Today's Topics

• Homework 8 review
• More on FSA
Homework 8: Part 1 Solution

```perl
# Perl code snippet

$be = qr/berami'llis'slarel'relwasi'werelbeingbeen/;  
$have = qr/have/haveldhad/;  
$neg = qr/(n'tinot)(\s+yet)/;  
$passive = qr/\b($be)\s+w+en\b/;  
$npassive = qr/\b($be)\s+($neg)\s+w+en\b/;  
$perfective = qr/\b($have)\s+w+en\b/;  
$npperfective = qr/\b($have)\s+($neg)\s+w+en\b/;  

open $fh, $ARGV[0] or die "Can't find $ARGV[0]!\n";

while (<$fh>) {
    while (/\$passive/g) {
        $count{'passive'}++
    }
    while (/\$perfective/g) {
        $count{'perfective'}++
    }
    while (/\$npassive/g) {
        $count{'negated passive'}++
    }
    while (/\$npperfective/g) {
        $count{'negated perfective'}++
    }
}

@sorted_keys = sort { $count{$b} <=> $count{$a} } keys %count;
foreach $key (@sorted_keys) {
    print "$key: $count{$key}\n";
}
```

[sandiway]$ perl hw8p1.pl wsj.txt

perfective: 1911
passive: 345
negated perfective: 161
negated passive: 22

[sandiway]$ perl hw8p1a.pl wsj.txt

perfective: 1817
passive: 340
negated perfective: 149
negated passive: 21

without g
Homework 8: Part 2 Solution

• One liner:
  – perl -ne '/	([\w\s]+)\s*$/' and print join(" 
",split("/", $1)), "\n" irregular_verbs.txt

• Program:

```perl
1open $fh, $ARGV[0] or die "Can't find $ARGV[0]!
";  
2while ($line = <$fh>) {
3  if ($line =~ \t([\w\s]+)\s*$/) {
4    print join(" 
", split("/", $1)), "\n";  
5  }
6}
```

arisen
awoken
been
borne
beaten
become
begun
bent
bet
bound
bitten
bled
blown
broken
bred
brought
broadcast
built
burnt
burned
Homework 8: Part 3 Solution

```plaintext
$be = qr/\b(lam|\llis|\lare|\lrel|\lrel|\lwas|\lre|\lbe|\leen)/;  
$have = qr/\b(\hav|\has|\had)/;  
$neg = qr/\b(\n't|\lnot)(\s+\y\et)/;  
$passive = qr/\b($be)$s+\w+\en\b/;  
$npassive = qr/\b($be)$s+(\neg)$s+\w+\en\b/;  
$perfective = qr/\b($have)$s+\w+\en\b/;  
$nperfective = qr/\b($have)$s+(\neg)$s+\w+\en\b/;  

open $fh1, $ARGV[0] or die "Can't find $ARGV[0]!\n";  
@irr = ();  
while (<$fh1>) {  
    chomp;  
    if (\t(\w\w\+/)$s*$/) {  
        push @irr, split('\/', $s1)  
    }  
}  
$s = join('!', @irr);  

$irr_passive = qr/\b($be)$s+(\$s)$b/;  
$irr_npassive = qr/\b($be)$s+(\neg)$s+(\s)$b/;  
$irr_perfective = qr/\b($have)$s+(\$s)$b/;  
$irr_nperfective = qr/\b($have)$s+(\neg)$s+(\$s)$b/;  

open $fh2, $ARGV[1] or die "Can't find $ARGV[1]!\n";  
while (<$fh2>) {  
    while (/\$passive/) {  
        $count{'passive'}++;  
    }  
    while (/\$irr_passive/) {  
        $count{'irregular passive'}++;  
    }  
    while (/\$perfective/) {  
        $count{'perfective'}++;  
    }  
    while (/\$irr_perfective/) {  
        $count{'irregular perfective'}++;  
    }  
    while (/\$npassive/) {  
        $count{'negated passive'}++;  
    }  
    while (/\$irr_npassive/) {  
        $count{'irregular negated passive'}++;  
    }  
    while (/\$nperfective/) {  
        $count{'negated perfective'}++;  
    }  
    while (/\$irr_nperfective/) {  
        $count{'irregular negated perfective'}++;  
    }  

S1 foreach @key (sort {$count{$b} <=> $count{$a}} keys $count) {  
    print "$key: $count{$key}\n";  
}  
```

What about –ed past participles?
Homework 8: Part 3 Solution

- they were all sentenced

Pretoria has n't forgotten why they were all sentenced to life imprisonment in the first place : for sabotage and conspiracy to overthrow the government .

In addition , the government is figuring that the releases could create a split between the internal and external wings of the ANC and between the newly freed leaders and those activists who have emerged as leaders inside the country during their imprisonment .

In order to head off any divisions , Mr. Mandela , in a meeting with his colleagues before they were released , instructed them to report to the ANC headquarters in Lusaka as soon as possible .

The men also will be faced with bridging the generation gap between themselves and the country 's many militant black youths , the so-called young lions who are anxious to see the old lions in action .

Says Peter Mokaba , president of the South African Youth Congress : `` We will be expecting them to act like leaders of the ANC . ''

They never considered themselves to be anything else .

At last night 's rally , they called on their followers to be firm , yet disciplined , in their opposition to apartheid .

`` We emphasize discipline because we know that the government is very , very sensitive , '' said Andrew Mlangeni , another early Umkhonto leader who is now 63 .

`` We want to see Nelson Mandela and all our comrades out of prison , and if we are n't disciplined we may not see them here with us .''
What's the real number of passives in the corpus?

- The WSJ corpus has labeled trees with POS tags and phrasal tags. A serious search reveals about **8050** passive constructions.
Back to FSAs...
$\varepsilon$-transitions

- jump from state to another state with the empty character
  - $\varepsilon$-transition (*textbook*) or $\lambda$-transition
  - no increase in expressive power

- examples

what’s the equivalent without the $\varepsilon$-transition?
Non-Deterministic Finite State Automata (NDFSA)

- non-deterministic FSA (NDFSA)
- no restriction on ambiguity (surprisingly, no increase in power)
- Example:
Non-Deterministic Finite State Automata (NDFSA)

Possible strategies for keeping track of multiple states:
1. Backtracking (backup)
2. Lookahead
3. Parallelism

algorithm gets complicated fast

```
function ND-RECOGNIZE(tape, machine) returns accept or reject
    agenda ← {(Initial state of machine, beginning of tape)}
    current-search-state ← NEXT(agenda)
    loop
        if ACCEPT-STATE?(current-search-state) returns true then
            return accept
        else
            agenda ← agenda ∪ GENERATE-NEW-STATES(current-search-state)
            if agenda is empty then
                return reject
            else
                current-search-state ← NEXT(agenda)
        end
    function GENERATE-NEW-STATES(current-state) returns a set of search-states
        current-node ← the node the current search-state is in
        index ← the point on the tape the current search-state is looking at
        return a list of search states from transition table as follows:
        (transition-table[current-node, index])
        ∪ (transition-table[current-node, tape[index]], index + 1)
    function ACCEPT-STATE?(search-state) returns true or false
        current-node ← the node search-state is in
        index ← the point on the tape search-state is looking at
        if index is at the end of the tape and current-node is an accept state of machine
        then
            return true
        else
            return false
```

Figure 2.19 An algorithm for NFA recognition. The word node means a state of the NFA, and state or search-state means “the state of the search process”, i.e., a combination of node and tape position.
NDFSA → (D)FSA

[discussed at the end of section 2.2 in the textbook]

• construct a new machine
  – each state of the new machine represents the set of possible states of the original machine when stepping through the input

• Note:
  – new machine is equivalent to old one (but has more states)
  – new machine is deterministic

• example
Ungraded Homework 9

• do not submit, but do the following exercise to check your understanding
  – apply the set-of-states construction technique to the two machines on the ε–transition slide (repeated below)
  – self-check your answer:
    • verify in each case that the machine produced is deterministic and accurately simulates its ε–transition counterpart