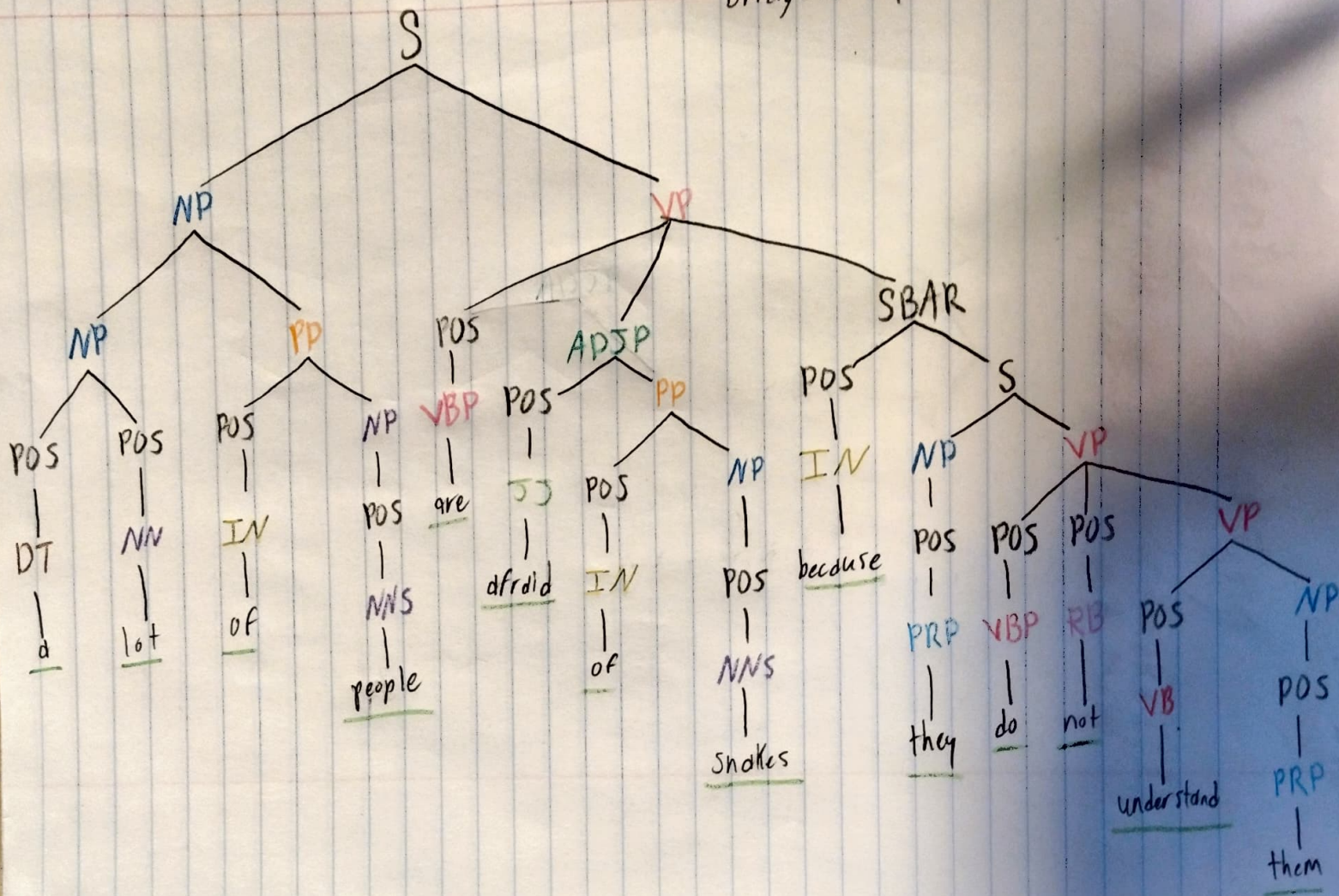


PSG Tree

Sentence Parsing

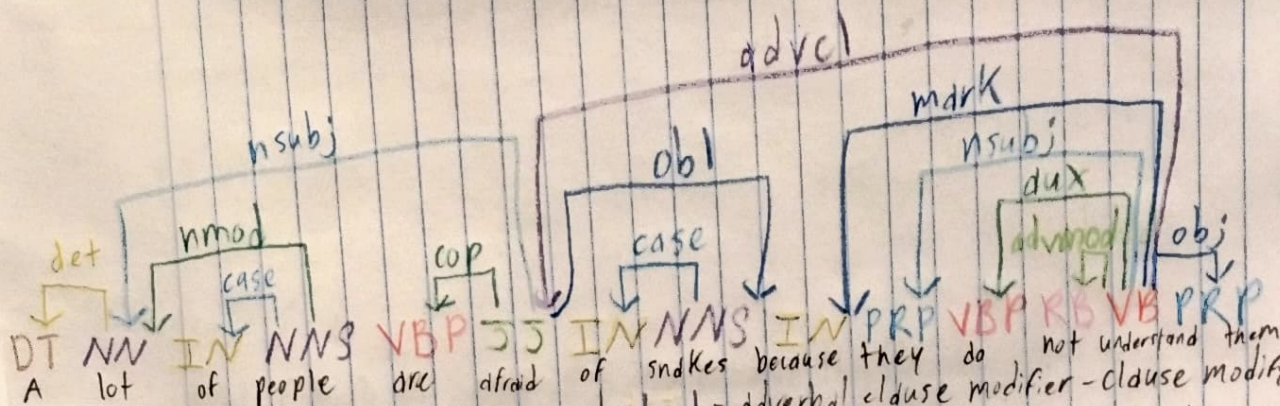
Bridgette Bryant BMB130001



PSG Tree Labels

S	- simple declarative clause
SBAR	- clause introduced by a subordinating conjunction
POS	- possessive ending
NP	- noun phrase
NN	- noun, singular or mass
NNS	- noun, plural
PRP	- personal pronoun
VP	- verb phrase
VBP	- verb, non-third person singular present
VB	- verb, base form
RB	- adverb
PP	- prepositional phrase
IN	- preposition or subordinating conjunction
ADJP	- adjective phrase
JJ	- adjective
DT	- determiner

Dependency Parse



DT - determiner

NN - noun, singular or mass

NNS - noun, plural

IN - preposition or subordinating conjunction

VBP - verb, non-third person singular present

VB - verb, base form

RB - adverb

JJ - adjective

PRP - personal pronoun

advcl - adverbial clause modifier - clause modifying the verb

mark - marker - word introducing a finite clause subordinate to another clause.

obl - oblique nominal - non-core (oblique) argument or adjunct.

nsubj - nominal subject - noun phrase which is the syntactic subject of a clause

case - case marking - relationship between preposition and noun.

aux - auxiliary - non-main verb of the clause

advmod - adverb modifier - adverb/adverb-headed phrase that serves to modify the meaning of word.

nmod - nominal modifier - relation for nominal dependents of another noun/noun phrase.

cop - copula - relationship between the complement of copular verb and the copular verb.

obj - object - the object of a verb

det - determiner

SRL Parse

Frames for are:

A lot of people are afraid of snakes because they do not understand them

ARG1 V ARG2 ARGM-CAU

Frames for do:

A lot of people are afraid of snakes because they do not understand them

V

Frames for understand:

A lot of people are afraid of snakes because they do not understand them

ARG0 ARGM-NEG V ARG1

ARG0

ARG1

ARG2

ARGM-CAU

ARGM-NEG

V

- argument - agent
- argument - patient
- argument - instrument, benefactive, attribute
- clause modifier - cause clauses
- negation modifier
- verb

Pros/cons

The PSG parse is easy to read in tree form and simple to understand. Even for my sentence with 2 clauses. However, it is very long and lengthy tree. This could become inefficient to use for long and complex sentences. My sentence had 9 parsing splits; many seemed a bit unnecessary (excessive POS). The dependency parse is easier to read each word label. It seems more efficient than the PSG because it has less relations to map compared to the amount of tree splits. However, with long and complex sentences with many relations can make the graph difficult to read all the relations. For example, my sentence had a reasonable amount of relations and is slightly difficult to read the overlapping relations. SRL parse is the easiest to read out of all of them, even for very complex sentences. My sentence was very easy to read as it shows the whole sentence together with a few highlighted labels. But, for long or complex sentences it typically needs multiple frames; therefore multiple parses. This could be very difficult to utilize in a program, for example, my sentence need 3 frames because it had 3 verbs. The way it parses is also very vague and doesn't go into nearly as much detail as the other parsers (doesn't split into types of nouns, etc).