Edit Distance and Word Paths

By the end of this video you will be able to…

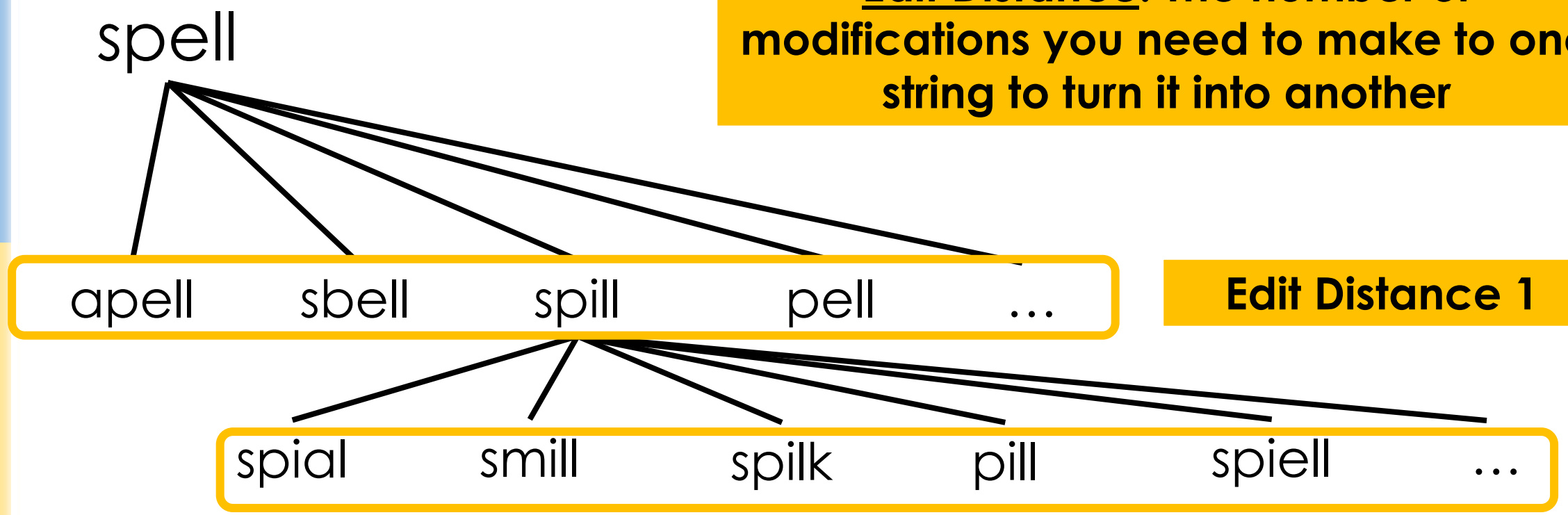
- Define the notion of edit distance
- Describe a (naïve) algorithm for calculating the edit distance between two strings
- Describe how pruning can be used to reduce the problem space

spell … mine

What path of words takes us from "spell" to "mine"?

spell → spill → pill → pile → pine → mine

Edit Distance: The number of modifications you need to make to one string to turn it into another

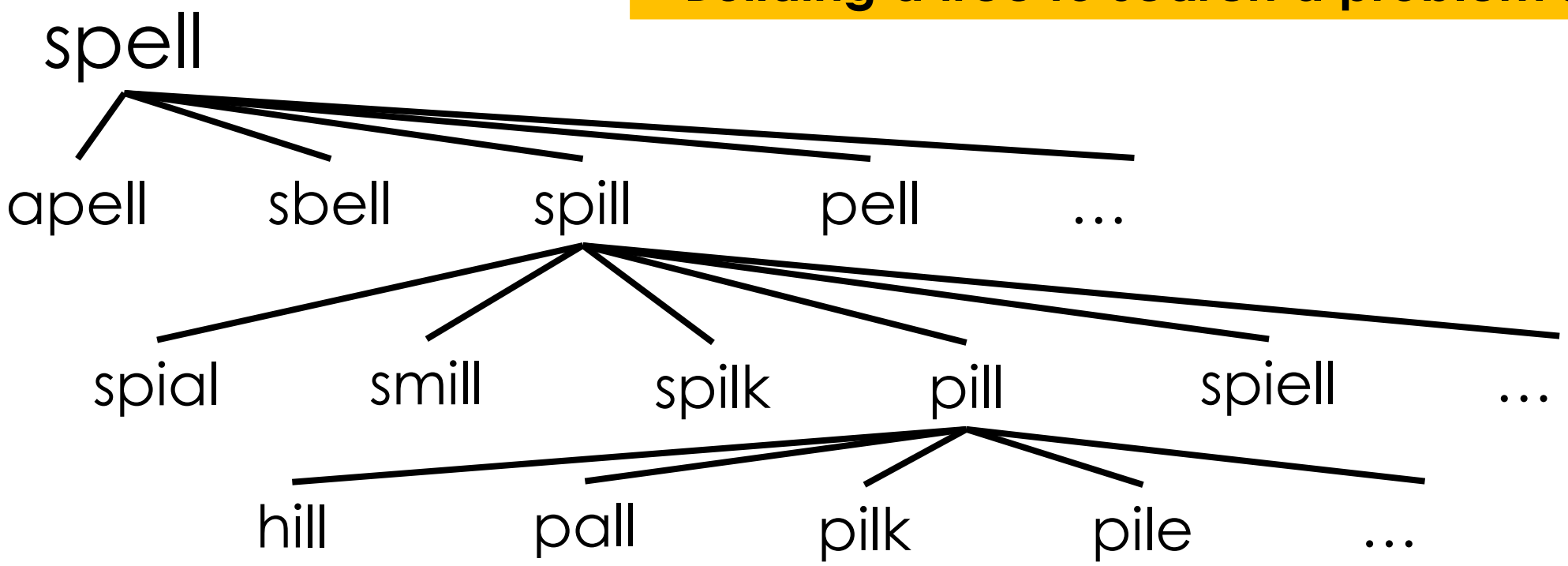


Edit Distance 1

Edit Distance 2

Other applications: Computational Biology and Natural Language Processing

Building a tree to search a problem space



Building a tree to search a problem space

How many strings are "1 away" from an initial word, where k is the length of the word?

$$25k + 26(k+1) + k$$

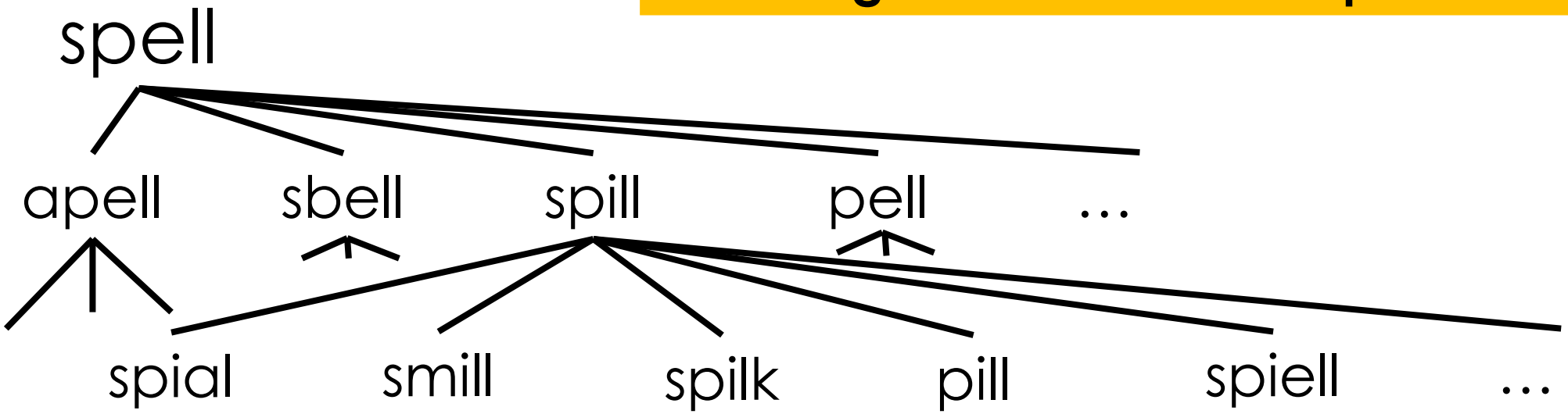
- $25k$: **Substitutions**
- $26(k+1)$: **Insertions**
- k: **Deletions**

Building a tree to search a problem space

How many strings are "1 away" from an initial word, where k is the length of the word?

52k+26

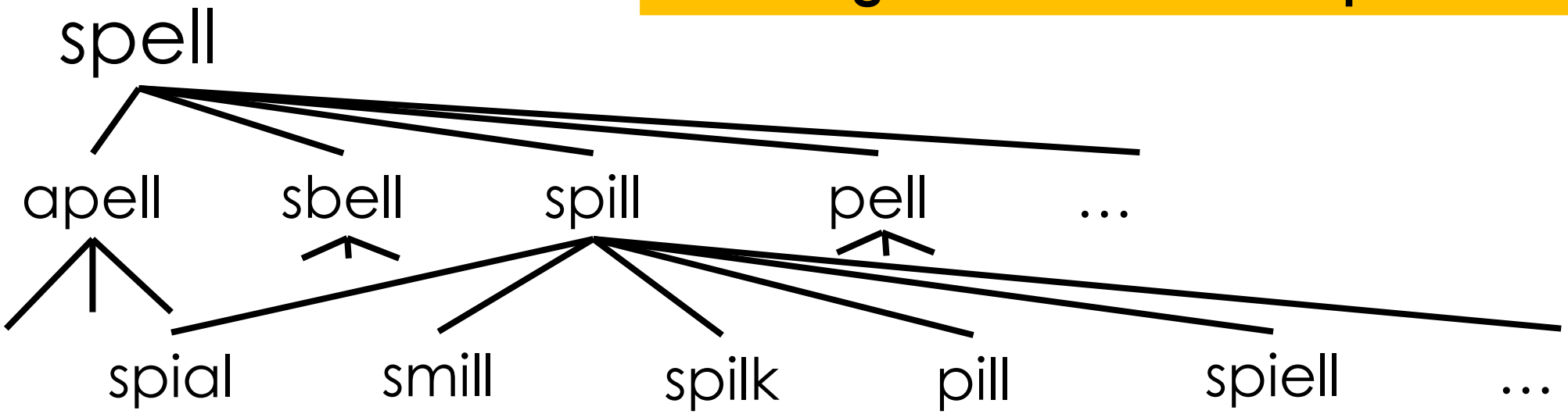
Building a tree to search a problem space



How many strings are "2 away" from an initial word, where k is the length of the word?

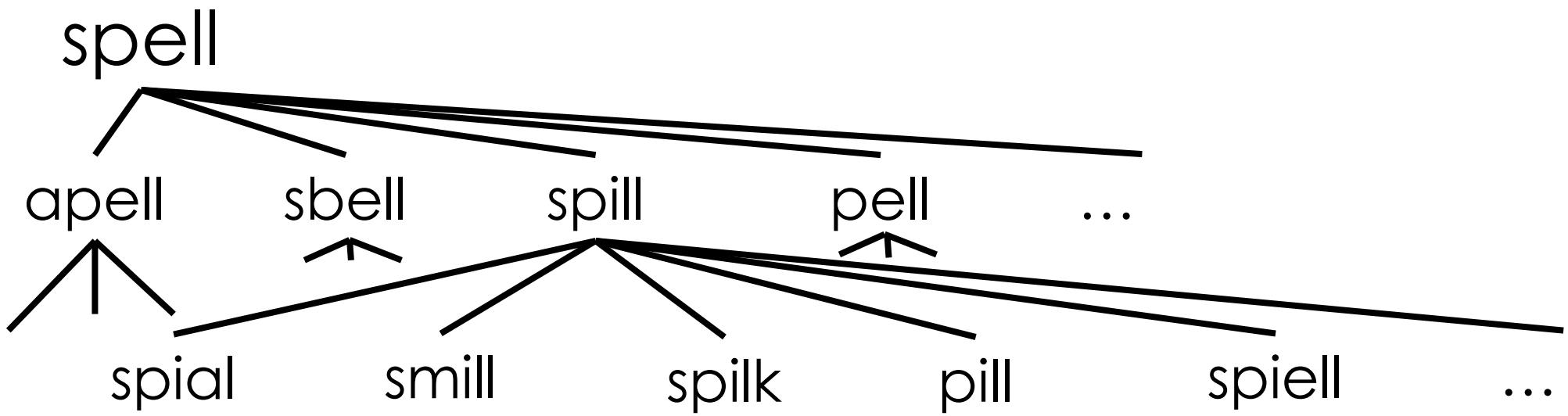
> $(52k+26)^2$

Building a tree to search a problem space



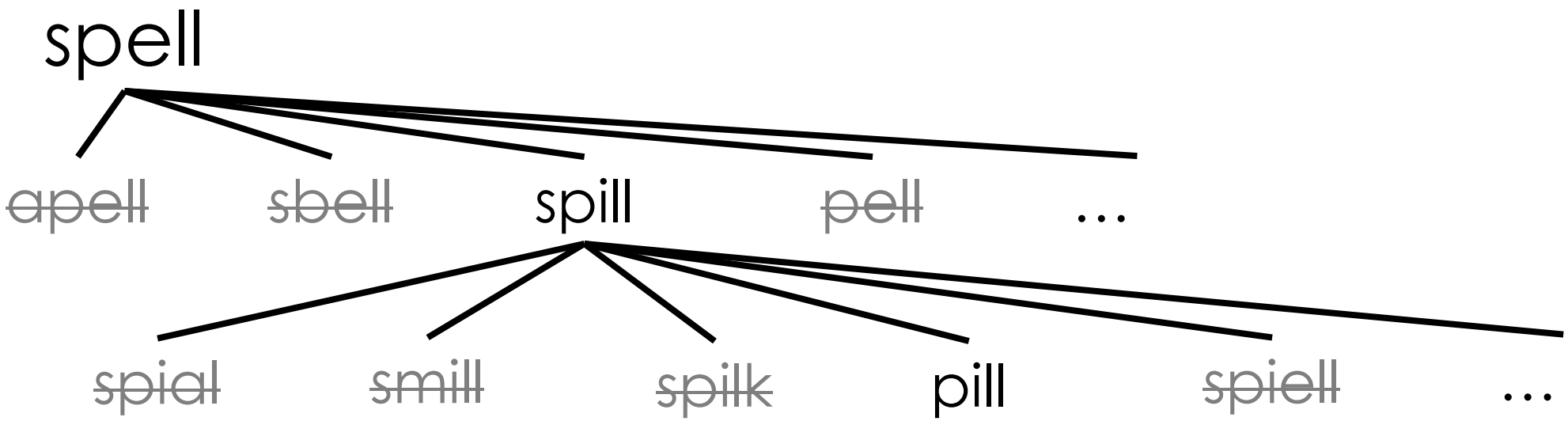
How many strings are "n away" from an initial word, where k is the length of the word?

$> (52k+26)^n$



Two possible solutions:

1. **Dynamic programming → $O(k^2)$**
2. **Pruning: Restrict the path to only valid words**



Two possible solutions:

1. **Dynamic programming → $O(k^2)$**
2. **Pruning: Restrict the path to only valid words**