Approximate seeds, seeds with errors

contiguous seed

spaced seed

seed with errors
Seeds with errors: Levenshtein distance

- alphabet $\Sigma$
- two words over $\Sigma$
- three edit operations

```
A B L  A - L  A B L
A L L  A B L  A - L
```

substitution  insertion  deletion

- Levenshtein distance: smallest number of operations needed to transform one word into another

```
B A L L A D -
B A L L A D -
S A L - A D S
```
Levenshtein automaton

$P=SALAD$ and $k = 2$
Bit vector representation

\[ P = \text{BALLAD}, \ V = \text{SALAD}, \ k = 2 \]

- pattern \( P \rightarrow \ $ \)
- word \( V \rightarrow \ V \ $ \)
- sequence of \( |P| + k \) bit vectors of length \( 2k + 1 \)
Bit vector representation

- $P=$BALLAD, $V=$SALAD, $k = 2$
- pattern $P \rightarrow S^k P S^{2k}$
- word $V \rightarrow V S^{|P| - |V| + k}$
- sequence of $|P| + k$ bit vectors of length $2k + 1$
Nondeterministic Universal Levenshtein Automaton

The diagram shows a matrix with columns labeled B, A, L, L, A, D and rows labeled S, A, L, A, D, S. The matrix contains numbers that represent the Levenshtein distance between the characters in the strings. For example, the distance between 'B' and 'A' is 0, between 'A' and 'L' is 1, and so on.
Nondeterministic Universal Levenshtein Automaton

<table>
<thead>
<tr>
<th></th>
<th>B</th>
<th>A</th>
<th>L</th>
<th>L</th>
<th>A</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **substitution**
- **deletion (1 del + id)**
- **identity**
- **insertion**
Nondeterministic Universal Levenshtein Automaton

state \((x, y) : \text{"I am in the lane } y \text{ and have made } x \text{ errors so far"}\)
Nondeterministic Universal Levenshtein Automaton

\[ \text{NULA}(2) \ (k = 2) \]
Deterministic Universal Levenshtein Automaton

DULA(1) \((k = 1)\)
Seeds with error: a new type of seeds
Seeds with error: a new type of seeds

3 errors within the seed

AUCAGUGCAAAUGCUCAAGA

"selectivity": 1/256

5 parts of length 4
2 parts out of 5

AUCA GUGC AAAU GCUC AAGA
Seeds with error: a new type of seeds

3 errors within the seed

AUCAGUGCAAUAUGCUCAAGA

AUCAG UGCAA AUGCU CAAGA

4 parts of length 5
1 part out of 4
"selectivity": 1/256
Seeds with error: a new type of seeds

3 errors within the seed

AUCAGUGCAAAUGCUCAAGA

AUCAG UGCAA AUGCU CAAGA

4 parts of length 5
1 part out of 4
“selectivity” : 1/256

AUCA GUGC AAAU GCUC AAGA

5 parts of length 4
Seeds with error: a new type of seeds

3 errors within the seed

AUCAGUGCAAAUUGCUCACAAGA

AUCAG UGCAA AUGCU CAAGA

AUCA GUGC AAAU GCUC AAGA

4 parts of length 5
1 part out of 4
"selectivity" : 1/256

5 parts of length 4
2 parts out of 5
"selectivity" : 1/2100
<table>
<thead>
<tr>
<th>AUCA</th>
<th>GUGC</th>
<th>AAAU</th>
<th>GCUC</th>
<th>AAGA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUCA</td>
<td>AUGC</td>
<td>A-AU</td>
<td>GCGC</td>
<td>AAGA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AUCA</th>
<th>GUGC</th>
<th>AAAU</th>
<th>GCUC</th>
<th>-AAGA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUC-</td>
<td>GUGC</td>
<td>AUAAU</td>
<td>GCUC</td>
<td>AAAGA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AUCA</th>
<th>GUGC</th>
<th>AAAU</th>
<th>GCUC</th>
<th>AAGA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AUCA</td>
<td>GAGA</td>
<td>AAAU</td>
<td>GC-C</td>
<td>AAGA</td>
</tr>
</tbody>
</table>
01*0 seeds

- A 01*0 seed is a pair of exact parts surrounding 0 or more parts with exactly 1 error.

- Given a pattern partitioned in at least $k + 2$ parts, then any occurrence with at most $k$ errors contains such a 01*0 seed.
01*0 seeds – filtration efficiency

number of seed occurrences per pattern

100 patterns of length 20, text of length $10^8$, up to 3 errors