

# Root and Rule-Based Natural Language Processing: Subject-Predicate-Object Phrases

Peter Zhang and Jonah Tash

# Motivation

- We have used a root and rule-based programs
  - Can identify terms and build taxonomies
  - Next step is to identify relations and build RDFs
- We want to express the terms and sentences in subject-predicate-object groups

# Example

The crystal packing is mainly stabilized by van der Waals interactions

crystal:0:packing

van:2:der:1:waals:0:interacitons

crystal packing

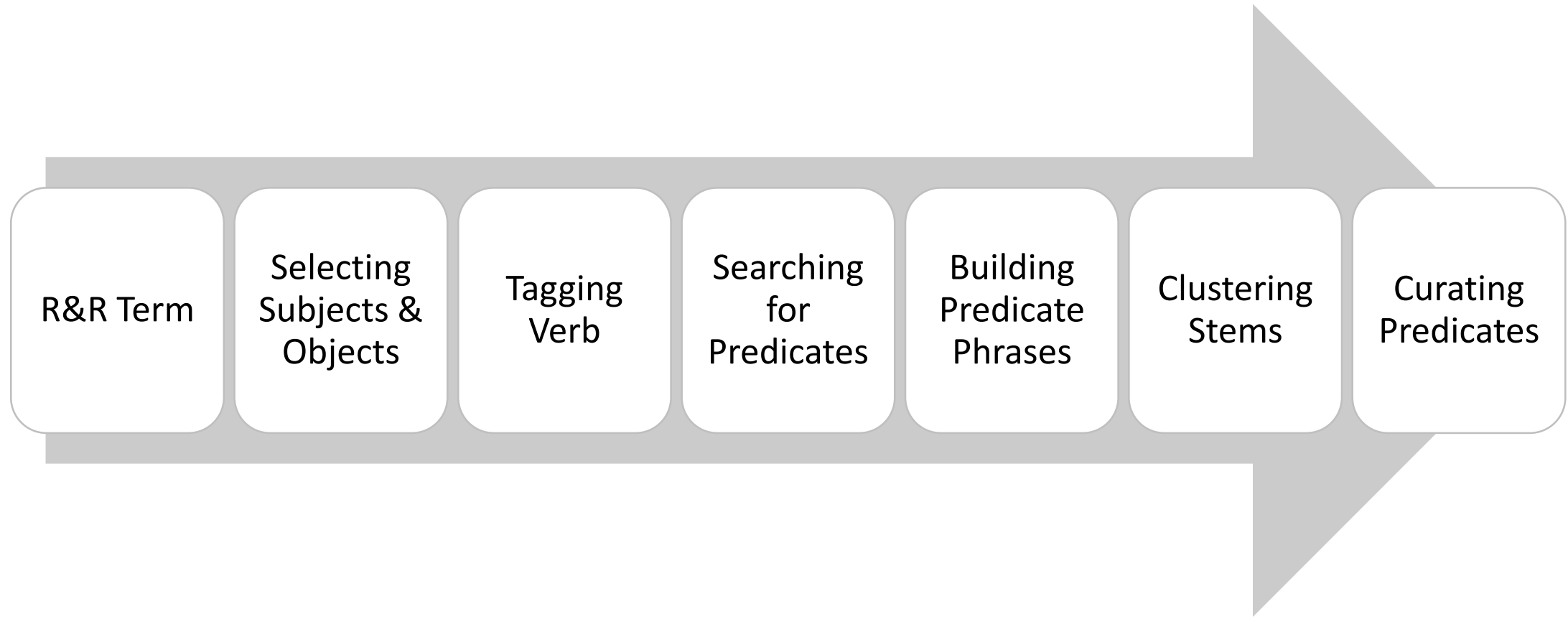
stabilized by

Van der Waals interactions

Subject

Predicate Phrase

Object



# Process

# R&R Term

term	originals
crystal:0:pack	crystal packing
crystal	crystal
pack	packing
van:2:der:1:waal:0:interaction	van der Waals interactions
van	van
der:1:waal:0:interaction	der Waals interactions
der	der
waal:0:interaction	Waals interactions
waal	Waals
interaction	interactions
one:1:triazine:0:molecule	One triazine molecule
triazine:0:molecule	triazine molecule
triazine	triazine
crystal:0:conformation	two conformations within the crystal
conformation	conformations
crystal	crystal
occupancy	occupancies

The crystal packing is mainly stabilized by van der Waals interactions

One triazine molecule is disordered over two conformations within the crystal, the occupancies being 62 emsp14 (1) and 38 emsp14

# Selecting Subjects & Objects

- Considered only short noun phrases
  - Want precise but useful search terms
  - Subjects and objects should be concise
  - Consider terms of level 0, 1 and 2

# Selecting Subjects & Objects

term	originals
crystal:0:pack	crystal packing
crystal	crystal
pack	packing
van:2:der:1:waal:0:interaction	van der Waals interactions
van	van
der:1:waal:0:interaction	der Waals interactions
der	der
waal:0:interaction	Waals interactions
waal	Waals
interaction	interactions
one:1:triazine:0:molecule	One triazine molecule
triazine:0:molecule	triazine molecule
triazine	triazine
crystal:0:conformation	two conformations within the crista
conformation	conformations
crystal	crystal
occupancy	occupancies

# Step 1: Tagging Verbs

- Use the part-of-speech tagger in the Natural Language Toolkit (NLTK) library to identify verbs
  - The model is trained by machine learning
  - It can isolate the sense of a word
  - E.g. “I wrote the report.”  $\Rightarrow$  “report” *noun*  
“I report the news”  $\Rightarrow$  “report” *verb*

Found at: <https://www.nltk.org/>



# Step 1: Cross-Checking

- NLTK is not perfect
  - Model is trained on ordinary literature, not scientific vocabulary
  - Symbols and molecule names are sometimes incorrectly classified as verbs
  - E.g. “middot” in  $\text{KNa}[\text{H}_6\text{PtMo}_6\text{O}_{24}]$  middot  $11\text{H}_2\text{O}$   
“oxazolo” in “oxazolo[3,2-c]pyrimidin-1-one hydrate”

# Step 1: Cross-Checking

- First, check results with Princeton's WordNet lexicon of verbs
  - Ensures a valid verb form is tagged
  - E.g. removes “middot” and “oxazolo”
- Second, remove tagged “verbs” that also appear in the terms
  - Some verb forms appear in noun phrases
  - E.g. “packing” in “crystal packing”

Found at: <https://wordnet.princeton.edu/>

## Step 2: Grouping Terms

- Classify terms into Level 0, 1, and 2 by maximum colon number
  - Higher level means a longer phrase
  - E.g. “crystal:0:packing”  $\Rightarrow$  Level 0  
“van:2:der:1:waals:0:interaction”  $\Rightarrow$  Level 2
- Conduct searches first on *only* Level 0 terms, then on Level 0 *and* Level 1, and finally on *all three* levels

## Step 2: Locating All Verbs

- Search the span between subject and object for verbs
  - E.g. between “crystal packing” and “Van der Waals interactions”
- Count the number of characters and words in the span

The **crystal packing** is mainly stabilized by **van der Waals interactions**

Span

Characters: 23

Words: 4

## Step 3: Catching Conjugations

- A verb by itself is not precise enough
  - Not useful for searches
  - E.g. “was”, “have”, “link”
- Use the Collaborative International Dictionary of English to capture conjugations of the verb
  - Hand-collected a list of English conjugations
  - E.g. *is* linking, *was* linked, *will* link

Found at: <http://www.ibiblio.org/webster/>

## Step 3: Collating Prepositions

- Reference the Pattern Dictionary of English Propositions to append neighboring prepositions
  - e.g. “stabilized”  $\Rightarrow$  “stabilized by”  
“connect”  $\Rightarrow$  “connect to”  
“extracting”  $\Rightarrow$  “extracting from”

Found at: <https://www.clres.com/pdep.html>

## Step 4: Stemming Verb Forms

- Want to group synonymous verb phrases
- Reference the British National Corpus to find the “stem” of each verb phrase
  - E.g. “is”  $\Rightarrow$  “be”  
“stabilized by”  $\Rightarrow$  “stabilize”

Found at: <https://github.com/skywind3000/lemma.en>

## Step 4: Clustering Synonyms

- Reference the Princeton's WordNet lexicon to find synonyms for the stems
  - Use these to cluster similar phrases
  - E.g. “appear”  $\Rightarrow$  {“appear”, “seem”, “look”}  
“bond”  $\Rightarrow$  {“bond”, “bind”, “attach”, “adhere”, “stick”}

Found at: <https://wordnet.princeton.edu/>



# Step 4: Clustering

{ 'organise', 'coordinate', 'align', 'organize', 'ordinate' }
{ 'dispatch', 'finish', 'complete', 'nail', 'discharge' }
{ 'concede', 'yield', 'grant', 'soften', 'generate', 'cede', 'succumb', 'render', 'afford', 'relent', 'give' }
{ 'dispatch', 'finish', 'complete', 'nail', 'discharge' }
{ 'concede', 'yield', 'grant', 'soften', 'generate', 'cede', 'succumb', 'render', 'afford', 'relent', 'give' }
{ 'acquire', 'borrow', 'embrace', 'assume', 'espouse', 'dramatise', 'adopt', 'dramatize', 'follow', 'take' }
{ 'connect', 'colligate', 'join', 'yoke', 'unite', 'associate', 'relate', 'link', 'tie' }
{ 'stabilise', 'steady', 'stabilize', 'brace' }
{ 'site', 'situate', 'settle', 'place', 'locate' }
{ 'betoken', 'argue', 'point', 'bespeak', 'show', 'signal', 'indicate', 'suggest' }
{ 'expose', 'exhibit', 'present', 'parade', 'demo', 'demonstrate', 'display', 'show', 'march' }
{ 'compile', 'write', 'indite', 'compose', 'frame', 'pen' }
{ 'bind', 'attach', 'adhere', 'stick', 'tie', 'bond' }
{ 'bind', 'attach', 'adhere', 'stick', 'tie', 'bond' }
{ 'organise', 'constitute', 'spring', 'form', 'organize', 'mould', 'imprint', 'forge', 'mold', 'shape' }

## Step 5: Curation

- Count the number of each subject, predicate, object, and verb stem
- Sort by frequency and identify phrases to ignore
  - Some predicate phrases lack meaningful relation
  - E.g. “is”, “was”, “are”, “have”
- Locally compile phrases to filter out

# Output

- For an input of 67k terms in 5681 sentences, the program extracts 5700+ pairs in under 15 seconds
- Output to a CSV file with the following columns:
  - Subject, Object, Predicate
  - Sentence
  - Stem and Synonyms
  - Frequencies of Subject, Object, Predicates and Stems
  - R&R Terms for Subject, Object, and Predicate
  - Document ID

# Output

- {'quaternary structure', 'exhibits', 'considerable variability'}
  - {'quaternary:0:structure', 'exhibits', 'considerable:0:variability'}
  - However the quaternary structure exhibits considerable variability
- {'Cl hydrogen bonds', 'stabilize', 'crystal structure'}
  - {'Cl:2:hydrogen:bond', 'stabilize', 'crystal:0:structure'}
  - Cl hydrogen bonds help to stabilize the crystal structure
- {'three lobes or fingers', 'delineate', 'central binding groove and additional grooves'}
  - {'finger:0:lobe', 'delineate', 'lobe:0:lobe:2:additional:0:groove:1:central:0:bind:1:groove'}
  - The three lobes or fingers delineate a central binding groove and additional grooves between lobes 1 and 3 and between lobes 2 and 3

# Implementation

- The program was written in Python with use of the NLTK, CSV and SQLite3 libraries
- Compilers for the verb, predicate, and lemma dictionaries are available as separate programs
  - Dictionaries can be easily expanded and re-integrated
- Filtered words are manually stored in a separated in a CSV file