynamic programming

This shares work among sentence beginnings

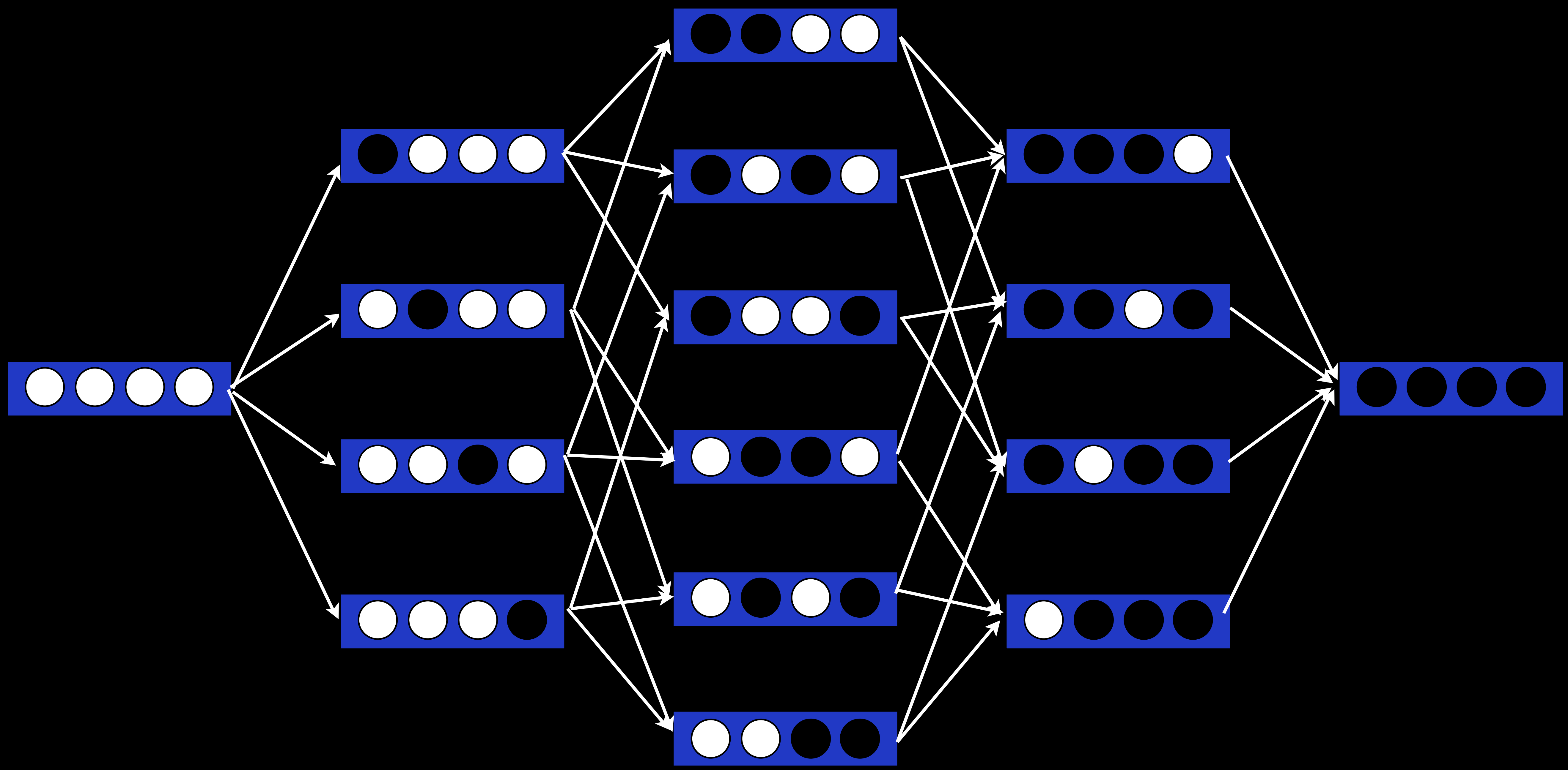


What about sentence endings?

Approximation: Pruning

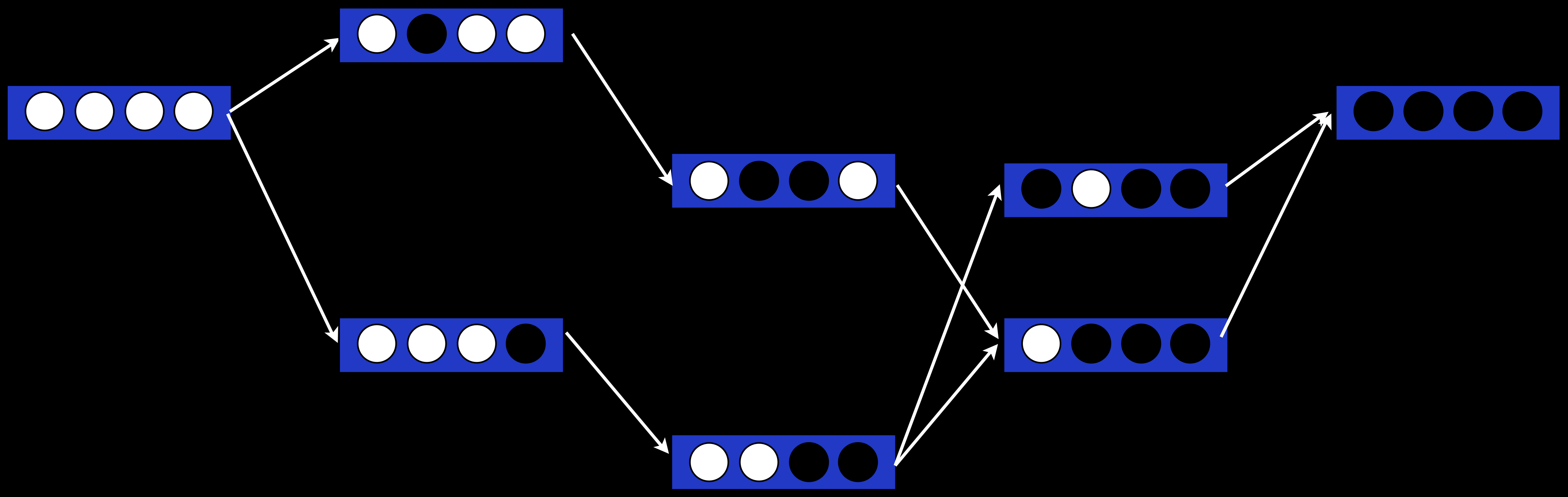


Approximation: Pruning



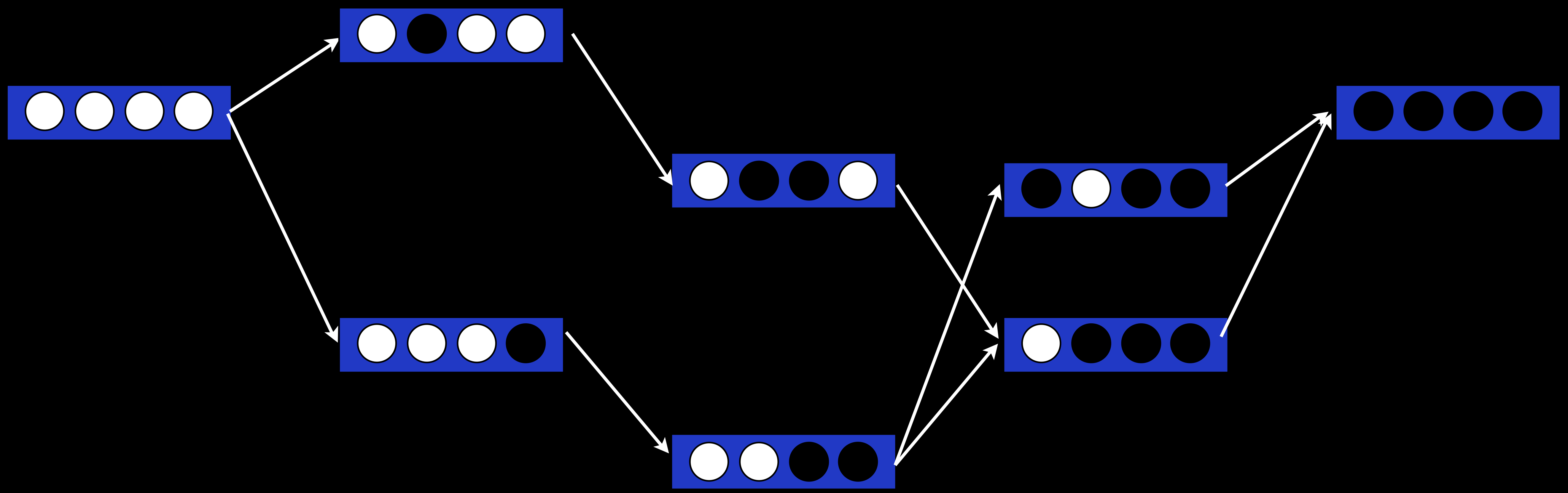
Idea: prune states by cost of shortest path to them

Approximation: Pruning



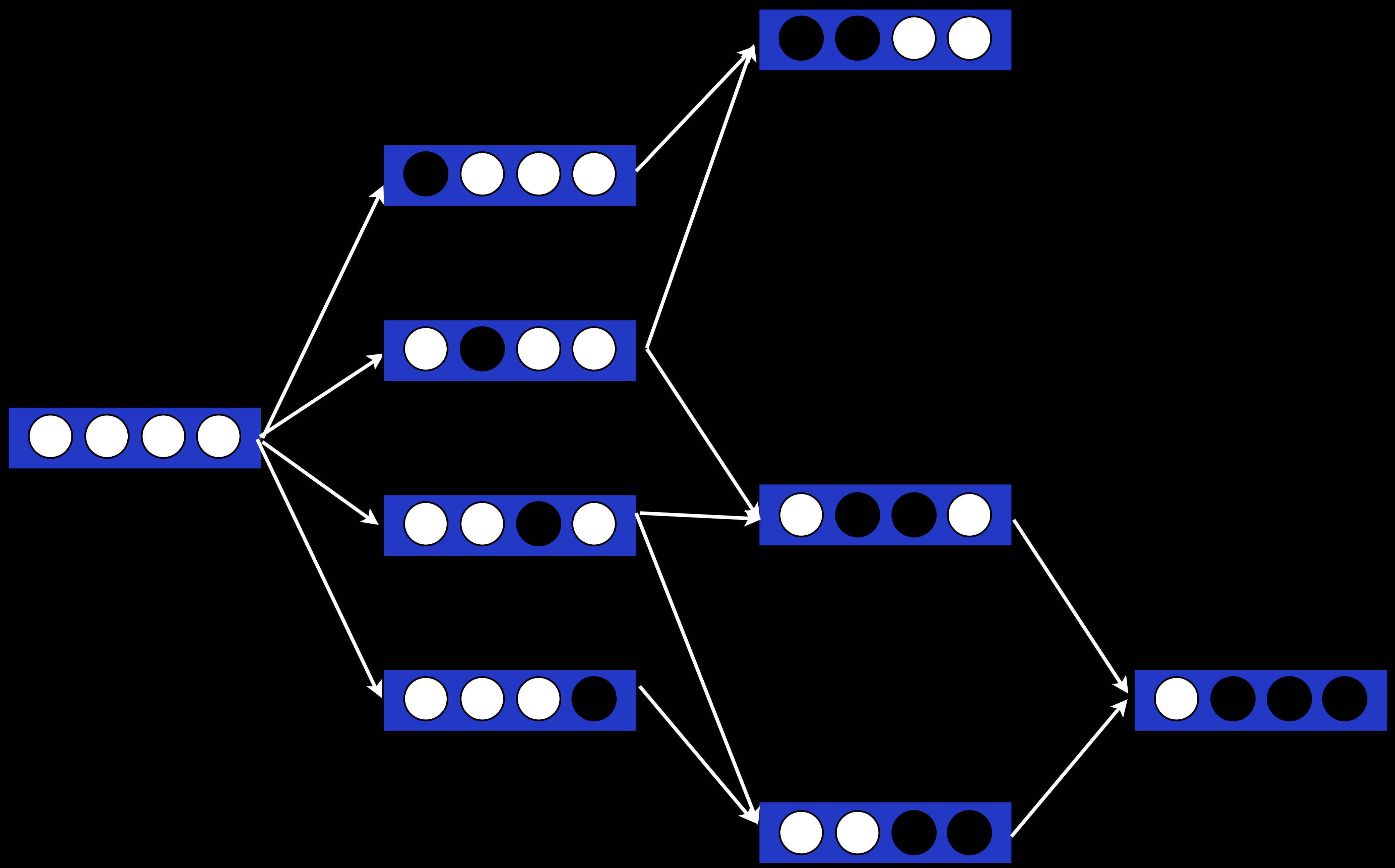
Idea: prune states by accumulated path length

Approximation: Pruning



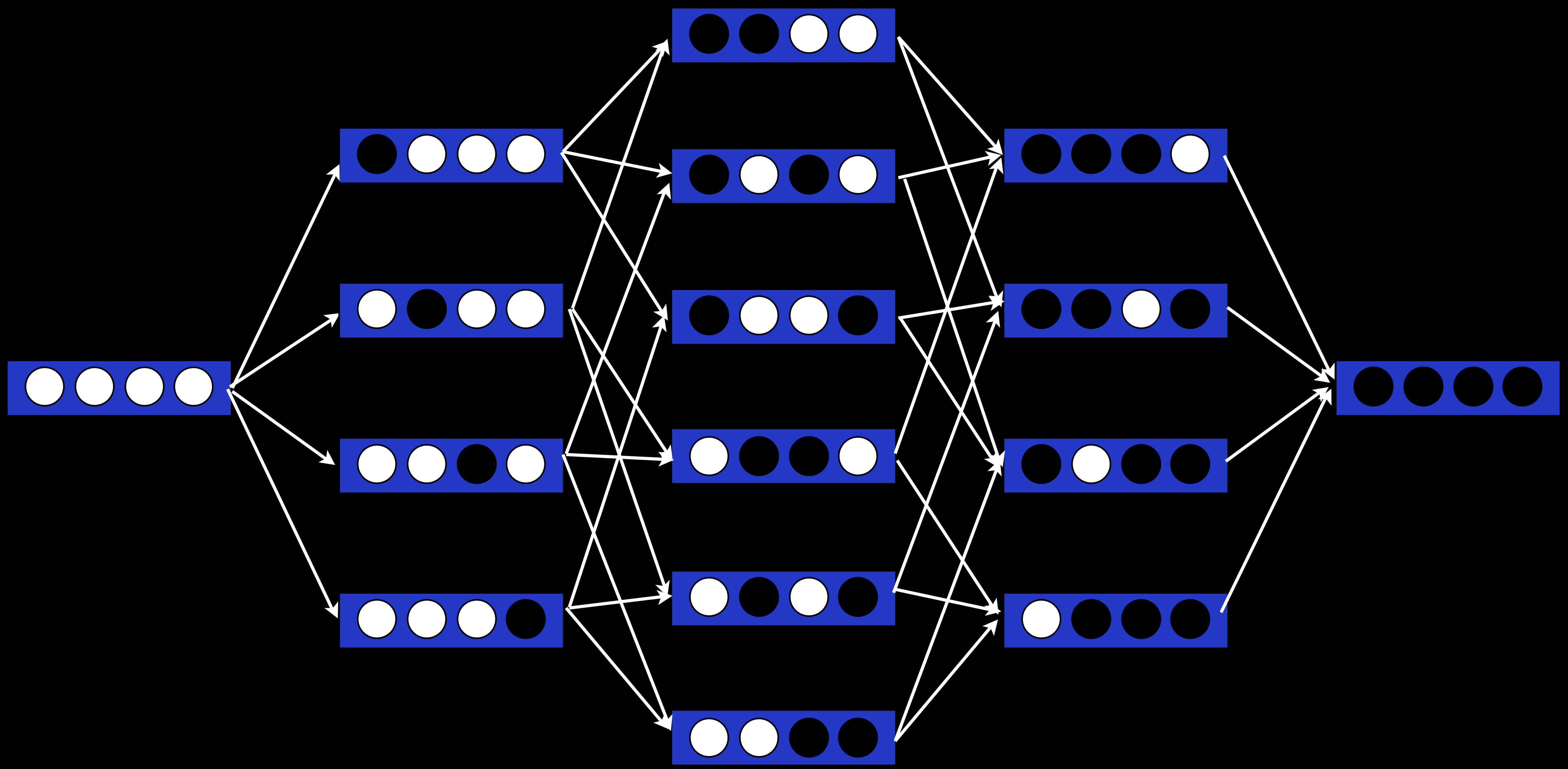
Ideal result: only high-probability paths enumerated

Approximation: Pruning

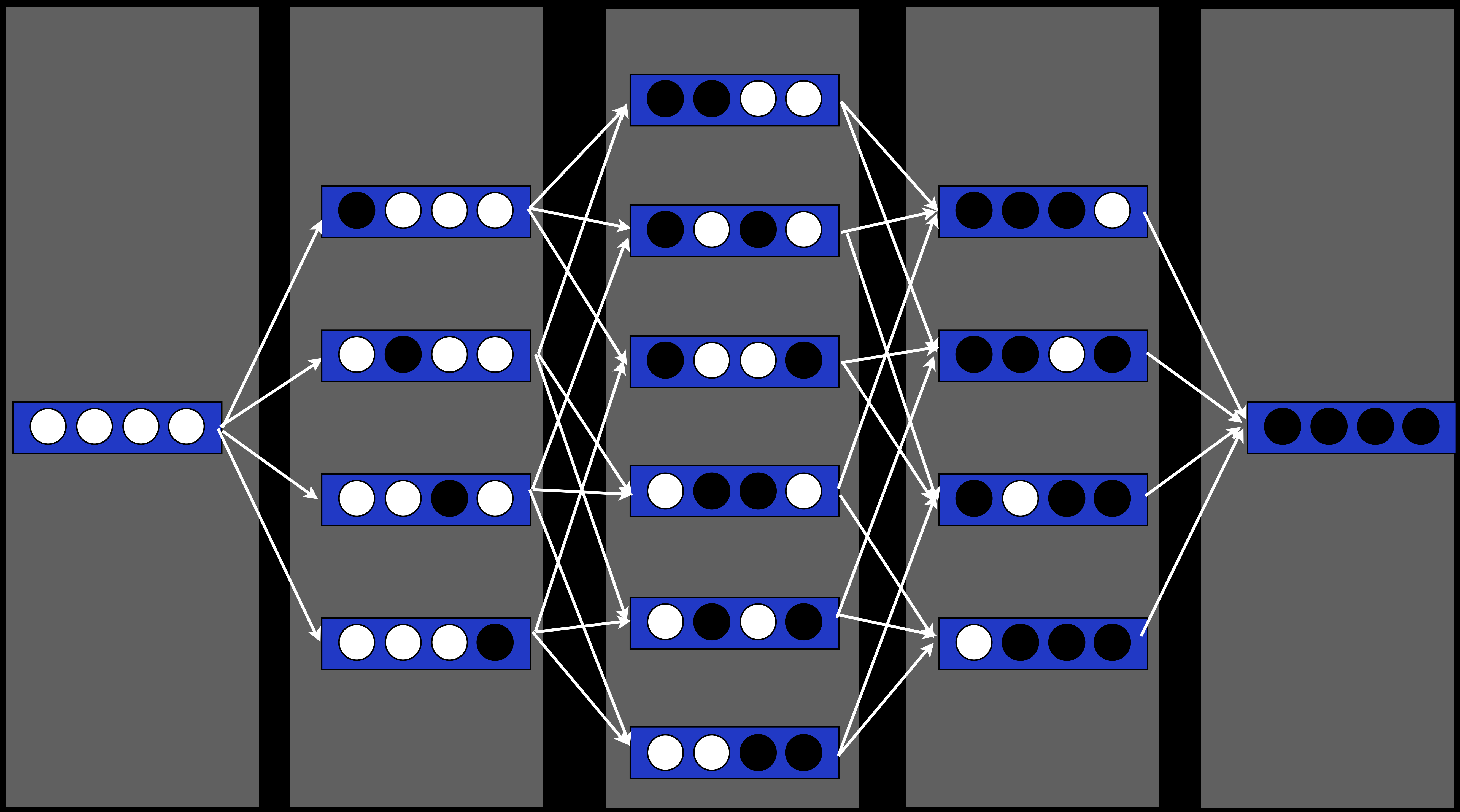


Actual result: longer paths have lower probability!

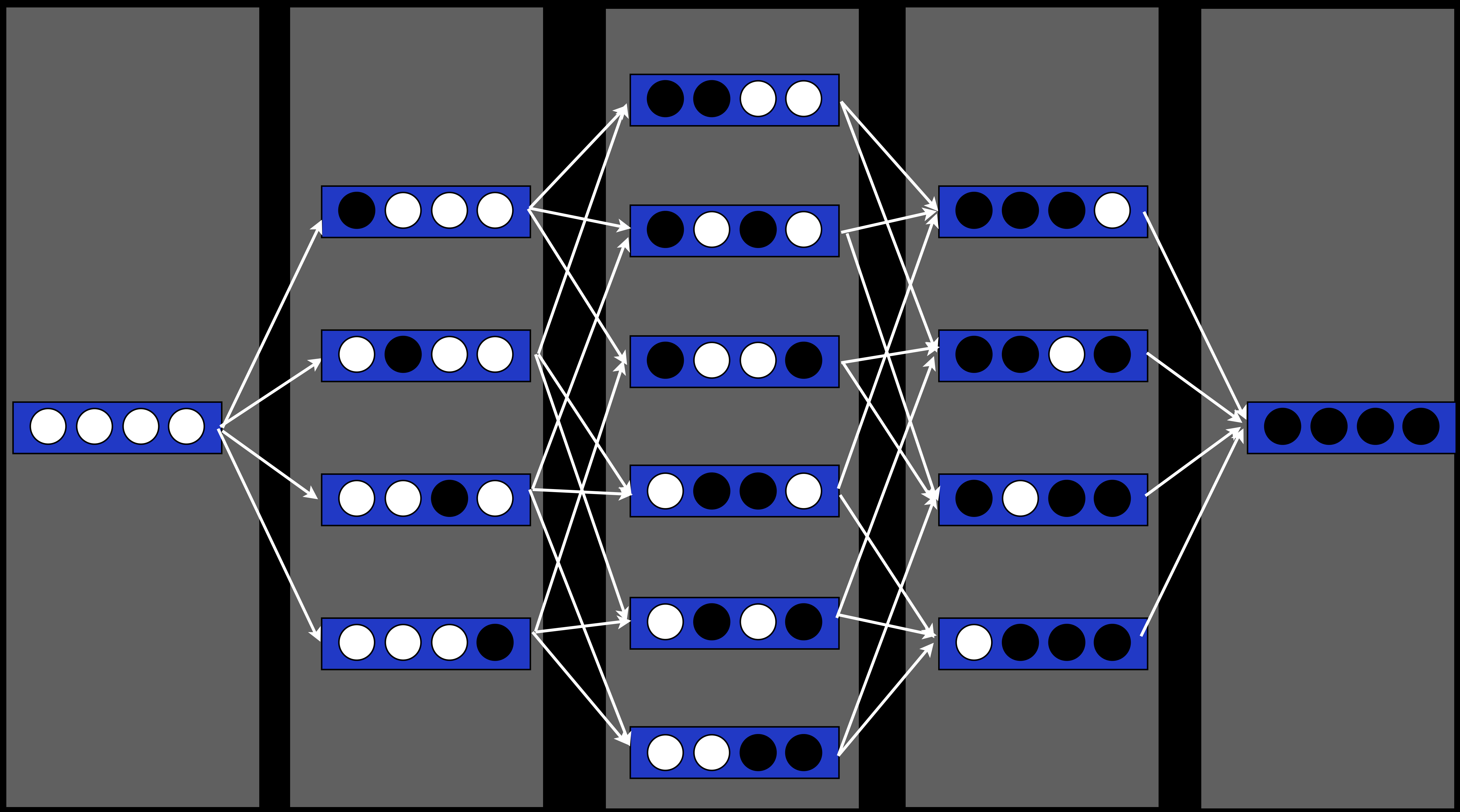
Approximation: Pruning



Approximation: Pruning

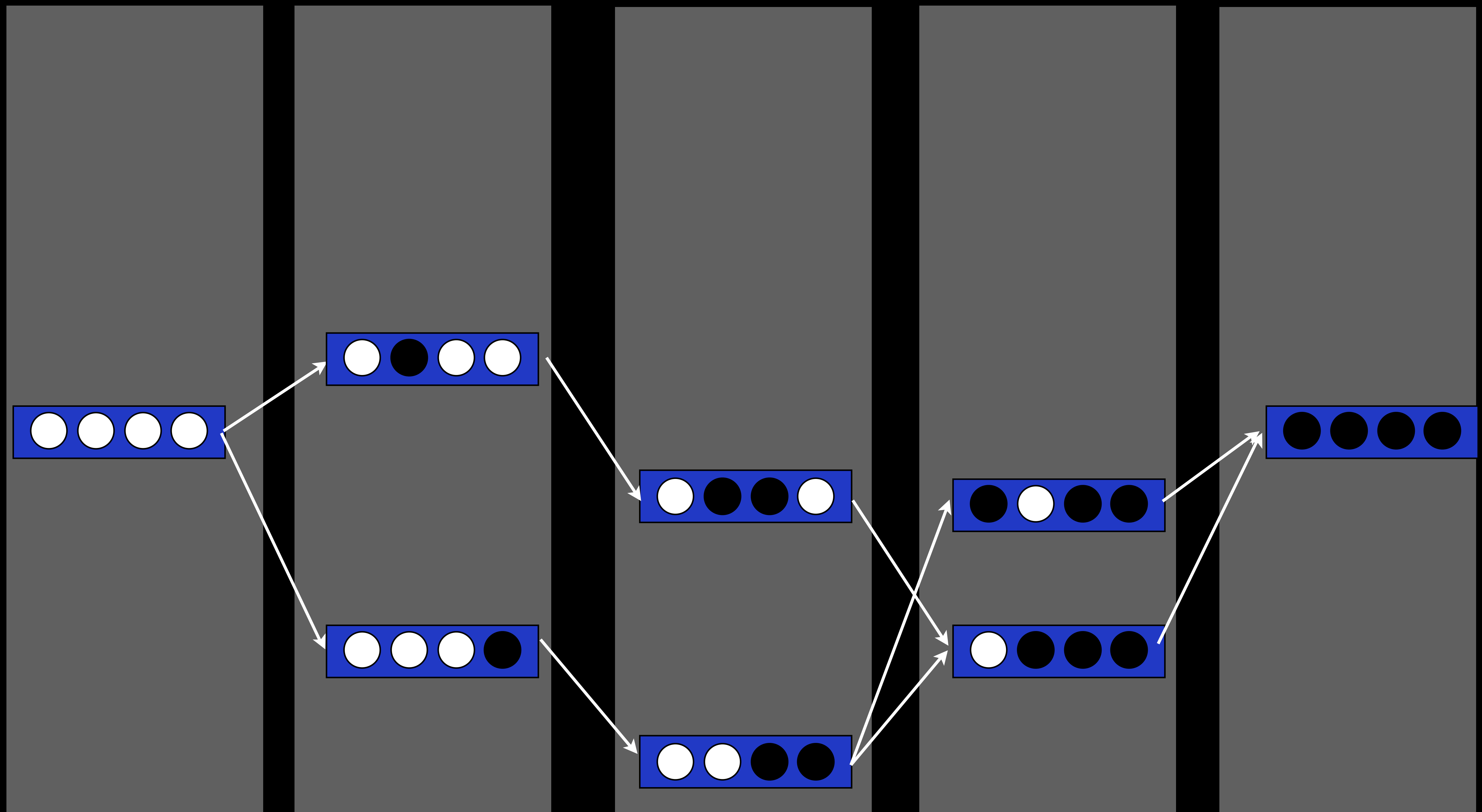


Approximation: Pruning



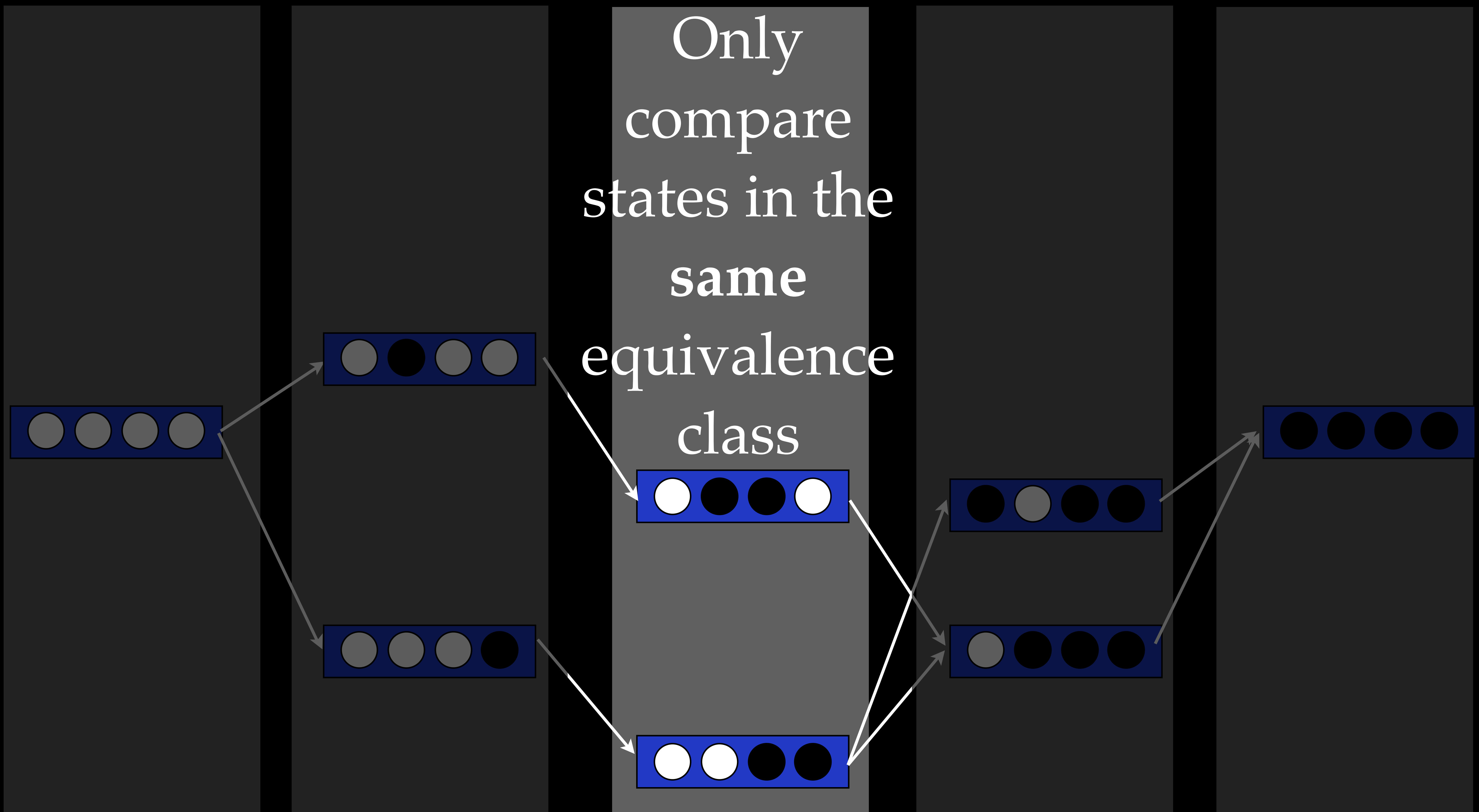
Solution: Group states by number of covered words.

Approximation: Pruning



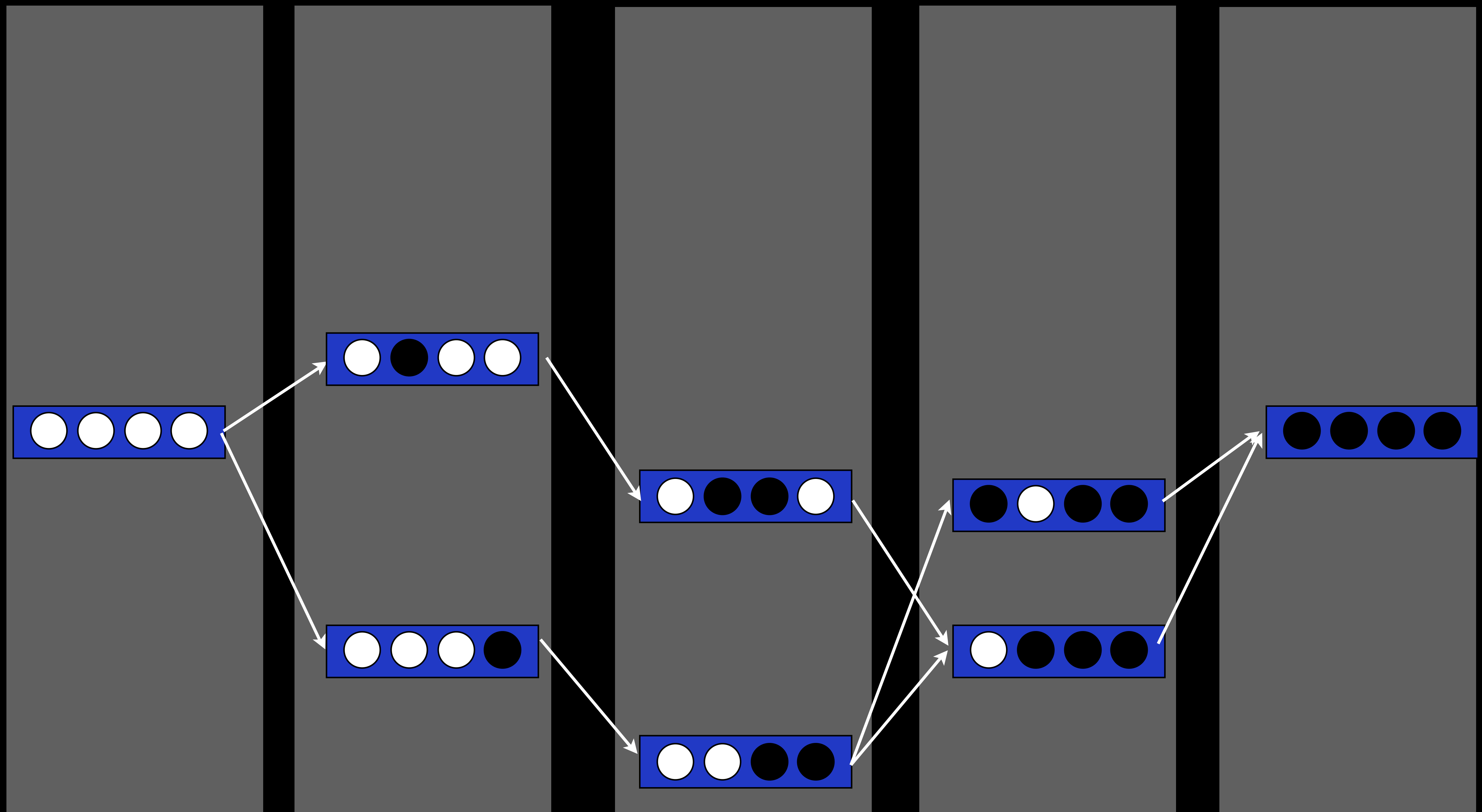
Solution: Group states by number of covered words.

Approximation: Pruning



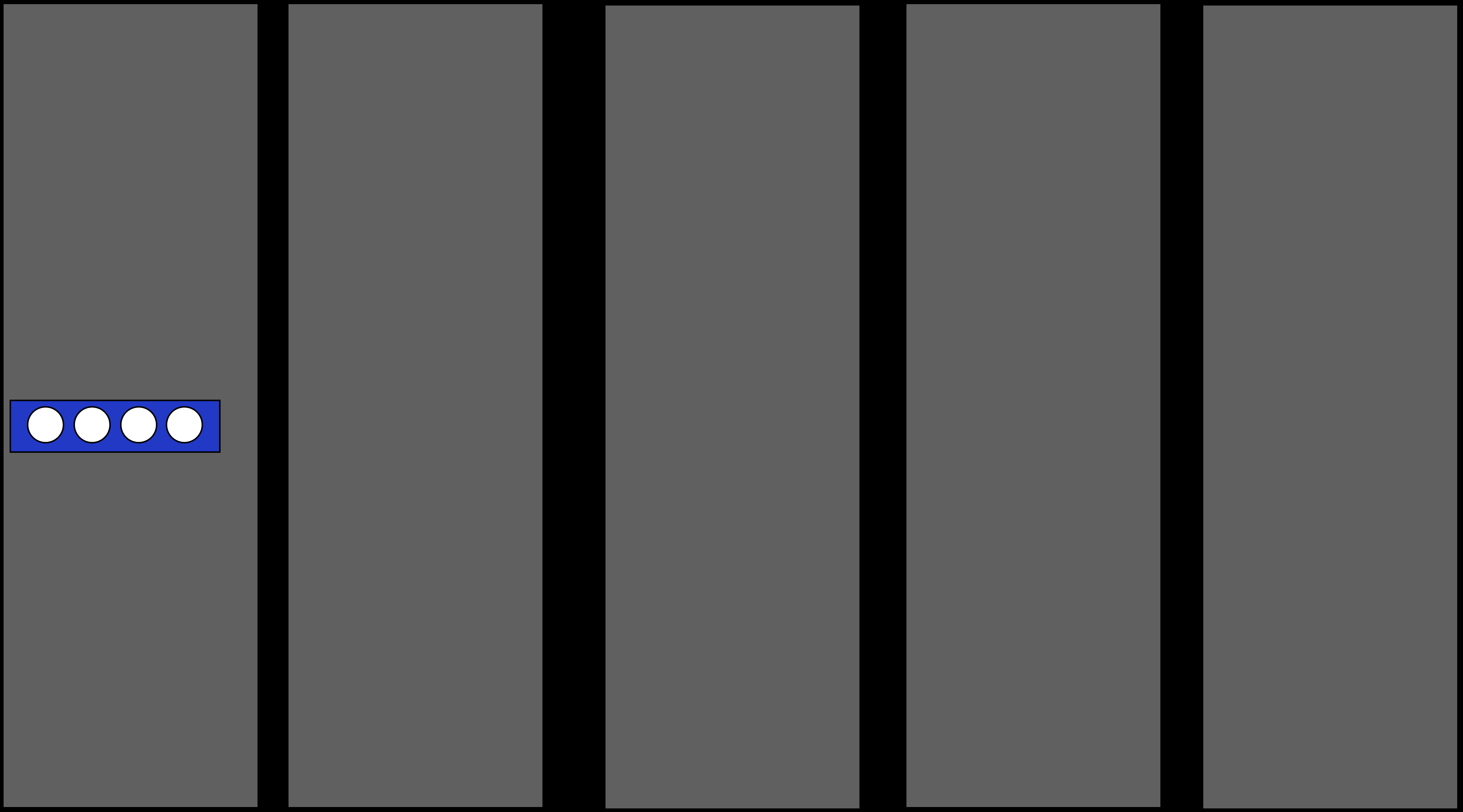
Solution: Group states by number of covered words.

Approximation: Pruning



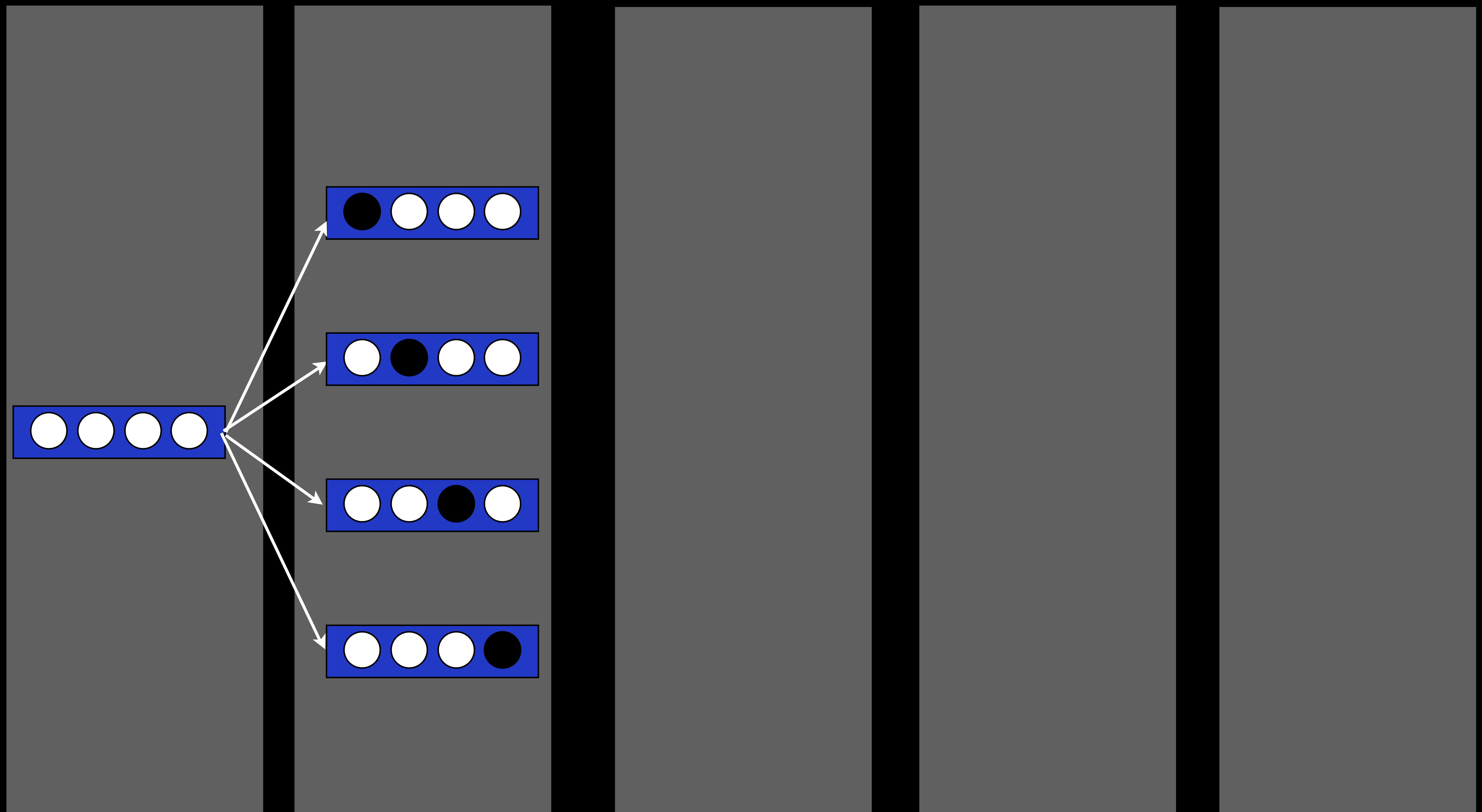
"Stack" decoding: a linear-time approximation

Approximation: Pruning



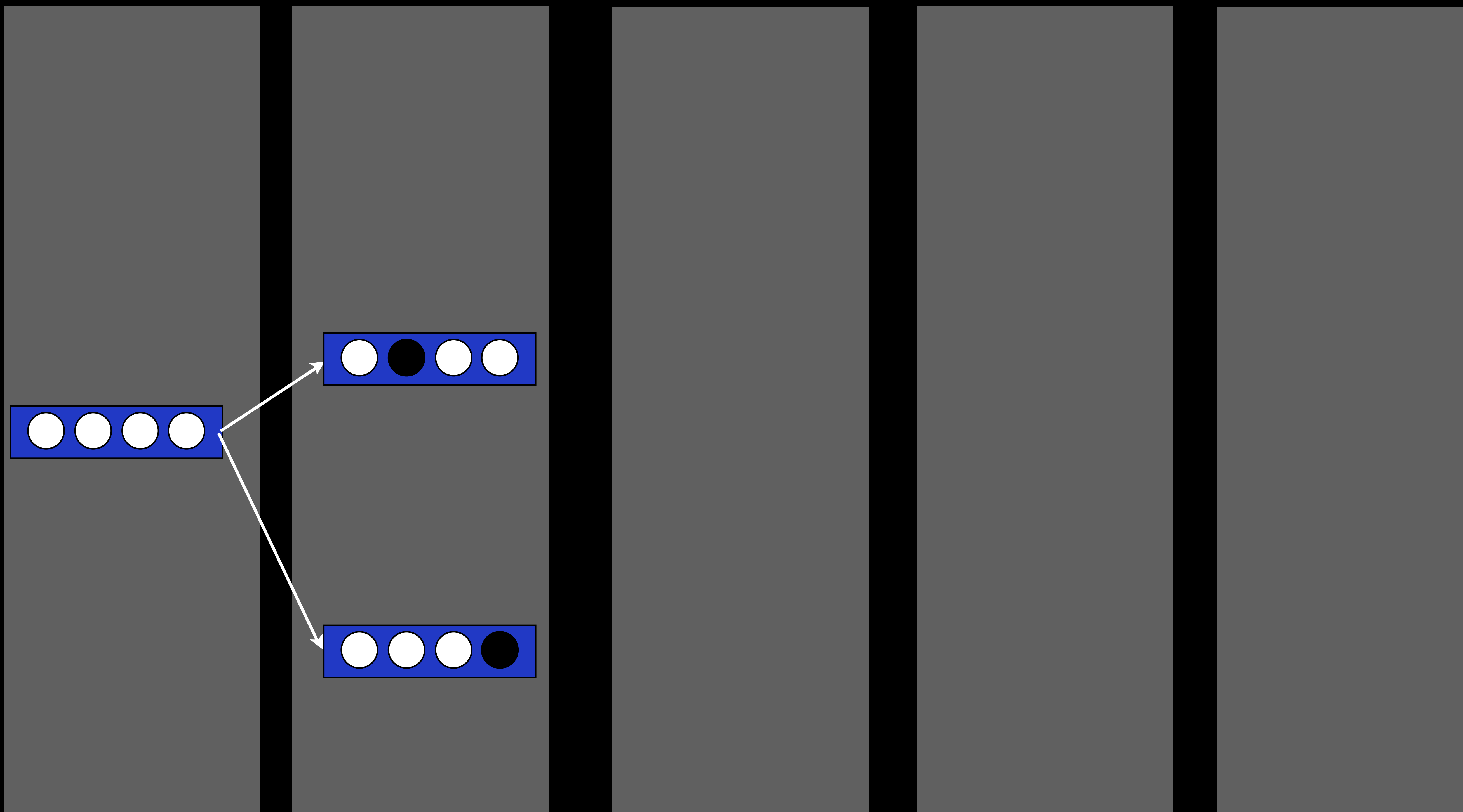
"Stack" decoding: a linear-time approximation

Approximation: Pruning



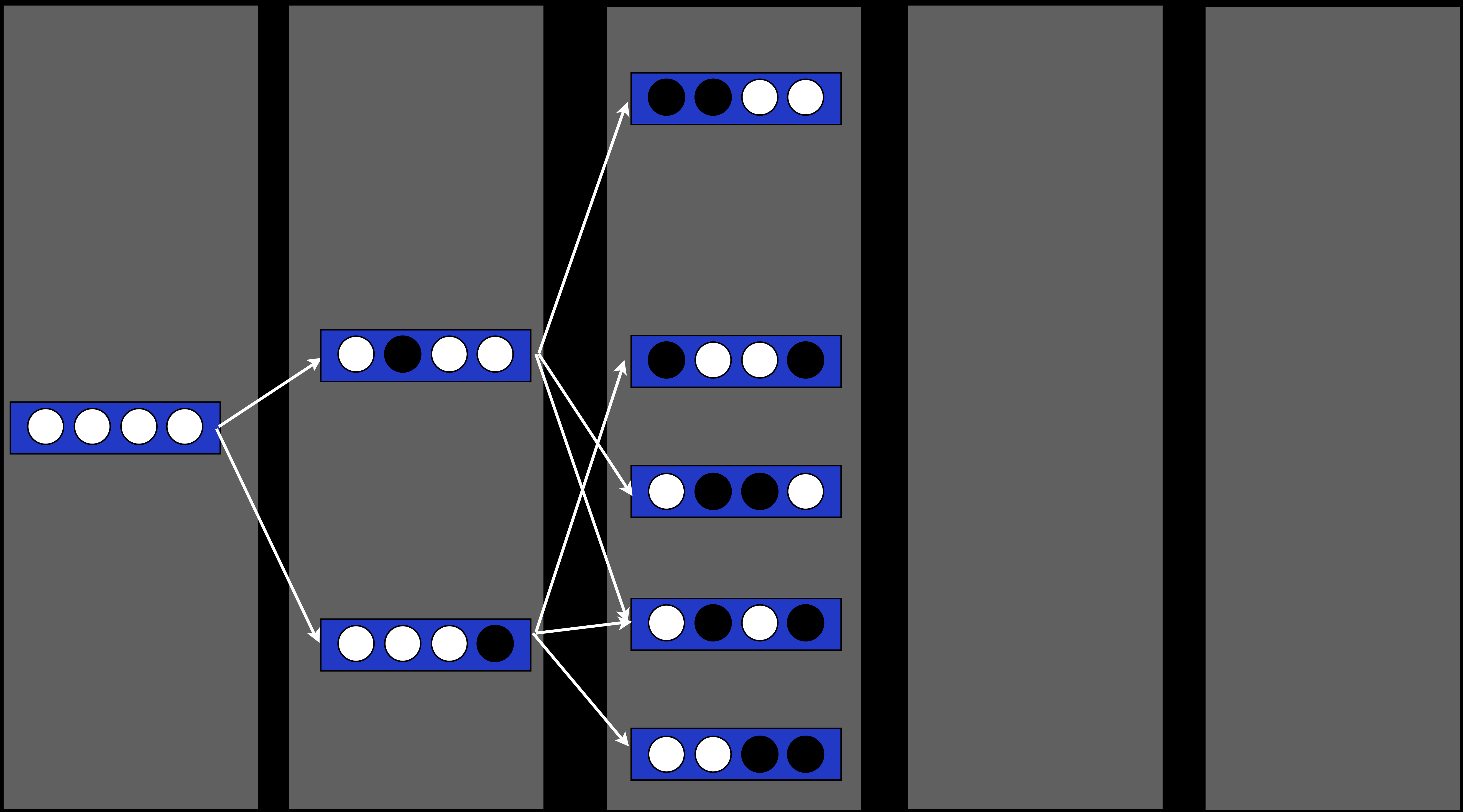
"Stack" decoding: a linear-time approximation

Approximation: Pruning



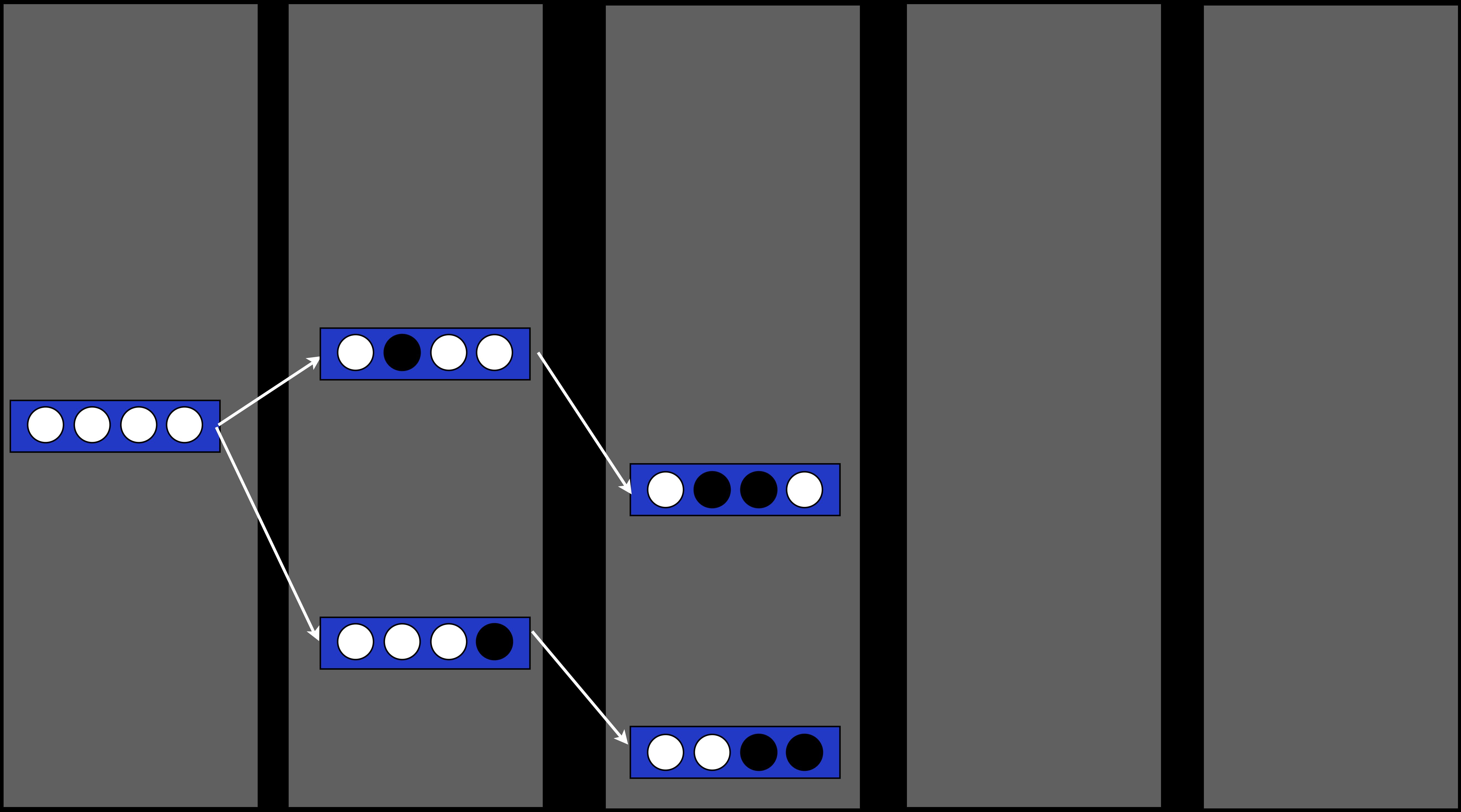
"Stack" decoding: a linear-time approximation

Approximation: Pruning



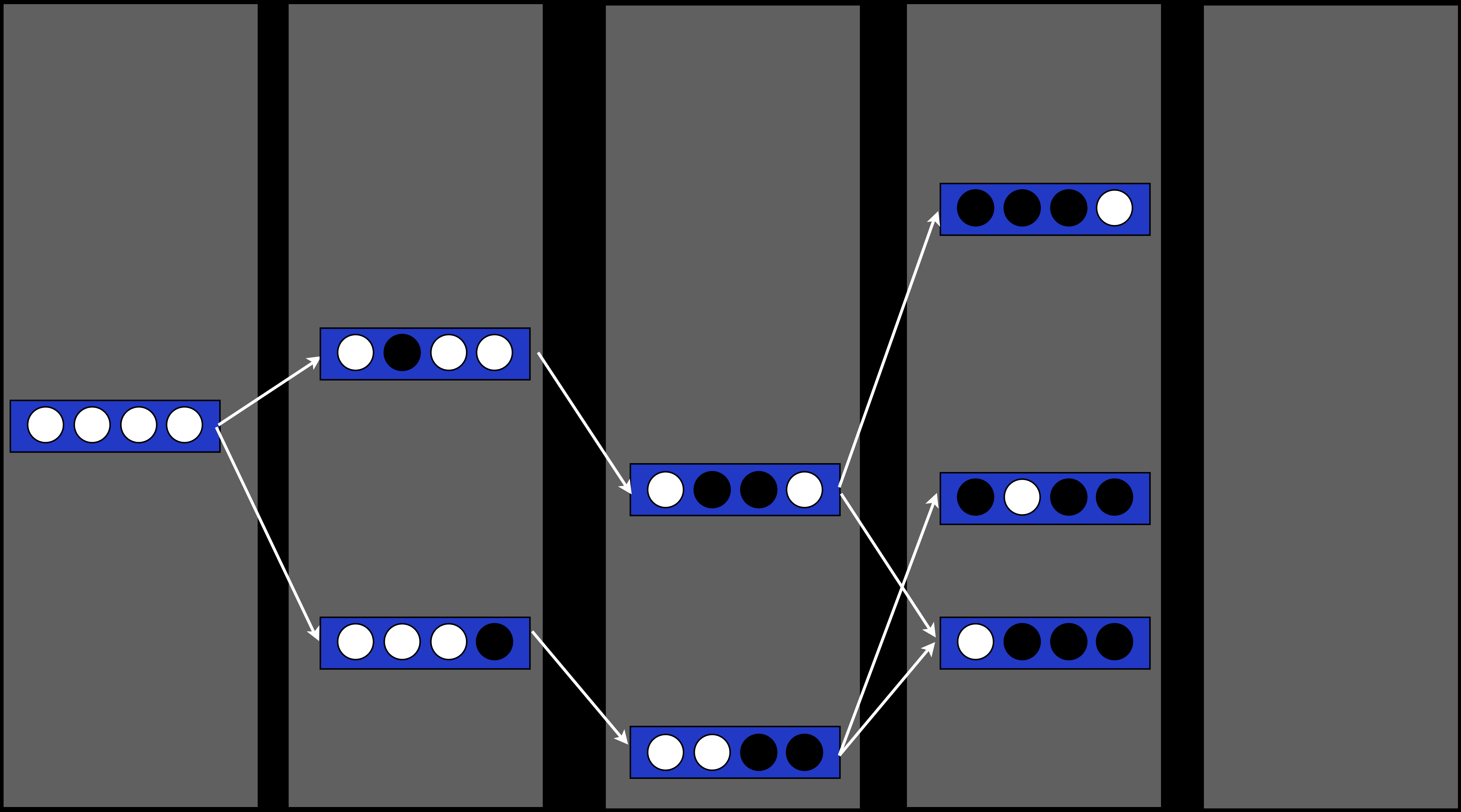
"Stack" decoding: a linear-time approximation

Approximation: Pruning



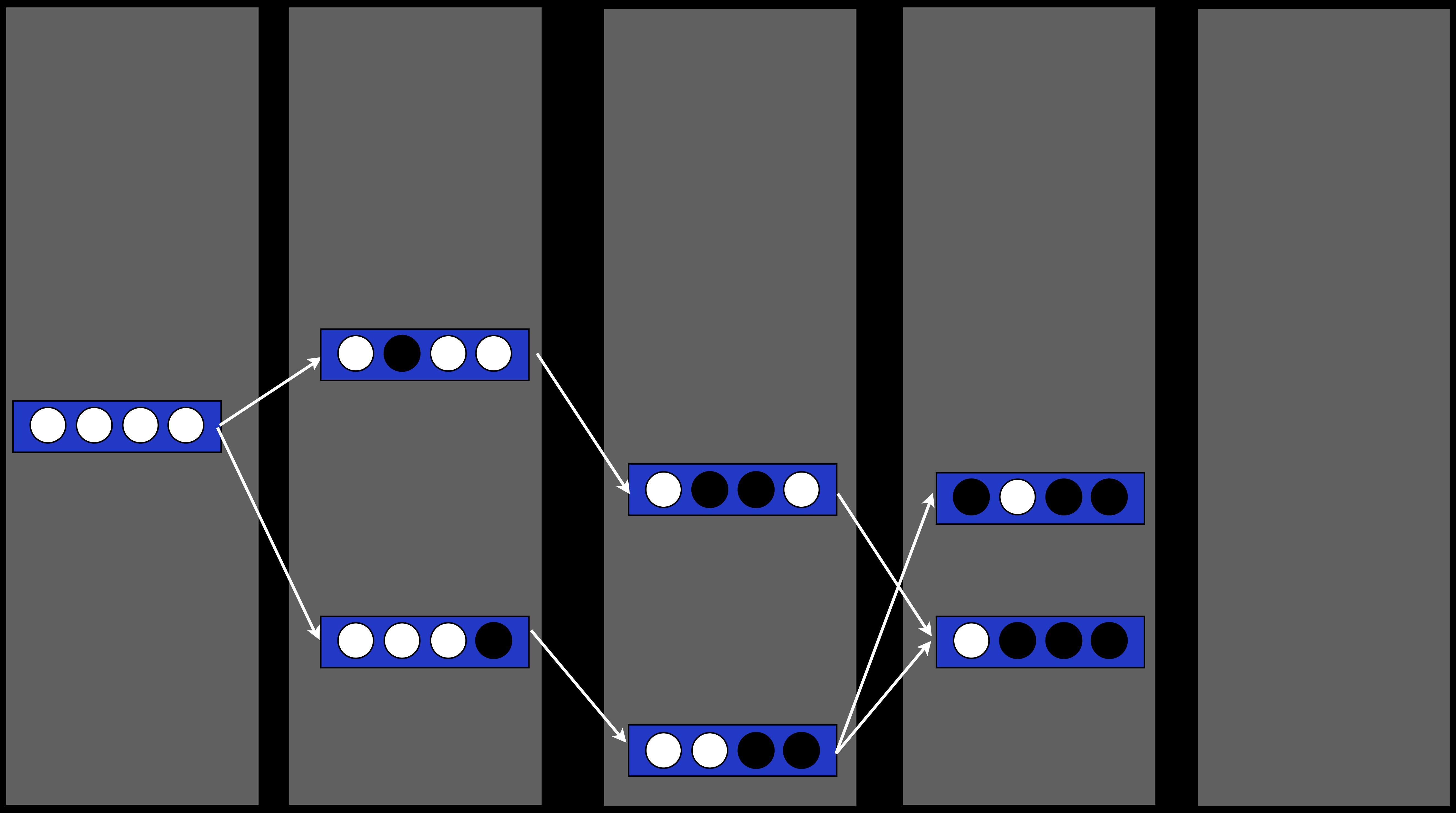
"Stack" decoding: a linear-time approximation

Approximation: Pruning



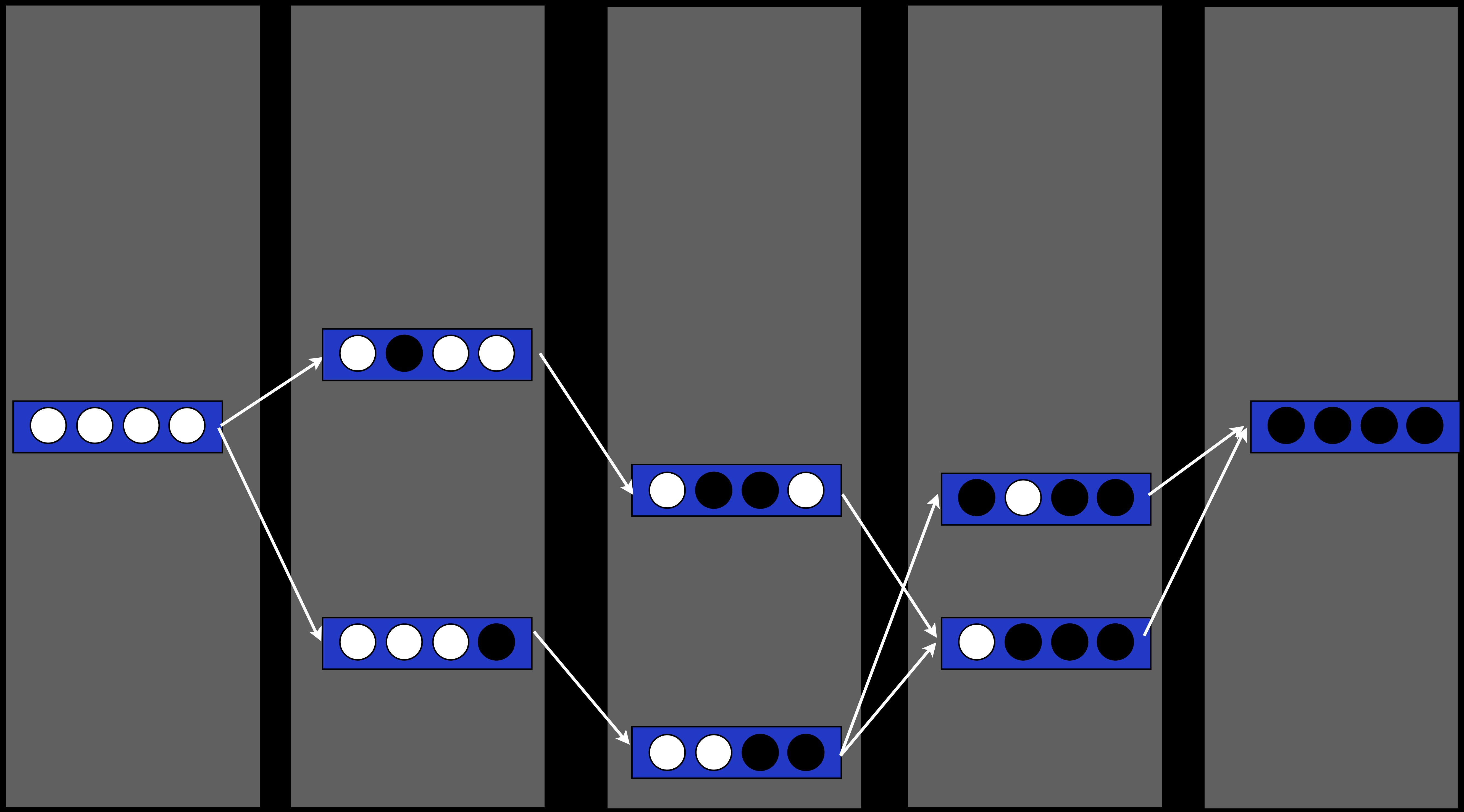
"Stack" decoding: a linear-time approximation

Approximation: Pruning



"Stack" decoding: a linear-time approximation

Approximation: Pruning



“Stack” decoding: a linear-time approximation

Approximation: distortion limits

the sky

虽然 北 风 呼啸 ， 但 天空 依然 十分 清澈 。

Approximation: distortion limits

number of vertices: $O(2^n)$

the sky

虽然 北 风 呼啸 , 但 天空 依然 十分 清澈 。

Approximation: distortion limits

number of vertices: $O(2^n)$

the sky

$\boxed{\text{虽然 北 风 呼啸}}$, $\boxed{\text{但 天空}}$ 依然 十分 清澈 。

$$d = 4$$

window

Approximation: distortion limits

number of vertices: $O(2^n)$

the sky

虽然 北 风 呼啸 , 但 天空 依然 十分 清澈 。

$d = 4$

outside window to left: covered

window

outside window to right: uncovered

Approximation: distortion limits

number of vertices: $O(n2^d)$

the sky

虽然 北 风 呼啸 , 但 天空 依然 十分 清澈 。

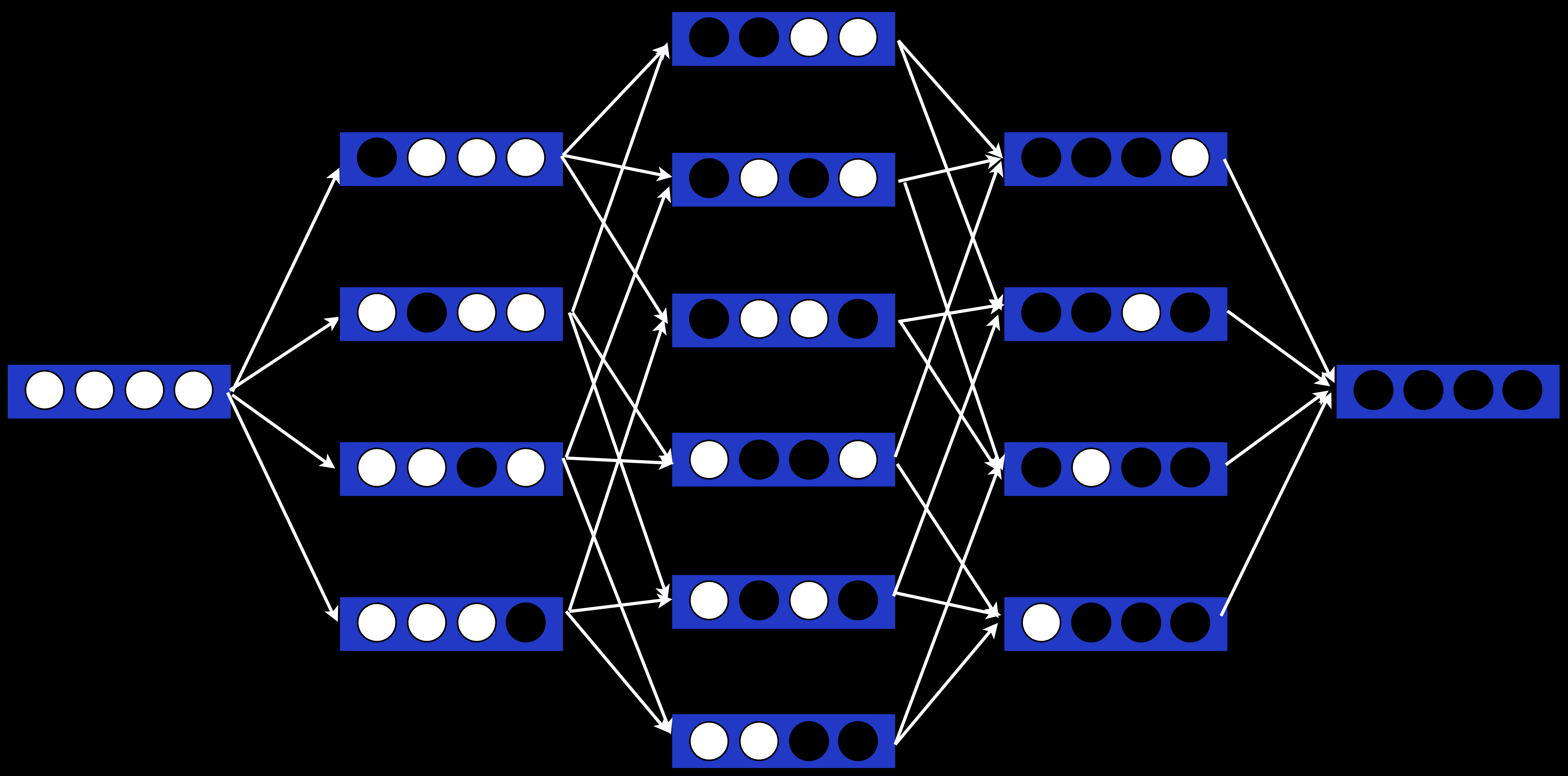
outside window
to left: covered

$d = 4$

window

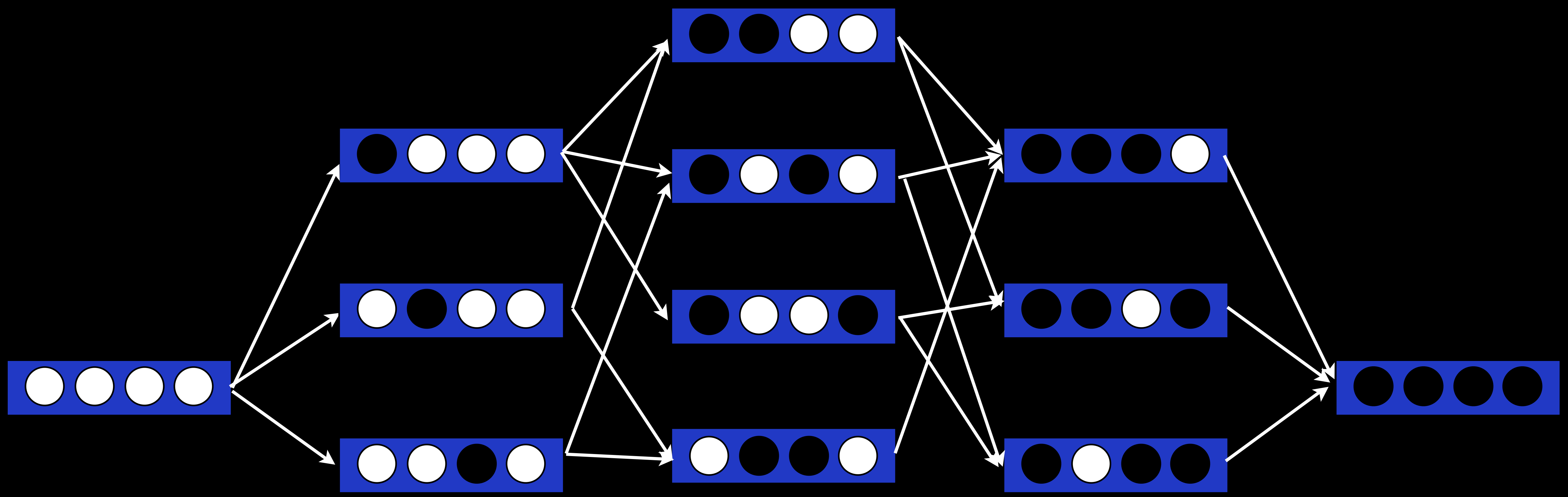
outside window
to right: uncovered

Approximation: distortion limits



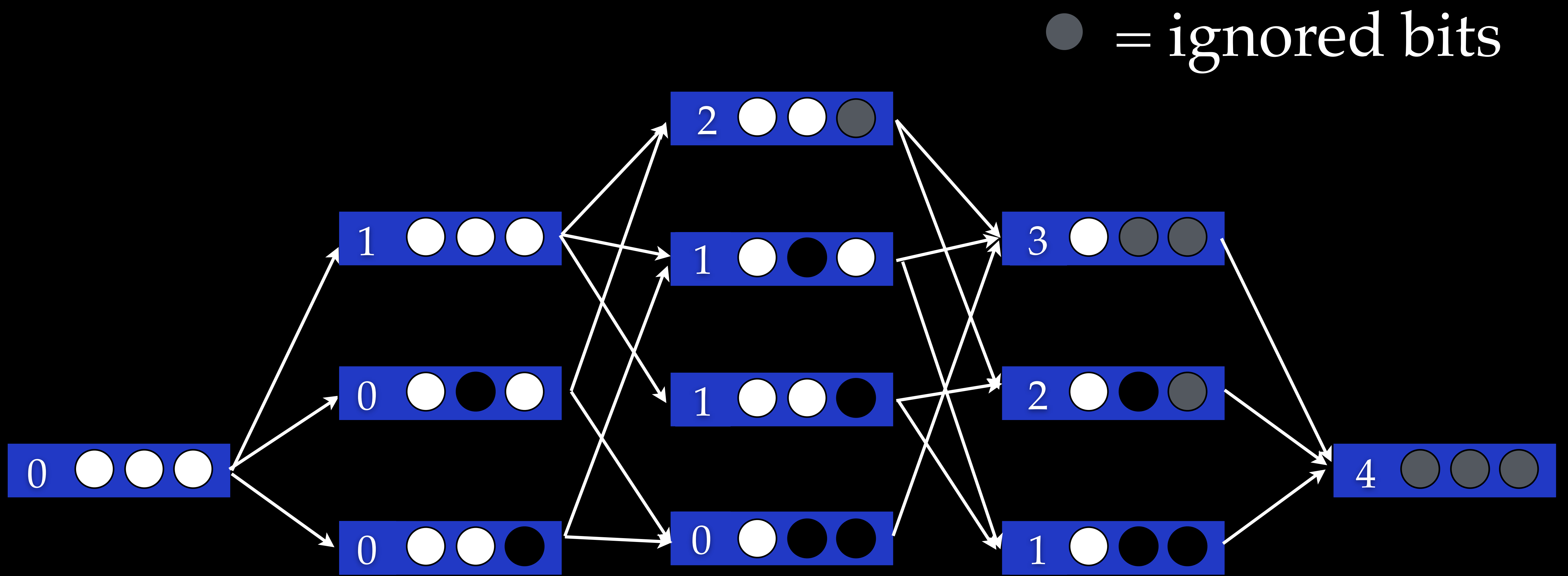
no distortion limit

Approximation: distortion limits



distortion limit = 3

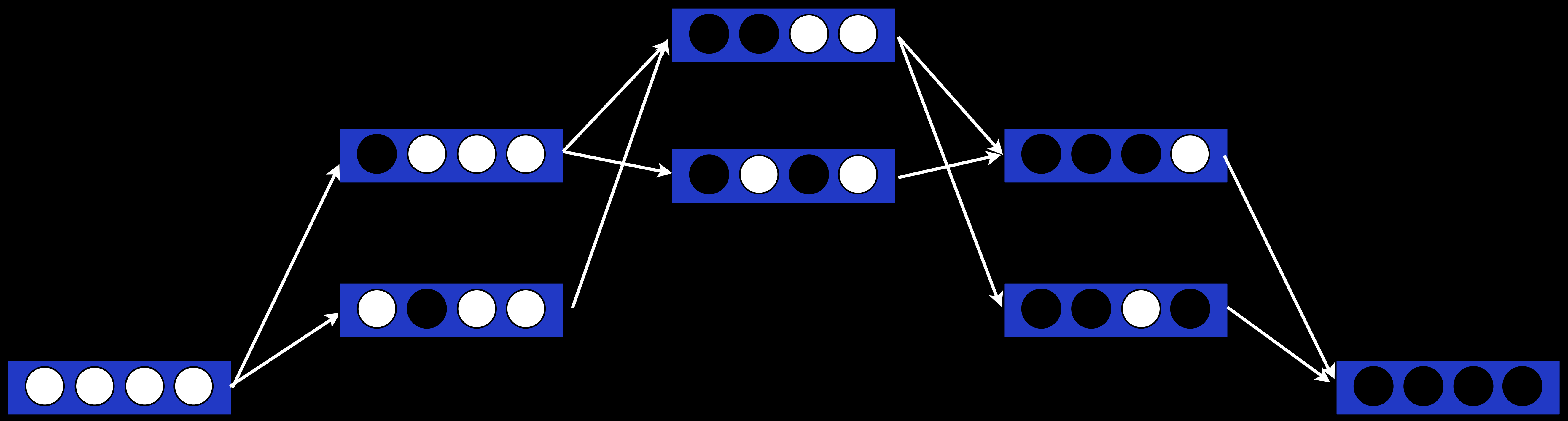
Approximation: distortion limits



alternative representation with length of covered section + bitmap of window

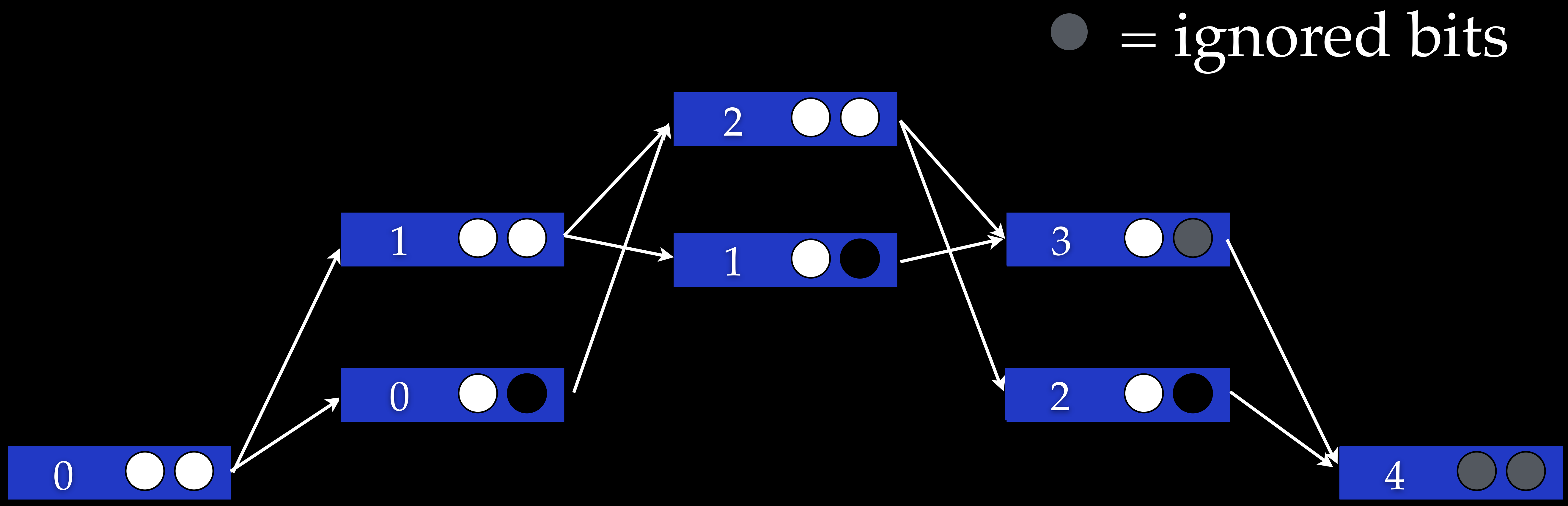
distortion limit = 3

Approximation: distortion limits



distortion limit = 2

Approximation: distortion limits

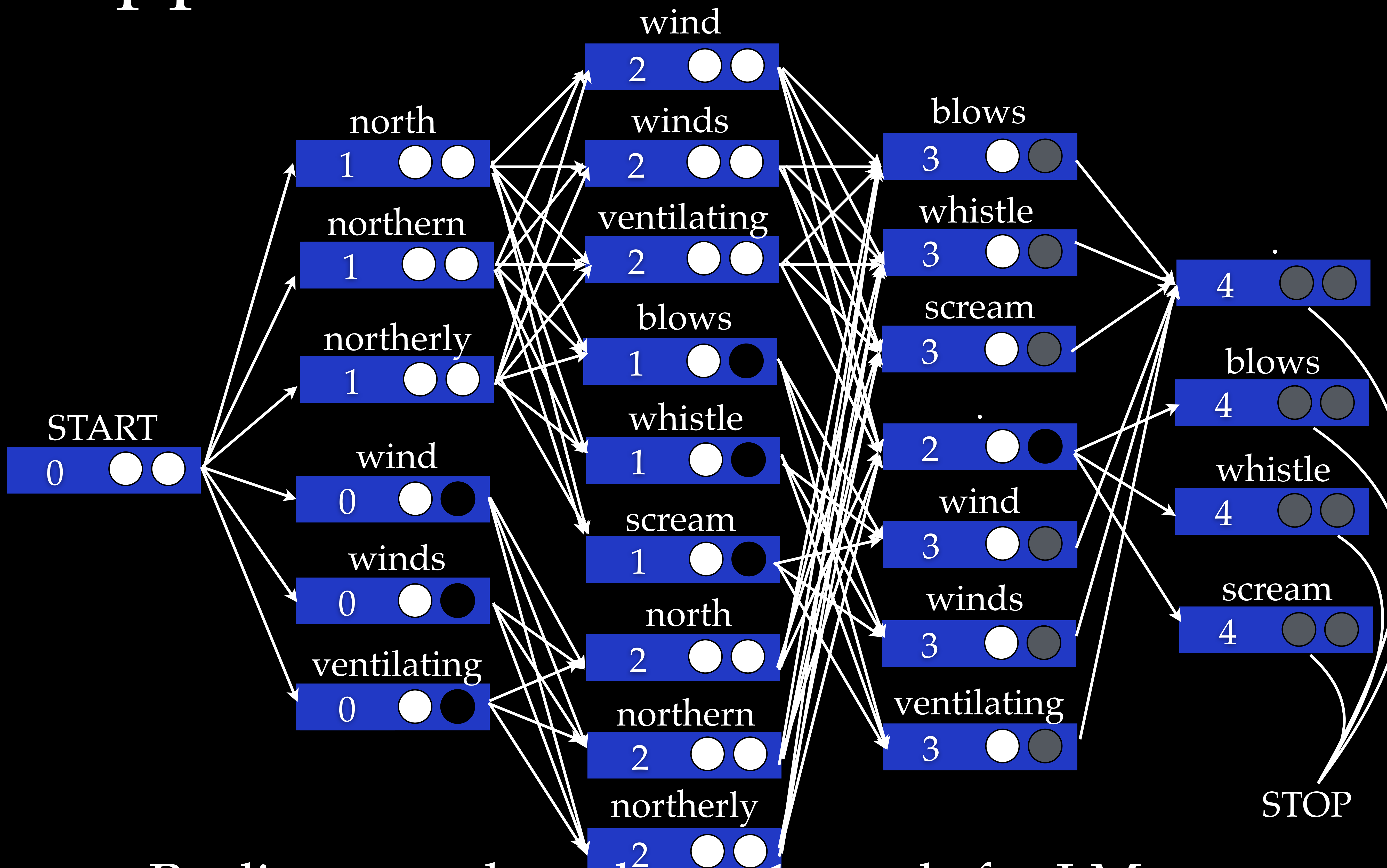


What is this representation missing?

alternative representation with length of covered section + bitmap of window

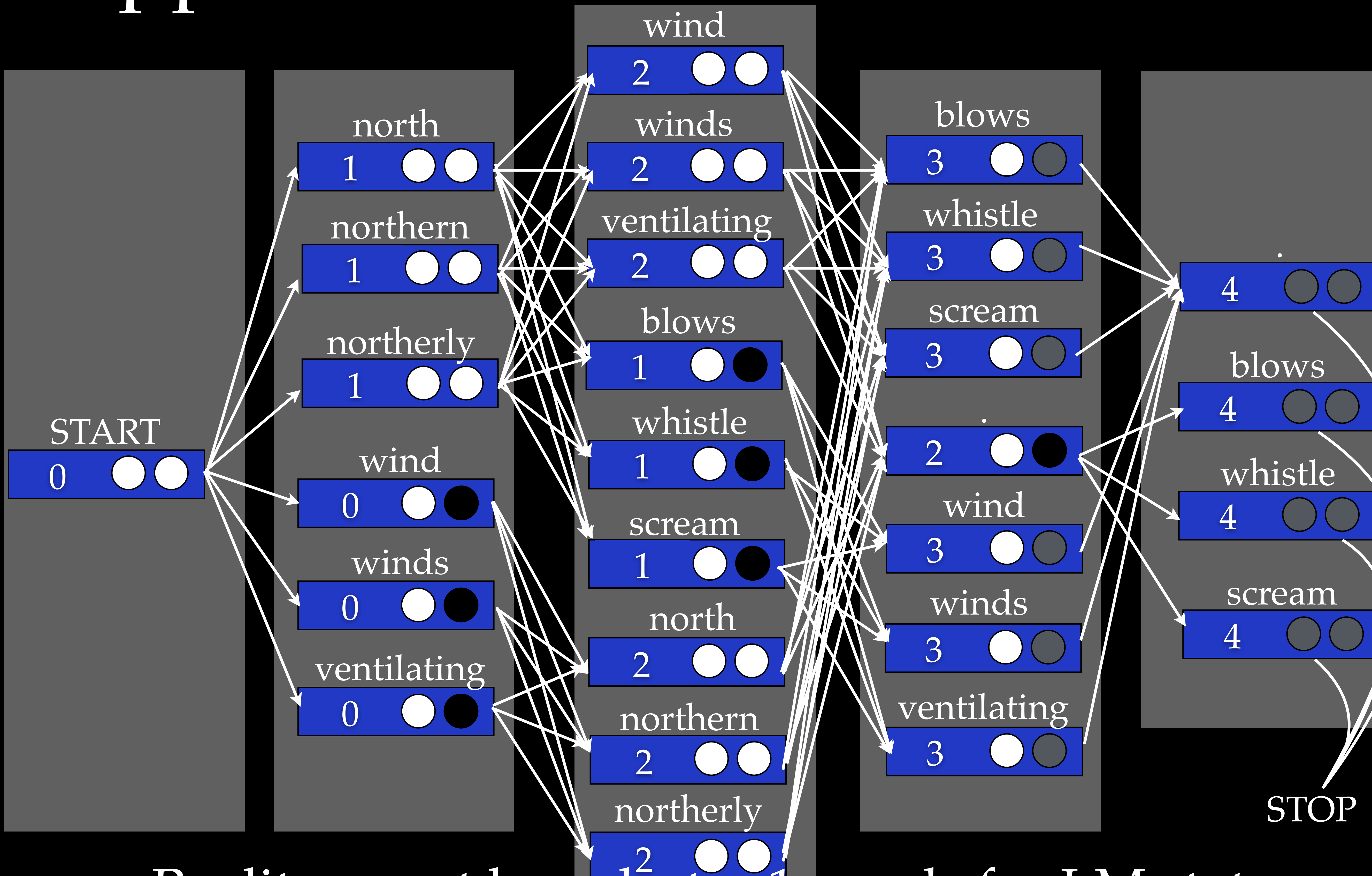
distortion limit = 2

Approximation: distortion limits



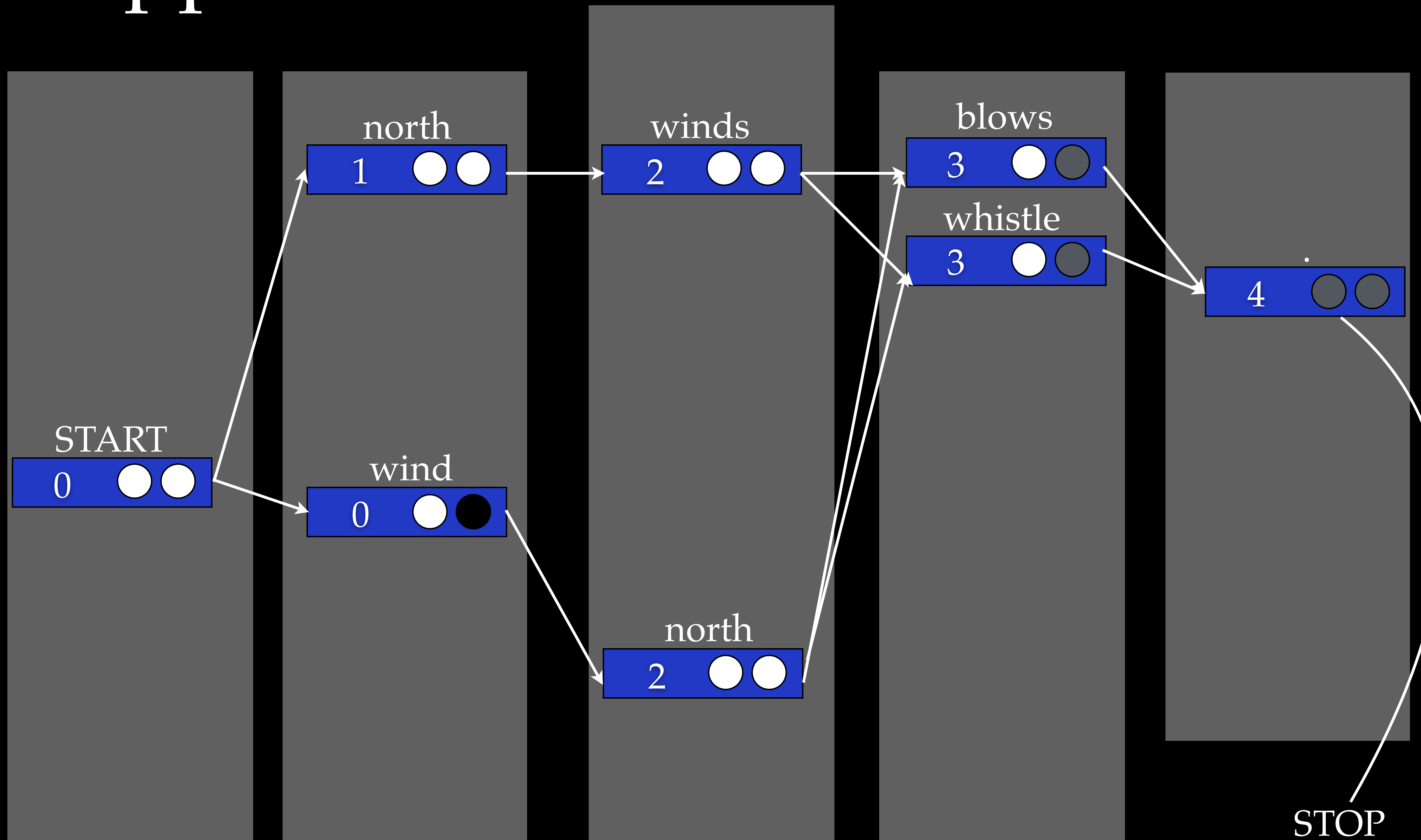
Reality: must keep last n-1 words for LM state

Approximation: distortion limits



Reality: must keep last n-1 words for LM state

Approximation: distortion limits



So, stack pruning is still useful

Putting it all together

```
place empty hypothesis into stack 0
for all stacks 0...n−1 do
  for all hypotheses in stack do
    for all translation options do
      if applicable then
        create new hypothesis
        place in stack
        recombine with existing hypothesis if possible
        prune stack if too big
      end if
    end for
  end for
end for
```

Things you need to determine:

What does a hypothesis (partial translation) look like?

What are all possible extensions of a hypothesis?

What is a reasonable equivalence class (stack)?