Collaborators:
    Faith Curtis
    Anu Korada
    Ben Faiman

Our current grammar:

S -> {NP, CP} TP
TP -> T {NegP, NP}
VP -> V (NP) (AP) (PP)*(CP)
VP -> V VP
V -> V Part
NegP -> Neg VP
NP -> {(PossP) (AP)* N (PP)* (CP), PN, Name}
AP -> (Deg) A (PP) (CP)
PP -> P NP
CP -> C S
PossP -> {NP Poss, D}
XP -> XP C XP

1. I sold a goat to the farmer
2. I sold the farmer a goat
3. We baked a cake for Sue's mother
4. We baked Sue's mother a cake

From the above sentences only 1 and 3 are currently generated by our grammar (their trees are shown below).
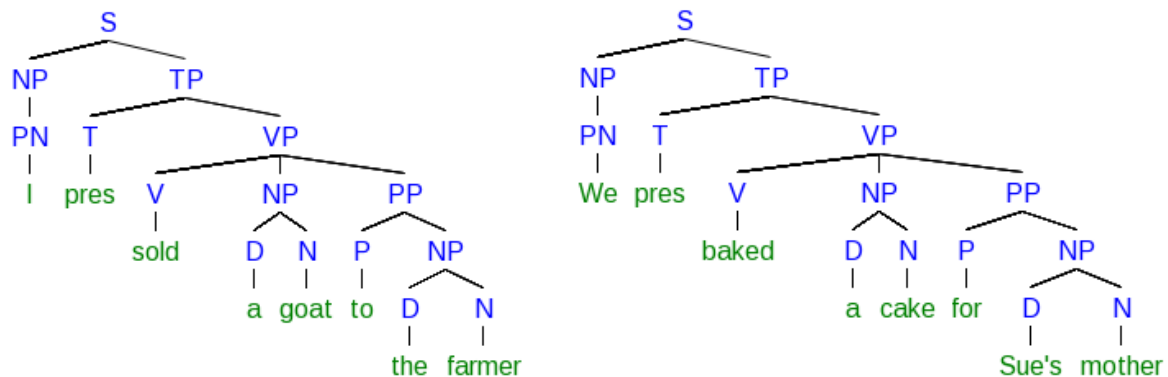
The phrase structure rules that we have right now don't allow for two NPs after a V. To make sentences like 2 and 4 possible we will have to either change our VPs structure and radically

change verb subcategorization in our lexicon or come up with a new transformation rule. I am going to try both approaches and then choose the one that appears to be more appropriate.

Without using transformations we are forced to build the NP NP structure directly into our VP rule. The VP rule is now:

VP -> V (NP) (NP) (AP)(PP)*(CP)

This will generate all kinds of ungrammatical sentences. Below are just a couple of examples:

* I put the cake the table
* The pig cooked the stove the cat
* The cat escorted the dog the jail

In order to make this anything close to a working proposal we are going to need a lot of subcategorization. We need verbs like baked, sold and bought to be able to take two NPs. Verbs like walked, rolled and escorted need to only allow for one NP. So for this to work we need to find all the verbs that can take two NPs and separate them from the verbs that can't. We then need to subcategorize each of these verbs in our lexicon. For exampe:

bake: V [_ NP (NP)]
sell: V [_ NP (NP)]
bought: V [_ NP (NP)]

and

walked: V [_ NP]
rolled: V [_ NP]
escorted: V [_ NP]

Using these verbs we can now generate:

- I baked Jim a cake
- The farmer baked the cake
- We sold the cat a ball
- The dog sold the cat
- I bought a dog
- The farmer bought the dog a bone

and
- I walked the dog
- I rolled the ball
- I escorted my aunt

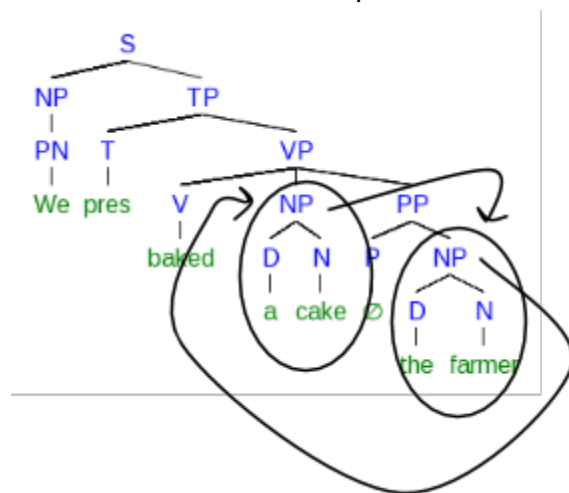That's is a lot of subcategorization that would need to be done.

If we can use transformations we can come up with a simpler solution. First we need a new verb category in our lexicon. Let's call it Dative Verb or DV for short. We want to be able to
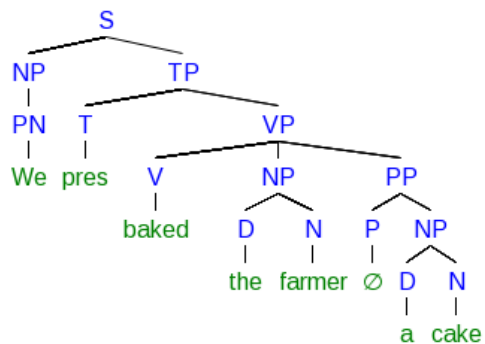
take a sentence with a deep structure that contains a V NP PP structure where the PP is a recipient (the PP must introduce the noun that is receiving the verbs action).

S
NP TP
PN T VP
We pres V NP PP
baked D N P NP
a cake for D N
the farmer
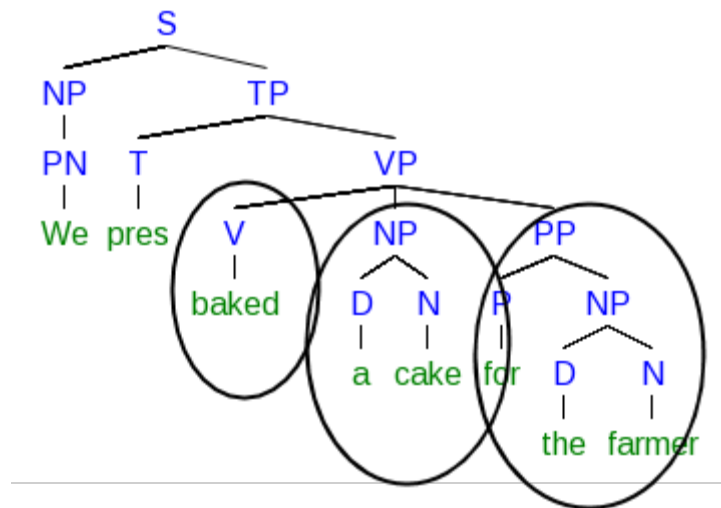
Then, to create the dative phrase we would eliminate the P from the PP and switch the NPs.

S
NP TP
PN T VP
We pres V NP PP
baked D N P NP
a cake Ø D N
the farmer

Leaving use with:

S
NP TP
PN T VP
We pres V NP PP
baked D N P NP
the farmer Ø D N
a cake

We can formalize the rule as follows:

Dative Rule

> SD: Take S with a DV (dative verb), NP and a PP
> SC: (1) Remove the P from the PP
>   (2) Switch the NP and the PP's NP

As mentioned before, this rule needs the lexicon to contain the set DV with all the verbs that can be used to form a dative sentence.

It seems obvious now that the proposal using transformations is preferable to the one that attempted to only alter the phrase structure and subcategorization. Using the transformation the only thing we need to change in the lexicon is adding DVs. The dative rule also allows us to keep the same deep structure for sentences like "I baked a cake for dad" and "I baked dad a cake". This seems neater, somehow, than having two different structures for sentences that mean exactly the same thing. We will, therefore, choose to adopt the dative rule.

> 1. I walked my mother to the store
> 2. * I walked the store my mother
> 3. I rolled the ball to the edge of the field
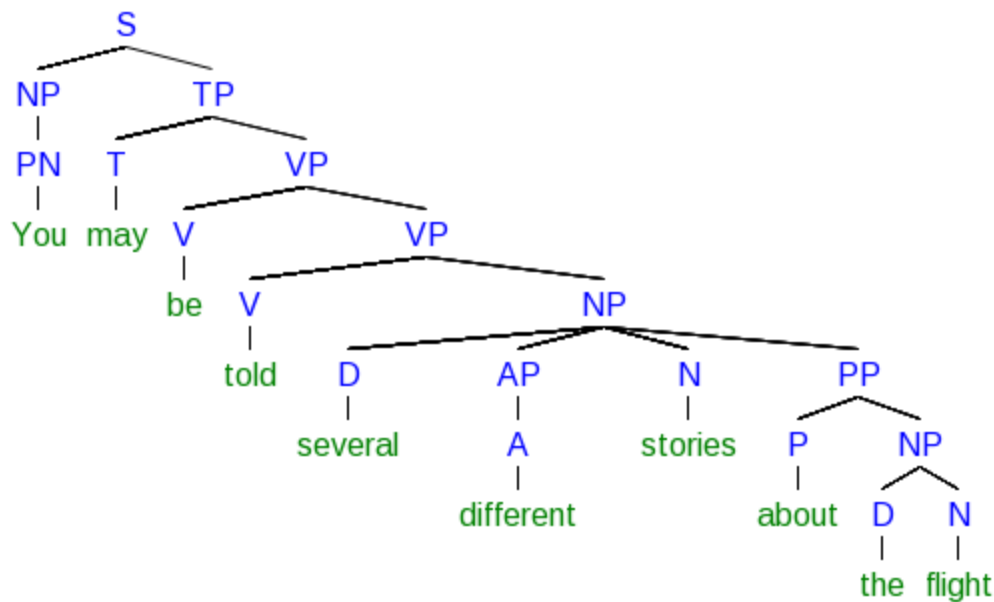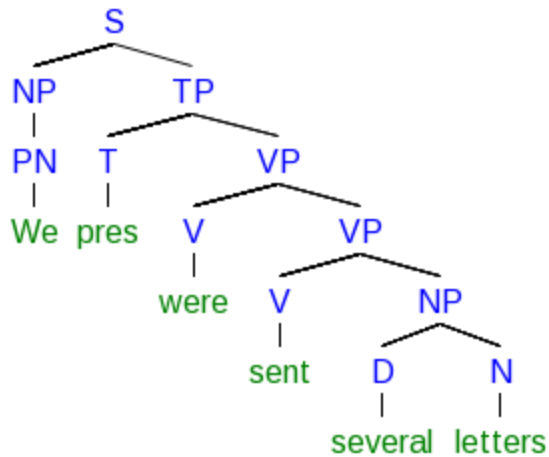> 4. * I rolled the edge of the field the ball

Sentences like 1-4 above are consistent with our new grammar. 1 is generated as before. 2 is not generated because walked is not a dative verb. 3 is generated as before. 4 is not generated because rolled is not a dative verb.

> 1. We baked it for Sally
> 2. * We baked Sally it
> 3. I sold it to the farmer
> 4. * I sold the farmer it

Sentences like 2 and 4 above present us with a problem. It appear that NPs that are PNs do not work in dative phrases. A possible solution for this would be to specifically exclude them in the dative rule's structural description.

> A. We were sent several letters
> B. You may be told several different stories about the fight
> C. I cooked my monkey up some mush

Our grammar does generate these. I am assuming that several is a determiner. If this is true then the trees below show how the sentences are constructed using a phrase structure rules.

```
              S
          ___/ \
        NP       TP
        |      _/  \___
        PN  T       VP
        |   |     _/  \___
        We pres  V        VP
                 |      __/  \___
                were   V         NP
                       |        / \
                      sent    D     N
                              |     |
                          several letters
```

```
              S
          ___/ \___
        NP          TP
        |        __/  \___
        PN   T        VP
        |    |     __/    \___
        You may   V           VP
                  |        ___/    \___
                 be   V                 NP
                      |      ___/  |  \___
                     told  D    AP    N      PP
                           |    |     |     /  \
                        several A   stories P    NP
                                |          |    / \
                            different    about D   N
                                               |   |
                                              the flight
```

C is generated using the first the Dative rule and then the Particle Movement rule.

Particle Movement Rule

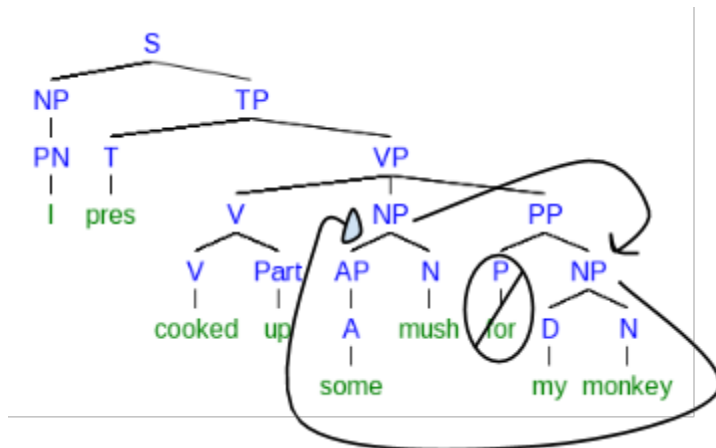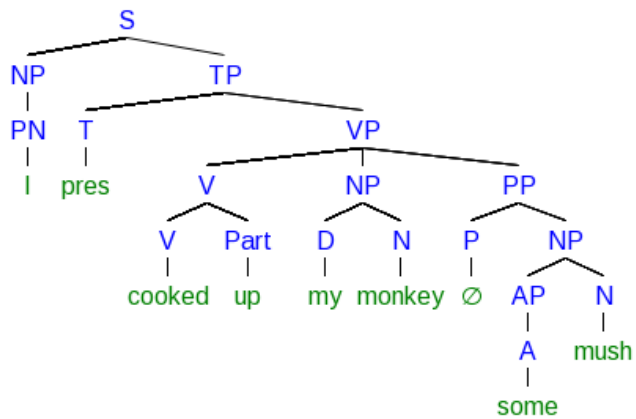| |
|---|
| SD: Take S that has a particle verb and a direct object NP
SC: Move the particle to the right of the NP |

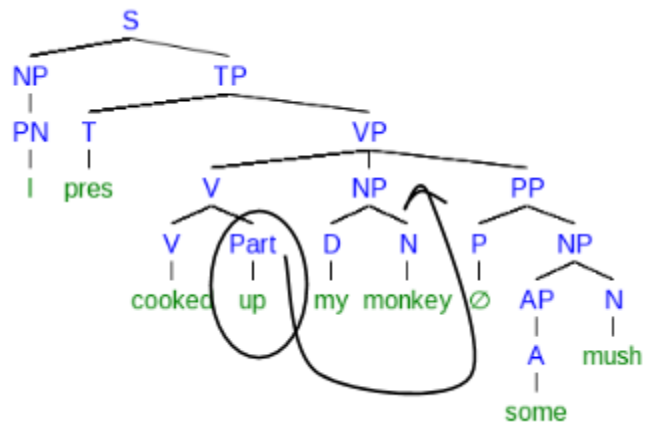Using C's base sentence, "I cooked up some mush for my monkey" we get
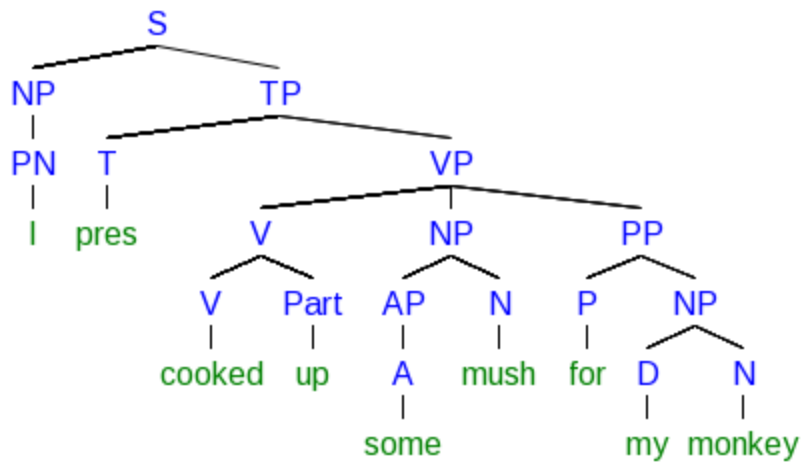
First we apply our new dative rule

This gives us

Then we apply the Particle rule

S
NP   TP
PN   T   VP
I   pres   V   NP   PP
V   Part   D   N   P   NP
cooked up   my monkey ∅   AP   N
A   mush
some

And get

S
NP   TP
PN   T   VP
I   pres   V   NP   Part   PP
V   Part D   N   up   P   NP
cooked ∅   my monkey   ∅   AP   N
A   mush
some