

Problem 2

For this problem set, we will use

<https://app.sketchengine.eu/#dashboard?corpname=preloaded%2F covid19>

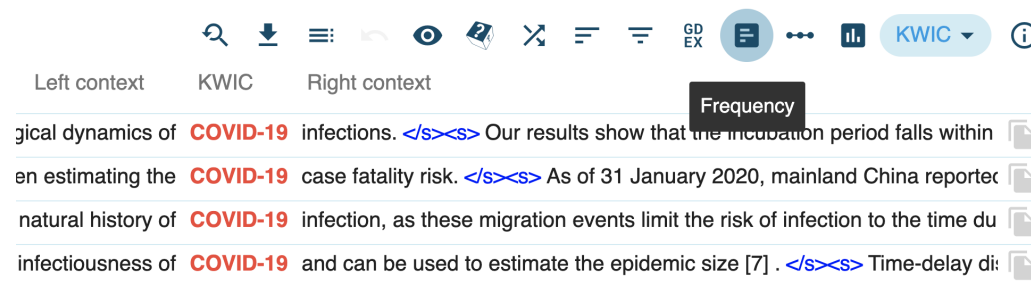
Problem 2.1

Click the above link, and follow this: Dashboard -> Concordance -> Advanced -> CQL.

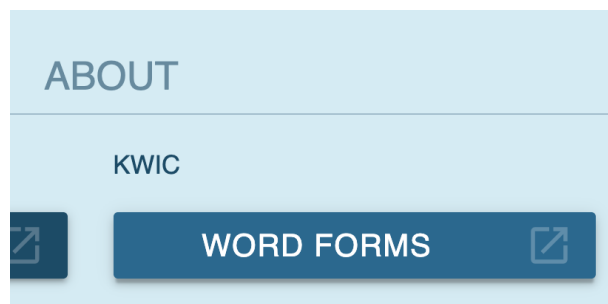
Now write a query to find sentences containing all forms of covid and execute it. Some forms include covid-19, covid19, COVID19, covid-36, covid-54.

Once you get the sentences, click `Frequency -> KWIC > WORD FORMS` to generate the frequency of words. These steps are shown below:

Step 1:



Step 2:



Step 3: The word list looks something like this:

		Word	↓ Frequency	Per million tokens
1	<input type="checkbox"/>	COVID-19	20,773	73.99
2	<input type="checkbox"/>	Covid-19	429	1.53
3	<input type="checkbox"/>	COVID19	169	0.60
4	<input type="checkbox"/>	COVID-2019	157	0.56
5	<input type="checkbox"/>	CoVID-19	32	0.11

What is the CQL query that you used for getting all forms of covid?

Answer: [word="(?)covid-?\d+\$"]

Include the snapshot of the top 20 words (5 words are shown above)?

Answer:

		Word	Frequency ↓	Relative [?]
1	<input type="checkbox"/>	COVID-19	20,773	73.99
2	<input type="checkbox"/>	Covid-19	429	1.53
3	<input type="checkbox"/>	COVID19	169	0.60
4	<input type="checkbox"/>	COVID-2019	157	0.56
5	<input type="checkbox"/>	CoVID-19	32	0.11
6	<input type="checkbox"/>	covid-19	30	0.11
7	<input type="checkbox"/>	CoViD-19	10	0.04
8	<input type="checkbox"/>	COVID-10	7	0.02
9	<input type="checkbox"/>	COVID-9	7	0.02
10	<input type="checkbox"/>	Covid-2019	4	0.01
11	<input type="checkbox"/>	covid19	3	0.01
12	<input type="checkbox"/>	Covid19	2	< 0.01
13	<input type="checkbox"/>	covid-10	1	< 0.01
14	<input type="checkbox"/>	COVID-138	1	< 0.01
15	<input type="checkbox"/>	Covid-10	1	< 0.01
16	<input type="checkbox"/>	Covid-56	1	< 0.01
17	<input type="checkbox"/>	COVID-173	1	< 0.01
18	<input type="checkbox"/>	COVID-27	1	< 0.01
19	<input type="checkbox"/>	COVID-110	1	< 0.01
20	<input type="checkbox"/>	COVID-2	1	< 0.01

Problem 2.2

Let's write CQL queries to find interesting words that occur in specific syntactic relations with covid (all forms). We did similar things in class. You will have to use tag and lemma in CQL queries. This [tagset](#) could be useful

I will demonstrate how to get the modifiers of covid:

Step 1: First write a CQL query that produces concordance (examples) like this:

Left context	KWIC	Right context
which even caused	severe COVID-19	pneumonia. </S></S> All rig
screening-identified	asymptomatic COVID-19	cases in Nanjing were adr
relative 1 developed	severe COVID-19	pneumonia and was admit

Step 2:

FREQUENCY

BASIC

ADVANCED

ABOUT

Select an attribute and its position in the concordance: ?

word

left context

6

5

4


3

2

1

KWIC

Step 3:


FREQUENCY

BASIC
 ADVANCED
 ABOUT

Select an attribute and its position in the concordance: ?
 word

left context

6
 5
 4
 3
 2
 1

KWIC
 First KWIC word
 Last KWIC word

☐ Group by first column

Step 4:

		Word	↓ Frequency	Per million tokens
1	<input type="checkbox"/>	severe	298	1.06
2	<input type="checkbox"/>	confirmed	115	0.41
3	<input type="checkbox"/>	current	103	0.37

What is the CQL query for modifiers of covid (all forms)?

Answer: [tag="J.*"][word="(?)covid-?\d+\$"]

Include the snapshot of modifiers (top three are shown above)

1	<input type="checkbox"/>	severe	298	1.06
2	<input type="checkbox"/>	confirmed	115	0.41
3	<input type="checkbox"/>	current	103	0.37
4	<input type="checkbox"/>	suspected	81	0.29
5	<input type="checkbox"/>	laboratory-confirmed	69	0.25
6	<input type="checkbox"/>	ongoing	64	0.23
7	<input type="checkbox"/>	new	43	0.15
8	<input type="checkbox"/>	first	42	0.15
9	<input type="checkbox"/>	mild	40	0.14
10	<input type="checkbox"/>	reported	31	0.11
11	<input type="checkbox"/>	critical	27	0.10
12	<input type="checkbox"/>	potential	26	0.09
13	<input type="checkbox"/>	global	24	0.09
14	<input type="checkbox"/>	ill	22	0.08
15	<input type="checkbox"/>	asymptomatic	18	0.06
16	<input type="checkbox"/>	early	17	0.06
17	<input type="checkbox"/>	moderate	16	0.06
18	<input type="checkbox"/>	active	15	0.05
19	<input type="checkbox"/>	novel	14	0.05
20	<input type="checkbox"/>	recent	14	0.05

What is the CQL query of words that are modified by covid (all forms)?

Answer: [word="(?)covid-?\d+\$"][tag="N.*"]

Include the snapshot of those words

1	<input type="checkbox"/>	patients	1,720	6.13
2	<input type="checkbox"/>	cases	954	3.40
3	<input type="checkbox"/>	outbreak	721	2.57
4	<input type="checkbox"/>	infection	696	2.48
5	<input type="checkbox"/>	epidemic	540	1.92
6	<input type="checkbox"/>	pneumonia	496	1.77
7	<input type="checkbox"/>	pandemic	409	1.46
8	<input type="checkbox"/>	resource	396	1.41
9	<input type="checkbox"/>	virus	153	0.54
10	<input type="checkbox"/>	case	147	0.52
11	<input type="checkbox"/>	infections	144	0.51
12	<input type="checkbox"/>	transmission	141	0.50
13	<input type="checkbox"/>	disease	125	0.45
14	<input type="checkbox"/>	patient	104	0.37
15	<input type="checkbox"/>	spread	63	0.22
16	<input type="checkbox"/>	diagnosis	58	0.21
17	<input type="checkbox"/>	outbreaks	55	0.20
18	<input type="checkbox"/>	testing	55	0.20
19	<input type="checkbox"/>	treatment	51	0.18
20	<input type="checkbox"/>	mortality	50	0.18

What is the CQL query for words that occur in right coordination with covid (all forms)
(e.g., in COVID-19 , SARS-2002 , and HCoV-NL63, the words iSARS-2002 and HCoV-NL63 are the right conjuncts/coordinates).

Answer: [word="(?)covid-?\d+\$"]([tag="CC" | word=",,"][tag="J.*"]{0,3}[tag="NP.*"]{1,10}

Include the snapshot of those words

1	<input type="checkbox"/>	SARS	47	0.17
2	<input type="checkbox"/>	MERS-COV	14	0.05
3	<input type="checkbox"/>	SARS-CoV-2	14	0.05
4	<input type="checkbox"/>	H1N1	9	0.03
5	<input type="checkbox"/>	MERS	9	0.03
6	<input type="checkbox"/>	SARS-2002	7	0.02
7	<input type="checkbox"/>	HAPE	6	0.02
8	<input type="checkbox"/>	CT	5	0.02
9	<input type="checkbox"/>	China	4	0.01
10	<input type="checkbox"/>	ARIMA	4	0.01
11	<input type="checkbox"/>	SARS-CoV	4	0.01
12	<input type="checkbox"/>	ACE2	4	0.01
13	<input type="checkbox"/>	CDC	3	0.01
14	<input type="checkbox"/>	COVID-19	3	0.01
15	<input type="checkbox"/>	CoV	3	0.01
16	<input type="checkbox"/>	H1N1-induced	2	< 0.01
17	<input type="checkbox"/>	January	2	< 0.01
18	<input type="checkbox"/>	SPP	2	< 0.01
19	<input type="checkbox"/>	CancerROP	2	< 0.01
20	<input type="checkbox"/>	Arbidol	2	< 0.01

What is the CQL query for verbs that can take covid (all forms) as subject?

Answer: [word="(?)covid-?\d+\$"] [tag="V.*"]

Include the snapshot of verbs that take covid as subject

1	<input type="checkbox"/>	is	726	2.59
2	<input type="checkbox"/>	has	346	1.23
3	<input type="checkbox"/>	was	299	1.06
4	<input type="checkbox"/>	were	235	0.84
5	<input type="checkbox"/>	are	194	0.69
6	<input type="checkbox"/>	have	133	0.47
7	<input type="checkbox"/>	had	73	0.26
8	<input type="checkbox"/>	confirmed	60	0.21
9	<input type="checkbox"/>	based	33	0.12
10	<input type="checkbox"/>	using	30	0.11
11	<input type="checkbox"/>	include	29	0.10
12	<input type="checkbox"/>	remains	23	0.08
13	<input type="checkbox"/>	reported	23	0.08
14	<input type="checkbox"/>	showed	22	0.08
15	<input type="checkbox"/>	caused	22	0.08
16	<input type="checkbox"/>	according	21	0.07
17	<input type="checkbox"/>	spreading	17	0.06
18	<input type="checkbox"/>	seems	17	0.06
19	<input type="checkbox"/>	continues	17	0.06
20	<input type="checkbox"/>	appears	17	0.06

What is the CQL query for verbs that can take covid (all forms) as object?

Answer: [tag="V.*"][tag="J.*"]{0,3}[word="(?)covid-?\d+\$"]

Include the snapshot of verbs that take COVID as object.

1	<input type="checkbox"/>	confirmed	352	1.25
2	<input type="checkbox"/>	treat	57	0.20
3	<input type="checkbox"/>	suspected	52	0.19
4	<input type="checkbox"/>	treating	47	0.17
5	<input type="checkbox"/>	have	42	0.15
6	<input type="checkbox"/>	hospitalized	41	0.15
7	<input type="checkbox"/>	having	34	0.12
8	<input type="checkbox"/>	named	33	0.12
9	<input type="checkbox"/>	declared	33	0.12
10	<input type="checkbox"/>	including	30	0.11
11	<input type="checkbox"/>	reported	29	0.10
12	<input type="checkbox"/>	diagnosed	27	0.10
13	<input type="checkbox"/>	causes	24	0.09
14	<input type="checkbox"/>	causing	22	0.08
15	<input type="checkbox"/>	regarding	21	0.07
16	<input type="checkbox"/>	control	21	0.07
17	<input type="checkbox"/>	detect	20	0.07
18	<input type="checkbox"/>	diagnose	20	0.07
19	<input type="checkbox"/>	prevent	19	0.07
20	<input type="checkbox"/>	called	19	0.07

Problem 2.3

What are the most important words that form collocations with COVID (where covid is the right word)?

You can generate collocations as follows: First get concordance of all forms of covid.

Step 1:

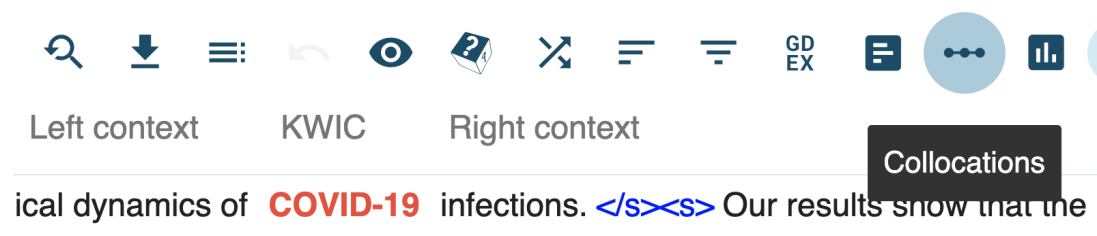
Left context

KWIC

Right

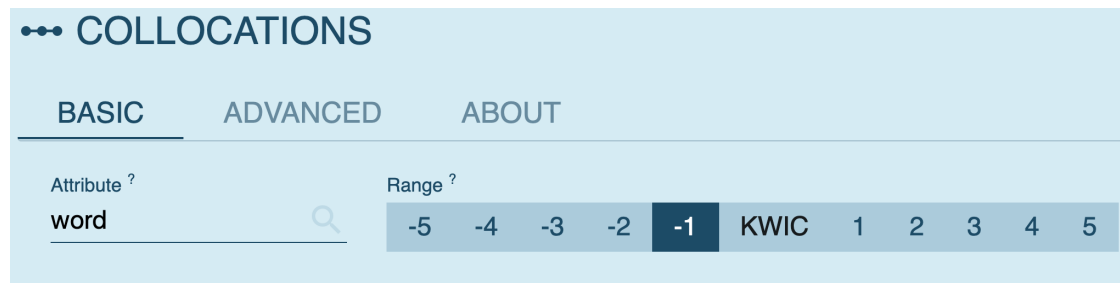
epidemiological dynamics of **COVID-19** infect
red when estimating the **COVID-19** case
idv the natural history of **COVID-19** infect

Step 2:



The screenshot shows a software interface with a search bar containing the text 'COVID-19'. Below the search bar, there are three tabs: 'Left context', 'KWIC', and 'Right context'. To the right of these tabs is a 'Collocations' button. The interface also includes a toolbar with various icons for search, download, and other functions.

Step 3:



The screenshot shows the 'COLLOCATIONS' interface. The 'BASIC' tab is selected. Below the tabs, there is a search bar with the text 'word'. To the right of the search bar is a range selection bar with the following values: -5, -4, -3, -2, -1, KWIC, 1, 2, 3, 4, 5. The value -1 is highlighted.

Step 4:

	Word	Cooccurrences ?
1	<input type="checkbox"/> confirmed	458
2	<input type="checkbox"/> suspected	133

Show the collocations sorted according to what you think is the best metric (T-Score, MI, LogDice). Indicate the metric you used.

LogDice appears to be best at finding appropriate collocations:

1	<input type="checkbox"/>	confirmed	458	65,495	21.17	6.50	7.43
2	<input type="checkbox"/>	suspected	133	21,439	11.39	6.33	6.66
3	<input type="checkbox"/>	laboratory-confirmed	75	3,601	8.63	8.08	6.61
4	<input type="checkbox"/>	severe	298	112,078	16.76	5.11	6.19
5	<input type="checkbox"/>	ongoing	64	12,451	7.88	6.06	5.94
6	<input type="checkbox"/>	treat	57	14,546	7.40	5.67	5.69
7	<input type="checkbox"/>	treating	46	9,478	6.67	5.98	5.60
8	<input type="checkbox"/>	current	103	50,596	9.76	4.72	5.55
9	<input type="checkbox"/>	declared	33	4,219	5.69	6.66	5.39
10	<input type="checkbox"/>	towards	44	17,999	6.42	4.99	5.18
11	<input type="checkbox"/>	hospitalized	40	15,225	6.14	5.09	5.15
12	<input type="checkbox"/>	named	33	9,921	5.61	5.43	5.10
13	<input type="checkbox"/>	about	114	98,409	9.97	3.91	4.96
14	<input type="checkbox"/>	non-severe	21	825	4.57	8.37	4.94
15	<input type="checkbox"/>	with	1,999	2,412,053	40.55	3.43	4.75
16	<input type="checkbox"/>	mild	40	30,282	5.96	4.10	4.66
17	<input type="checkbox"/>	against	151	180,158	11.16	3.44	4.62
18	<input type="checkbox"/>	contracted	17	1,822	4.09	6.92	4.57
19	<input type="checkbox"/>	having	34	27,817	5.46	3.99	4.49
20	<input type="checkbox"/>	of	5,930	8,766,274	68.23	3.13	4.47

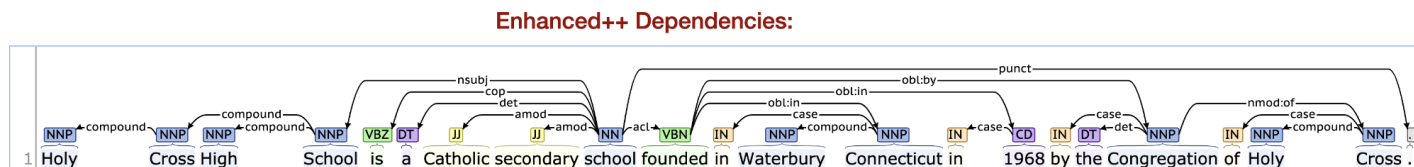
Problem 3:

Write [SemGreX](#) regular expressions that can detect organizations and their founders. Make use of <https://corenlp.run> to parse sentences to syntactic graphs and for running SemGreX expressions.

Here is an example:

Holy Cross High School is a Catholic secondary school founded in Waterbury Connecticut in 1968 by the **Congregation of Holy Cross** .

The corresponding Enhanced++ Dependencies syntactic graph is as follows:



The below SemGreX pattern extracts the headword of the organization and the headword of the founder.

`{}=organization <nsubj ({} >acl ({lemma:found} >/obl:by/ {}=founder))`

CoreNLP Tools:

TokensRegex
Semgrex
Tregex

Enter a **Semgrex** expression to run against the "enhanced dependencies" above:

`{}=organization <nsubj ({} >cop {} >acl ({lemma:found} >/obl:by/ {}=founder))`
Match

1 Holy Cross High School is a Catholic secondary school founded in Waterbury Connecticut in 1968 by the Congregation of Holy Cross .

match

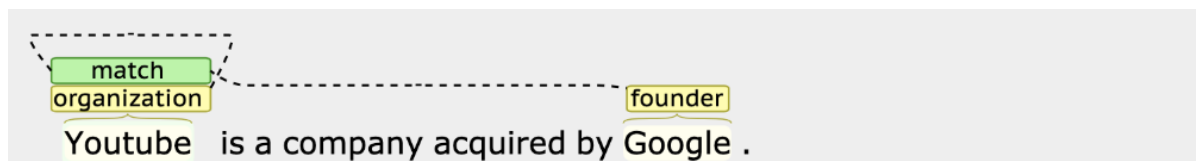
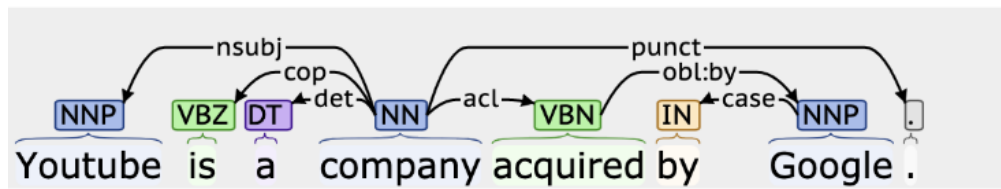
organization

founder

This pattern can be read as the “organization” that is a subject of something, and this something is founded by the founder.

Here it extracts School (i.e., the headword of Holy Cross High School) as the organization and Congregation (i.e., the headword of the Congregation of Holy Cross) as the founder.

Your goal is to write SemGreX expressions that can generalize to multiple sentences but at the same time don't match incorrect sentences. For example, if you don't use {lemma:found} in the above sentence, your pattern will also match a sentence like "Youtube is a company acquired by Google" (see below.)

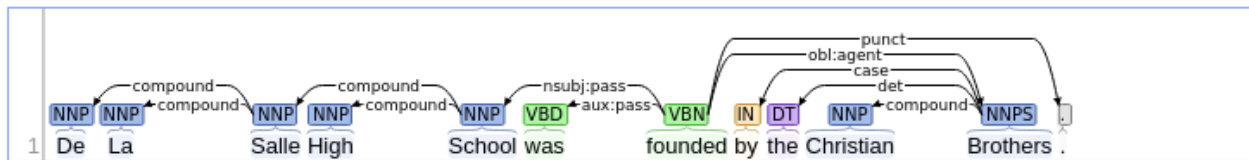


Problem 3.1

Write the SemGreX patterns for the following sentences that extract the organization name (headword is enough) and its founder (headword is enough). Sentences that can make use of the same expression should be in the same snapshot (containing Enhanced++ Dependencies, Semgrex expression, and the matchings):

De La Salle High School was founded by the Christian Brothers .

Enhanced++ Dependencies:



CoreNLP Tools:

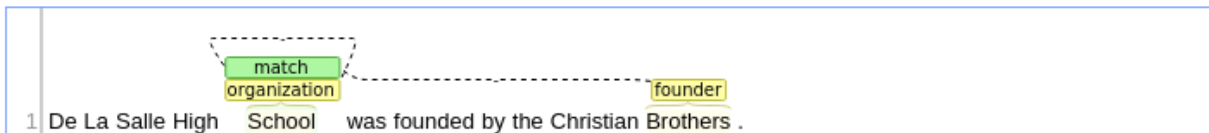
TokensRegex

Semgrex

Tregex

Enter a **Semgrex** expression to run against the "enhanced

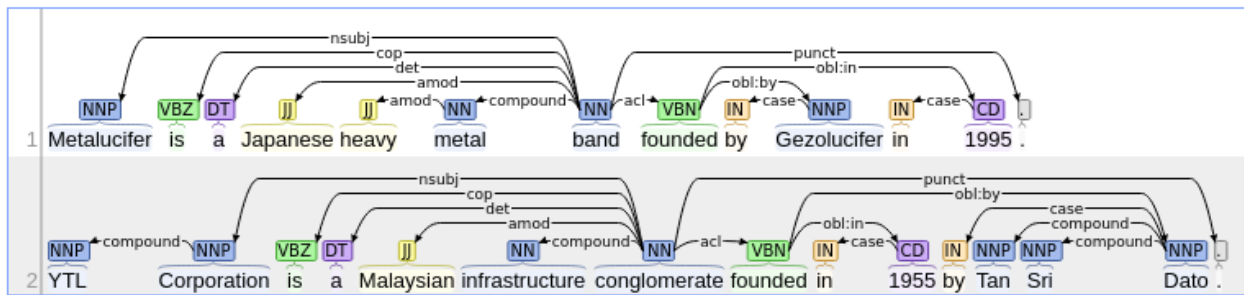
{ }=organization </nsubj:pass/ ({lemma:found } >/obl:agent/ { }=founder)



Metalucifer is a Japanese heavy metal band founded by **Gezolucifer** in 1995 .

YTL Corporation is a Malaysian infrastructure conglomerate founded in 1955 by **Tan Sri Dato**.

Enhanced++ Dependencies:



CoreNLP Tools:

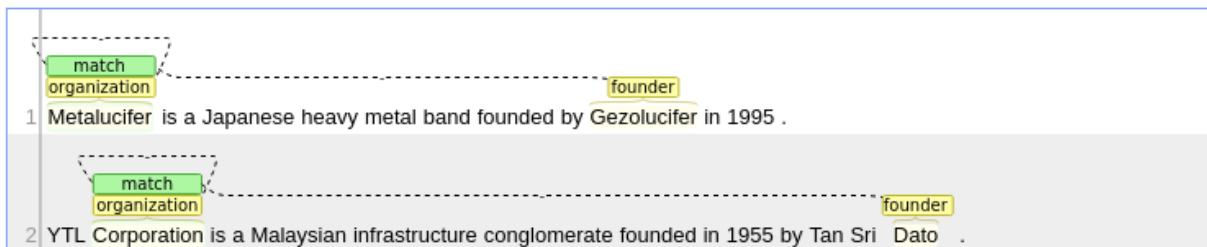
TokensRegex

Semgrex

Tregex

Enter a **Semgrex** expression to run against the "enhanced dependen

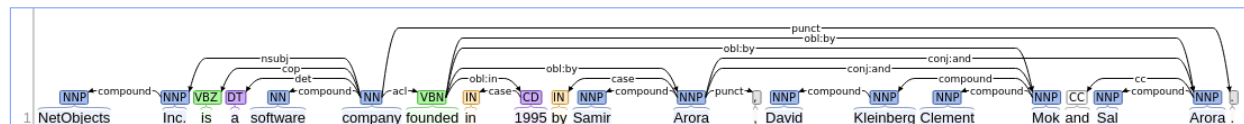
{}=organization <nsubj {} >acl ({lemma:found} >/obl:by/ {}=founder))



NetObjects Inc. is a software company founded in 1995 by **Samir Arora, David Kleinberg Clement Mok and Sal Arora** .

(If there are multiple founders, you have to extract headword corresponding to each founder)

Enhanced++ Dependencies:



CoreNLP Tools:

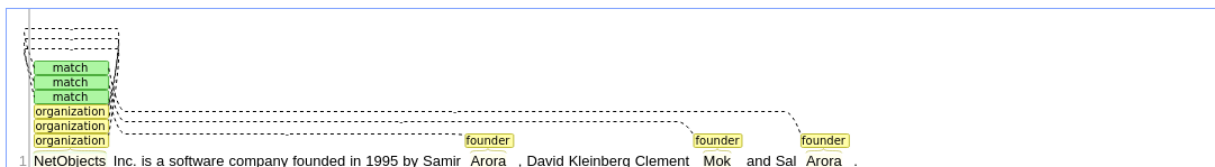
TokensRegex

Semgrex

Tregex

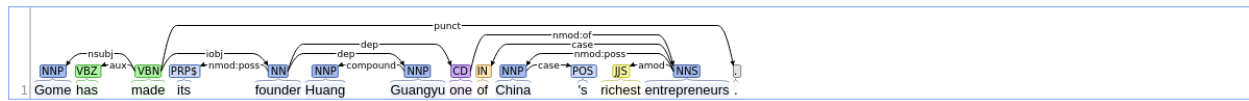
Enter a **Semgrex** expression to run against the "enhanced dependencies" above:

{}=organization <compound {} <nsubj {} >acl ({lemma:found} >/obl:by/ {}=founder)))



Gome has made its founder **Huang Guangyu** one of China's richest entrepreneurs.

Enhanced++ Dependencies:



CoreNLP Tools:

[TokensRegex](#) [Semgrep](#) [Tregex](#)

Enter a [Semgrep](#) expression to run against the "enhanced dependencies" above:

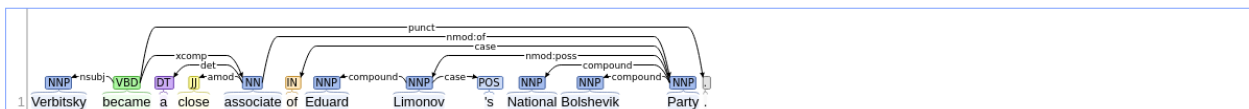
`{}=organization <nsubj ({} >iobj ((lemma:founder) >dep {tag:NNP}=founder))`

Match



Verbitsky became a close associate of **Eduard Limonov's National Bolshevik Party**.

Enhanced++ Dependencies:



CoreNLP Tools:

[TokensRegex](#) [Semgrep](#) [Tregex](#)

Enter a [Semgrep](#) expression to run against the "enhanced dependencies" above:

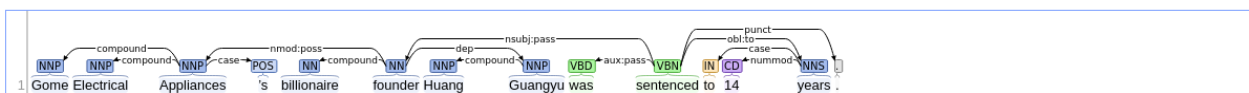
`{}=organization >nmod:poss/ {tag:NNP}=founder`

Match



Gome Electrical Appliances's billionaire founder **Huang Guangyu** was sentenced to 14 years.

Enhanced++ Dependencies:



CoreNLP Tools:

[TokensRegex](#) [Semgrep](#) [Tregex](#)

Enter a [Semgrep](#) expression to run against the "enhanced dependencies" above:

`{}=organization <nmod:poss/ ((lemma:founder) >dep {}=founder)`

Match

