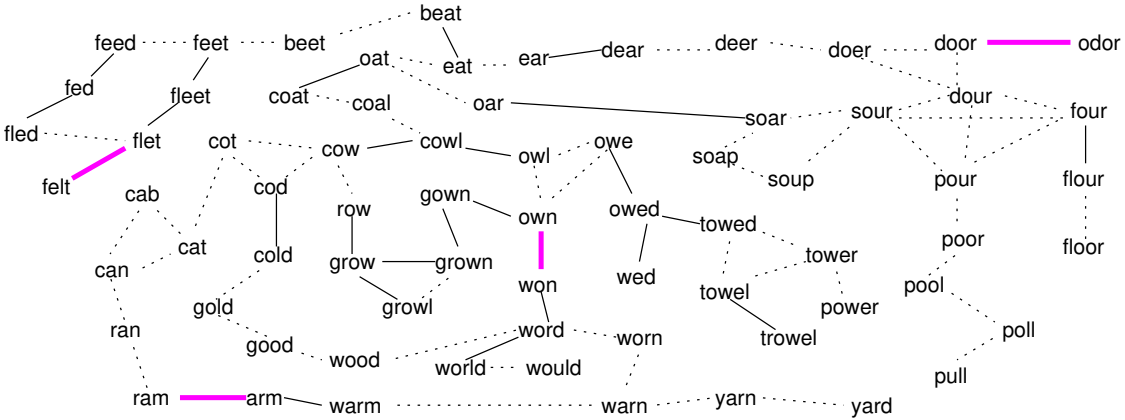
Edit distance and information theory units:

- The edit distance problema nd algorithm (**today**)
- A quick tour of information theory (next week Wednesday)
- Lab: Autoregressive text generation (next week Friday)

Today:

- Follow-up on regex chatbot
- The idea of edit distance
- The minimum edit distance problem
- The minumum edit distance algorithm

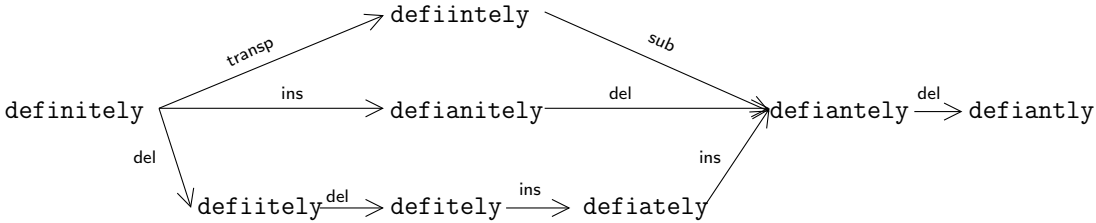

beat
feed
feet
beet
oat
eat
ear
dear
deer
doer
door
odor
fed
fleet
coat
coal
oar
soar
sour
dour
four
fled
flet
cot
cow
cowl
owl
owe
soap
soup
pour
flour
felt
cab
cod
row
gown
own
owed
towed
poor
cat
cold
grow
grown
won
wed
tower
floor
can
gold
growl
towel
power
pool
ran
good
word
worn
trowel
poll
wood
world
would
pull
ram
arm
warm
warn
yarn
yard

Versions of the minimum edit distance:

- Substitutions only: *Hamming distance*. (Richard Hamming, 1950)
- Insertions and deletions: *Longest common subsequence*.
- Substitutions, insertions, and deletions: *Levenshtein distance.* (Vladimir Levenshtein, 1966)
- Substitutions, insertions, deletions, and transpositions: *Damerau-Levenshtein distance.* (Fred Damerau, 1964)

$$\texttt{recieve} \xrightarrow{\text{del}} \texttt{receve} \xrightarrow{\text{ins}} \texttt{receive} \quad \text{versus} \quad \texttt{recieve} \xrightarrow{\text{transp}} \texttt{receive}$$

$$\texttt{seperate} \xrightarrow{\text{del}} \texttt{seprate} \xrightarrow{\text{ins}} \texttt{separate} \quad \text{versus} \quad \texttt{seperate} \xrightarrow{\text{sub}} \texttt{separate}$$


definitely
transp
defiintely
sub
ins
defianitely
del
defiantely
del
defiantly
del
defiitely
del
defitely
ins
defiately
ins

6	craven	craven	craven	craven	craven	craven	craven	craven
	␣	c	ca	car	carv	carvi	carvin	carving
5	crave	crave	crave	crave	crave	crave	crave	crave
	␣	c	ca	car	carv	carvi	carvin	carving
4	crav	crav	crav	crav	crav	crav	crav	crav
	␣	c	ca	car	carv	carvi	carvin	carving
3	cra	cra	cra	cra	cra	cra	cra	cra
	␣	c	ca	car	carv	carvi	carvin	carving
2	cr	cr	cr	cr	cr	cr	cr	cr
	␣	c	ca	car	carv	carvi	carvin	carving
1	c	c	c	c	c	c	c	c
	␣	c	ca	car	carv	carvi	carvin	carving
0	␣	␣	␣	␣	␣	␣	␣	␣
	␣	c	ca	car	carv	carvi	carvin	carving
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**

$$
D[i][j] = \begin{cases}
0 & \text{if } i = j = 0 & \text{(Empty prefixes: do nothing)} \\
j \cdot C[0] & \text{if } i = 0 \text{ and } j > 0 & \text{(Empty prefix of } a\text{: insert all the } b \text{ prefix)} \\
i \cdot C[1] & \text{if } i > 0 \text{ and } j = 0 & \text{(Empty prefix of } b\text{: delete all the } a \text{ prefix)} \\
\min \begin{pmatrix}
C[0] + D[i-1][j] & & \text{(insertion)} \\
C[1] + D[i][j-1] & & \text{(deletion)} \\
C[2] + D[i-1][j-1] & & \text{(substitution)} \\
C[3] + D[i-2][j-2] & \text{if } a[i-1] = b[j-2] \text{ and } a[i-2] = b[j-1] & \text{(transposition)} \\
D[i-1][j-1] & \text{if } a[i-1] = b[j-1] & \text{(nop)}
\end{pmatrix} & \text{Otherwise}
\end{cases}
$$

n	6/ins-all	5/ins	4/ins	4/ins	3/ins	3/ins	2/nop	3/del
e	5/ins-all	4/ins	3/ins	3/ins	2/ins	2/sub	3/del	4/del
v	4/ins-all	3/ins	2/ins	2/ins	1/nop	2/del	3/del	4/del
a	3/ins-all	2/ins	1/nop	1/transp	2/del	3/del	4/del	5/del
r	2/ins-all	1/ins	1/sub	1/nop	2/del	3/del	4/del	5/del
c	1/ins-all	0/nop	1/del	2/del	3/del	4/del	5/del	6/del
	0/del-all	1/del-all	2/del-all	3/del-all	4/del-all	5/del-all	6/del-all	7/del-all
		c	a	r	v	i	n	g

c	a	r	v	i	n	g
nop	transp	nop	sub	nop	del	
c	r	a	v	e	n	

Coming up:

- Reading from J&M, Section 2.5 (Fri, Sept 1)
- Regular expressions assignment (Fri, Sept 1)

- Edit distance quiz (Tues, Sept 5)
- Edit distance assignment (Fri, Sept 8)

- Reading from Stone (see Canvas) (Wed, Sept 6)

Next time: A quick tour of information theory