

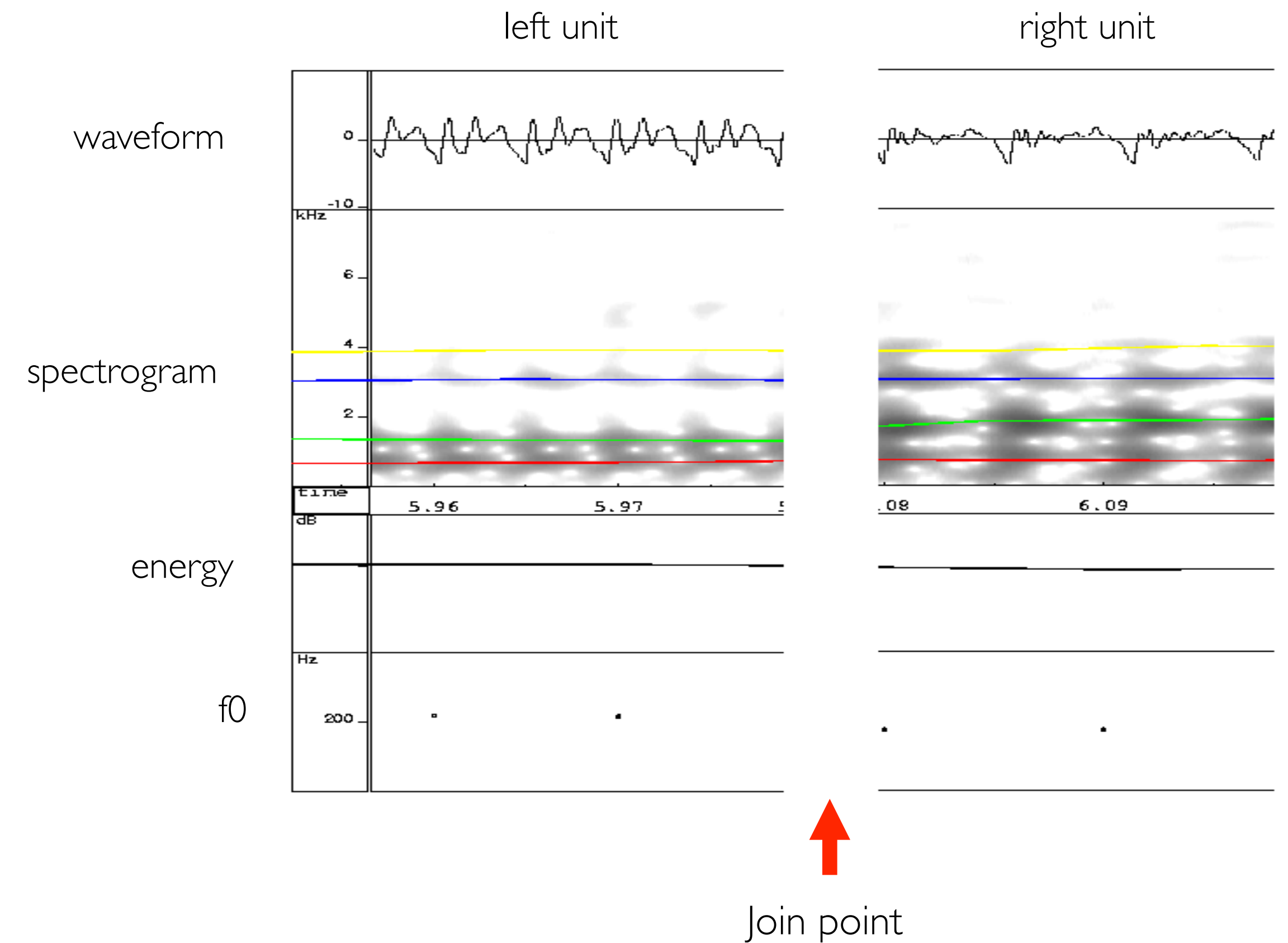
# Unit selection

---

- discussion points

# Join cost

- What acoustic features should we use?
- Any other improvements you can think of?
- What knowledge could you inject into the cost function?
  - e.g., phonetic



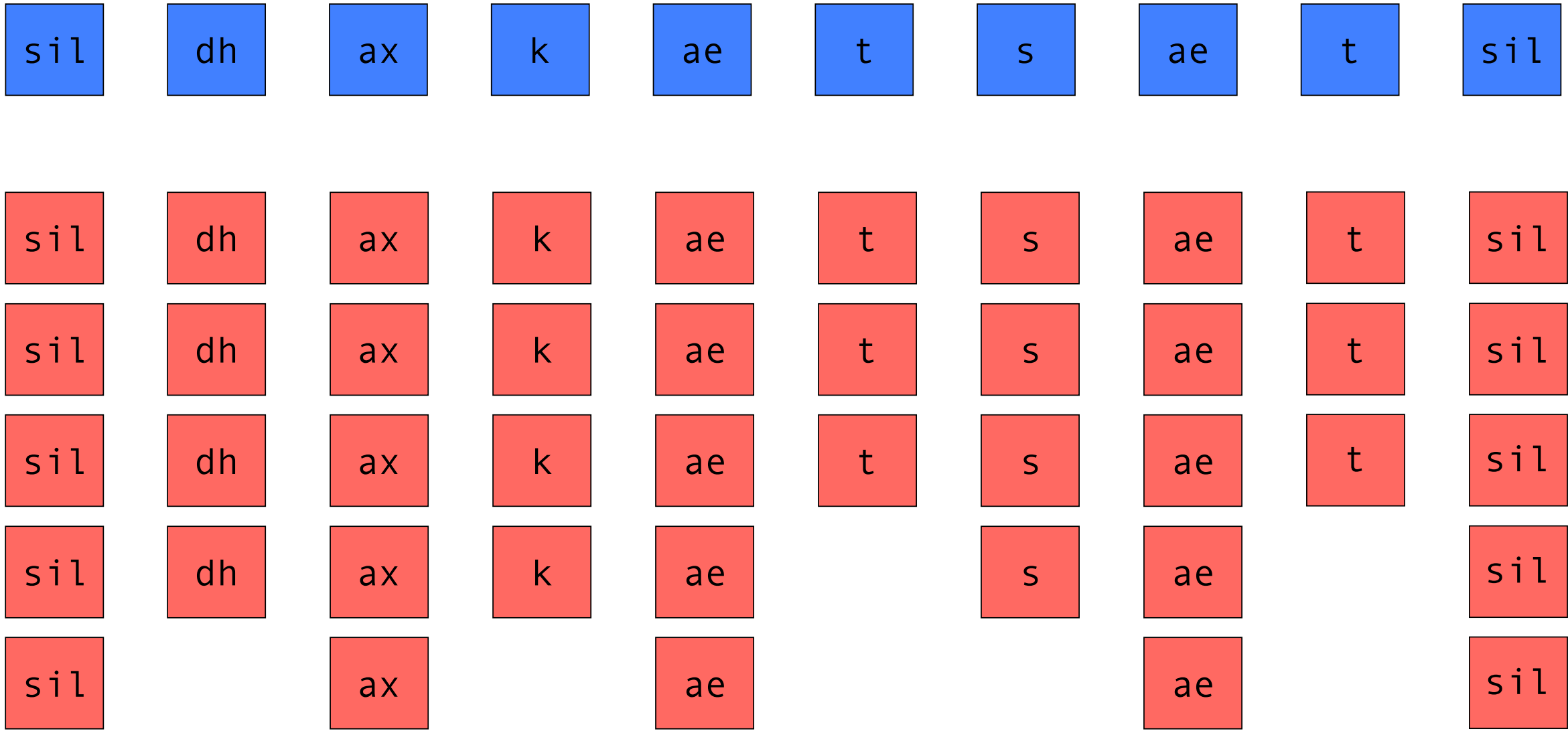
# Join cost

---

- Which of these would be a better place to make a join, and why?
- vowel
- diphthong
- stop
- approximant
- unvoiced fricative

# Can you make the search *faster*?

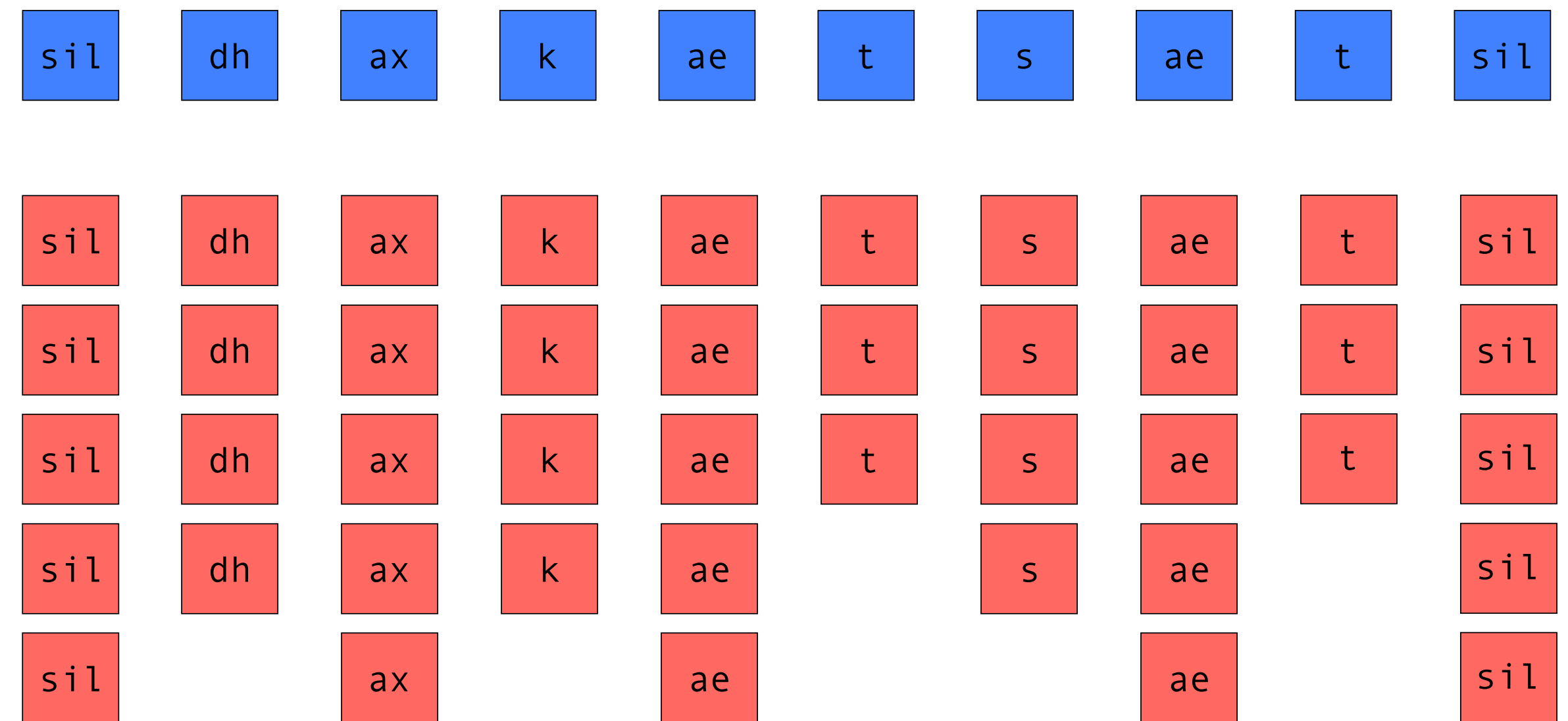
- Main computations are
  - target cost
  - join cost
- How many of each are computed per sentence being synthesised?



# Can you make the search ***faster***?

---

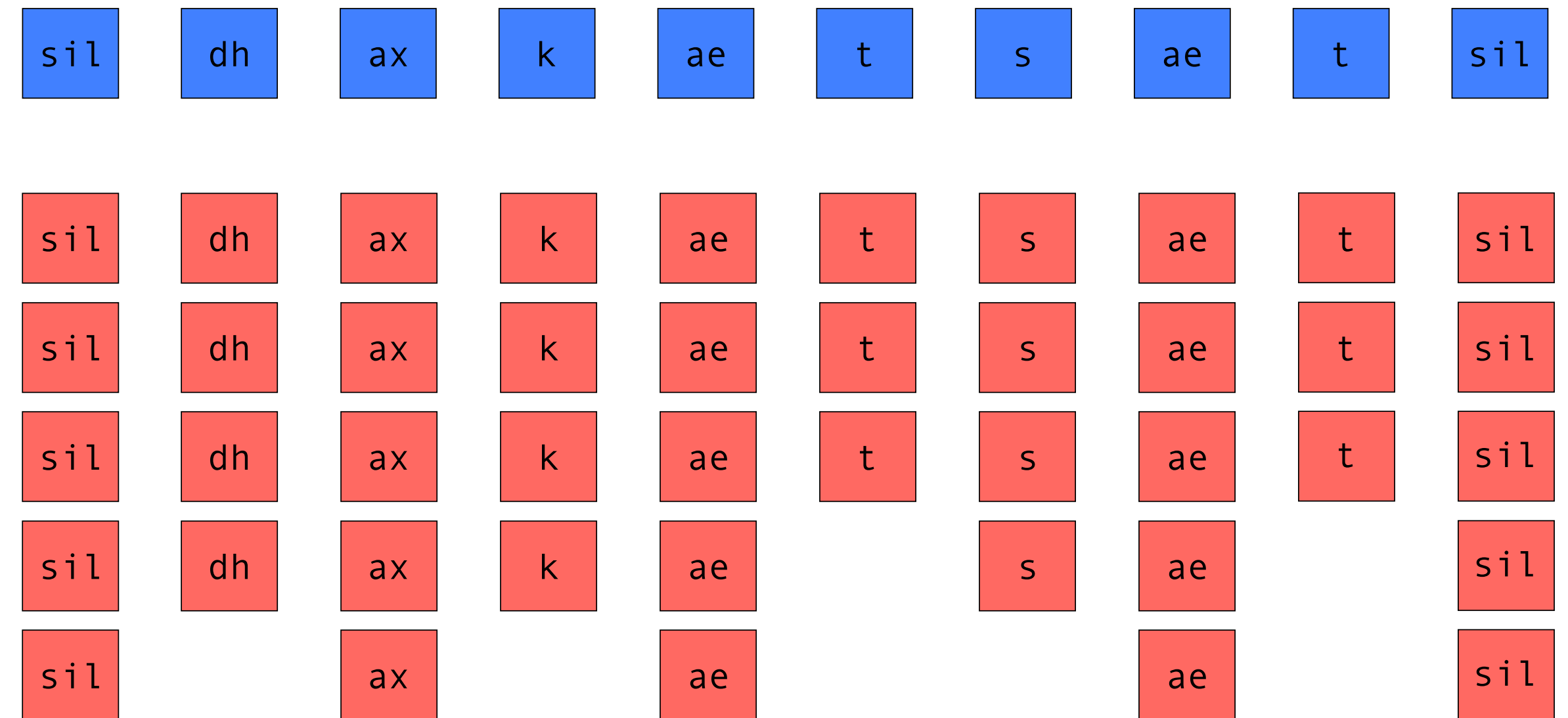
- What *dimensions* of the problem does computation time depend on?
- exactly **how** does compute time depend on them (linear? worse than linear?)
- Can you **reduce** those dimensions?
- What will the effect be on **quality**?



# Can you make the search ***faster***?

---

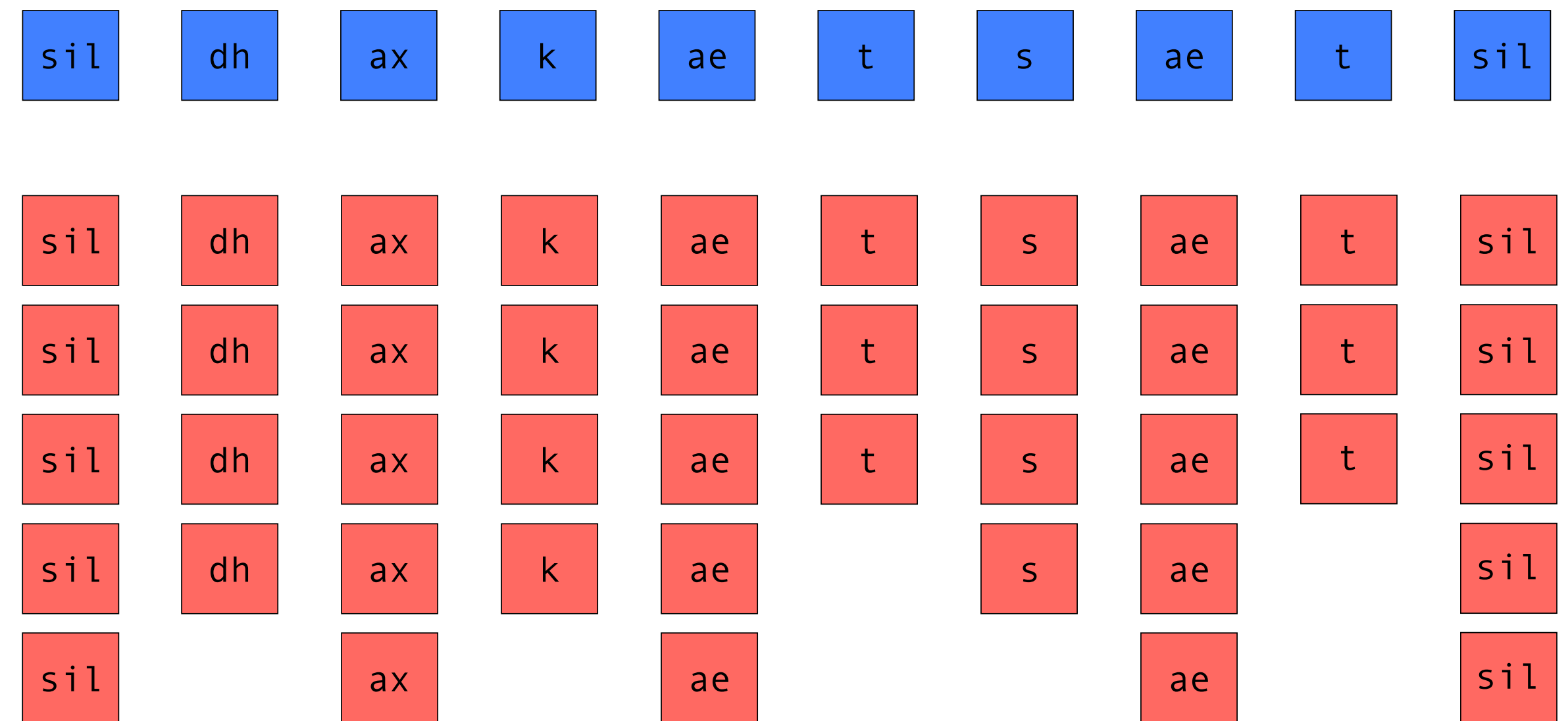
- Can we **cache** the target and/or join costs?
- i.e., save them, then re-use later for a different sentence, *without* recomputing



# “Looking forward and backwards”

---

- The search proceeds left-to-right
- Does this mean that the cost functions only “look” in one direction?



# Relative importance: target vs. join

---

- The overall cost is a sum of target and join costs
- We can weight one relative to the other
- What would be the effects of
  - higher weighting on the **target** cost
  - higher weighting on the **join** cost

