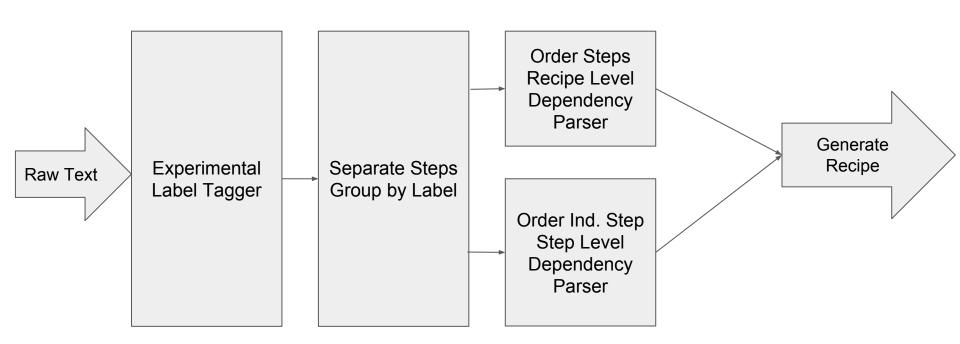
Turning Experimental Procedures into Machine-Readable Recipes

William Spitzer, Menghsuan Sam, Iveel Tsogsuren

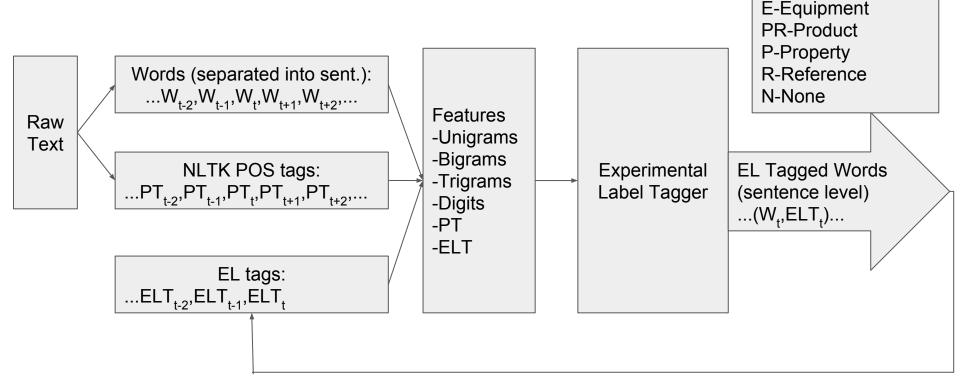
Process



Experimental Label Tagger

EL Tags: A-Action

I-Ingredient



Experimental Label Grouper and Step Sorter

EL Tagged Words (sentence level) ...(W_t,ELT_t)...

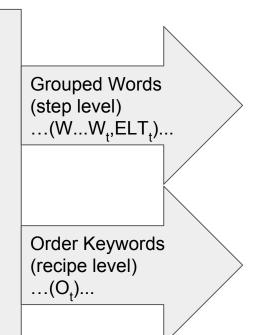
Rule Based Grouper

1) Group words with same label together ex (are, A), (calculated, A)

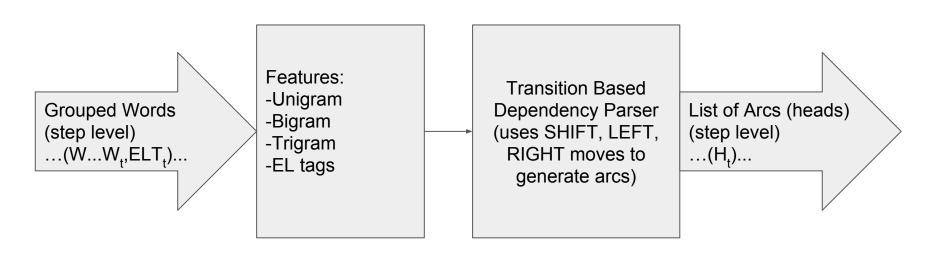
-> (are calculated, A)

Special cases (skip 1) are also grouped together ex (are, A), (then, N), (calculated, A)

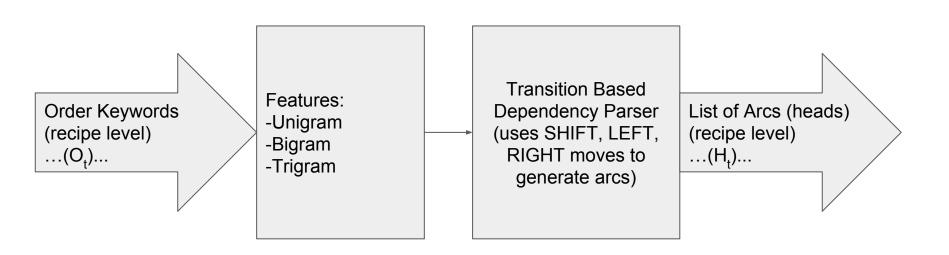
- ->(are (+) calculated, A)
- 2) Discards 'N' tagged words
- 3) Separates out Actions
 Each step contains exactly 1 Action
- 4) Extracts Ordering keywords ['first', 'second', 'third', 'fourth', 'fifth', 'next', 'then', 'after', 'before', 'last', 'lastly', 'finally']



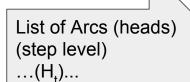
Dependency Parser for Grouped Words



Dependency Parser for Recipe Steps

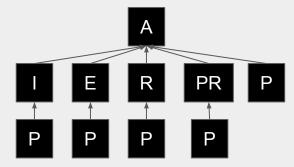


Generating Complete Recipe



List of Arcs (heads) (recipe level) ...(H,)...

Step Level: Use step level Arcs to link Properties to Ingredients, Equipments, Products, References, Actions, and link I/E/P/R to Actions.



Recipe Level: Order the steps based on the recipe level arcs

Complete Recipe!

Sample Output

```
6
                                  Python 2.7.10 Shell
File Edit Shell Debug Options Window Help
{'A': [('were pulverized', {'E': [('S/C composites', {}), ('ball mill', {})]})]
Step 2:
{'A': [('sieved', {'P': [('25 m opening stainless steel sieve .', {})]})]}
{'A': [('were prepared', {'I': [('anhydrous N-methyl-2-pyrolidinone (NMP)', {'P
': ['1 wt %']})], 'P': [('ratio of 1:5', {})], 'E': [('S/C composites', {}), ('
solution', {}) | } |
Step 4:
{'A': [('were applied', {'PR': [('slurries', {})], 'P': [('10', {}), ('120 C (+
) 4 h', {})]})]}
{'A': [('prepared', {'P': [('24.1 wt % sulfur loading', {'P': ['25.2 wt % were'
1})], 'E': [('identical procedure', {})]})]}
Step 6:
{'A': [('was used', {'E': [('S/C', {})]})]}
Step 7:
{'A': [('were assembled', {'PR': [('batteries', {}))], 'P': [('Swagelok', {}), (
'7', {}), ('10 mm diameter)', {})], 'E': [('S/C composite coated aluminum foil
(10 mm diameter', {}), ('lithium foil (7 mm thickness', {}), ('organic', {})],
'I': [('Celgard 3225 separator (10.3 mm diameter)', {})]})]}
{'A': [('were', {'PR': [('organic electrolytes', {})], 'I': [('(trifluoromethan
e)sulfonimide lithium (LiTFSI) (99.95% trace', {}), ('(DOL)', {})], 'P': [('55:
40', {})]})]}
Step 9:
{'PR': [('organic electrolyte', {'A': [('filled', {})]})]}
Step 10:
{'A': [('were pressed', {'E': [('spring', {})]})]}
Step 11:
{'A': [('were tested', {'PR': [('batteries', {})], 'E': [('Maccor 4000 series b
attery tester', {})]})]}
{'A': [('were cycled', {'PR': [('batteries', {})], 'P': [('between 1.0 to 3.6 V
', {})]})]}
Step 13:
{'A': [('was started', {})]}
{'A': [('were tested', {'P': [('0.5 mA', {})], 'R': [('batteries', {})]})]}
Step 15:
{'A': [('was set', {'P': [('0.05', {})]})]}
{'A': [('is', {'I': [('discharge', {}), ('sulfur', {})]})]}
>>>
```