Assignment 2: Regular Expressions

Sam Zhang 261072449

2024-02-06

Problem 1

Located in the notebook file.

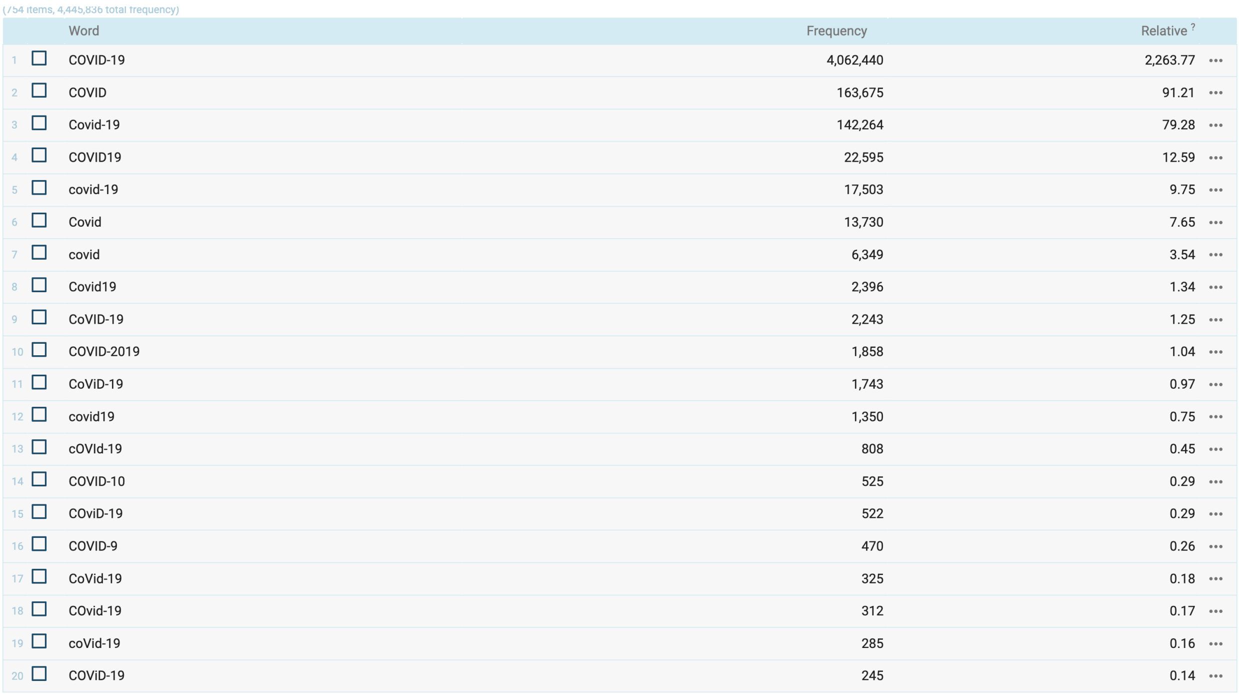
Problem 2

Problem 2.1

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

```
[word="(?i)covid(-|\s)?(\d+)?"]
```

Include the snapshot of the 20 most frequent forms of Covid:



(754 items, 4,445,836 total frequency)

		Word	Frequency	Relative ?	
1	□	COVID-19	4,062,440	2,263.77	•••
2	□	COVID	163,675	91.21	•••
3	□	Covid-19	142,264	79.28	•••
4	□	COVID19	22,595	12.59	•••
5	□	covid-19	17,503	9.75	•••
6	□	Covid	13,730	7.65	•••
7	□	covid	6,349	3.54	•••
8	□	Covid19	2,396	1.34	•••
9	□	CoVID-19	2,243	1.25	•••
10	□	COVID-2019	1,858	1.04	•••
11	□	CoViD-19	1,743	0.97	•••
12	□	covid19	1,350	0.75	•••
13	□	cOVId-19	808	0.45	•••
14	□	COVID-10	525	0.29	•••
15	□	COviD-19	522	0.29	•••
16	□	COVID-9	470	0.26	•••
17	□	CoVid-19	325	0.18	•••
18	□	COvid-19	312	0.17	•••
19	□	coVid-19	285	0.16	•••
20	□	COViD-19	245	0.14	•••

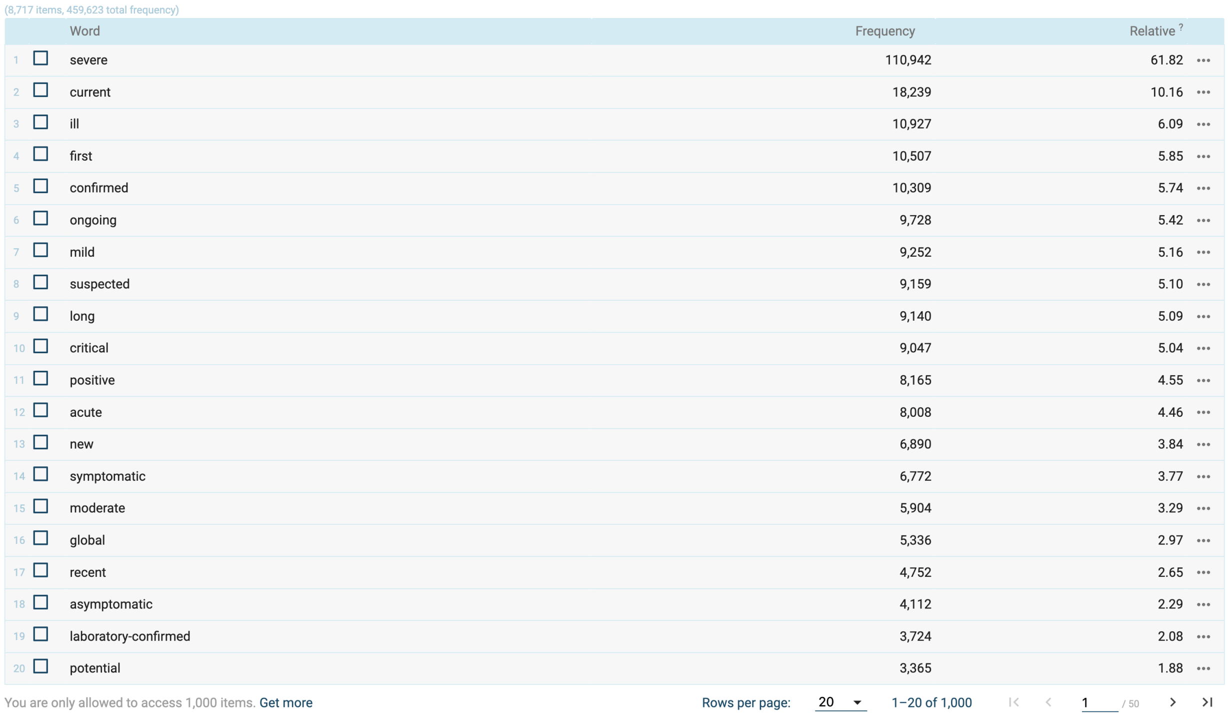
Figure 1: Top 20 Most Frequent Forms of COVID

Problem 2.2

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

```
[tag="JJ"] [word="(?i)covid(-|\s)?(\d+)?"]
```

Include the snapshot of the 20 most frequent modifiers modifiers:



(8,717 items, 459,623 total frequency)

		Word	Frequency	Relative ?	
1	□	severe	110,942	61.82	•••
2	□	current	18,239	10.16	•••
3	□	ill	10,927	6.09	•••
4	□	first	10,507	5.85	•••
5	□	confirmed	10,309	5.74	•••
6	□	ongoing	9,728	5.42	•••
7	□	mild	9,252	5.16	•••
8	□	suspected	9,159	5.10	•••
9	□	long	9,140	5.09	•••
10	□	critical	9,047	5.04	•••
11	□	positive	8,165	4.55	•••
12	□	acute	8,008	4.46	•••
13	□	new	6,890	3.84	•••
14	□	symptomatic	6,772	3.77	•••
15	□	moderate	5,904	3.29	•••
16	□	global	5,336	2.97	•••
17	□	recent	4,752	2.65	•••
18	□	asymptomatic	4,112	2.29	•••
19	□	laboratory-confirmed	3,724	2.08	•••
20	□	potential	3,365	1.88	•••

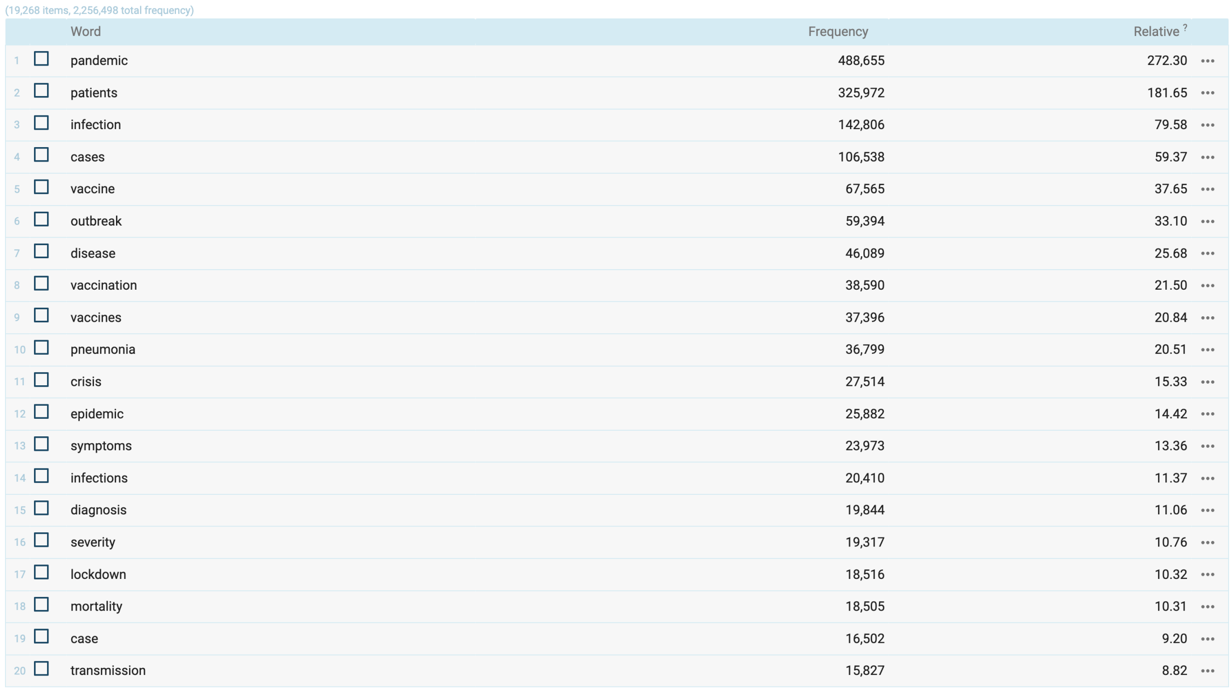
You are only allowed to access 1,000 items. Get more

Figure 2: Top 20 Most Frequent Modifiers of COVID

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

```
[word="(?i)covid(-|\s)?(\d+)?"] [tag="N.*"]
```

Include the snapshot of those words:



(19,268 items, 2,256,498 total frequency)

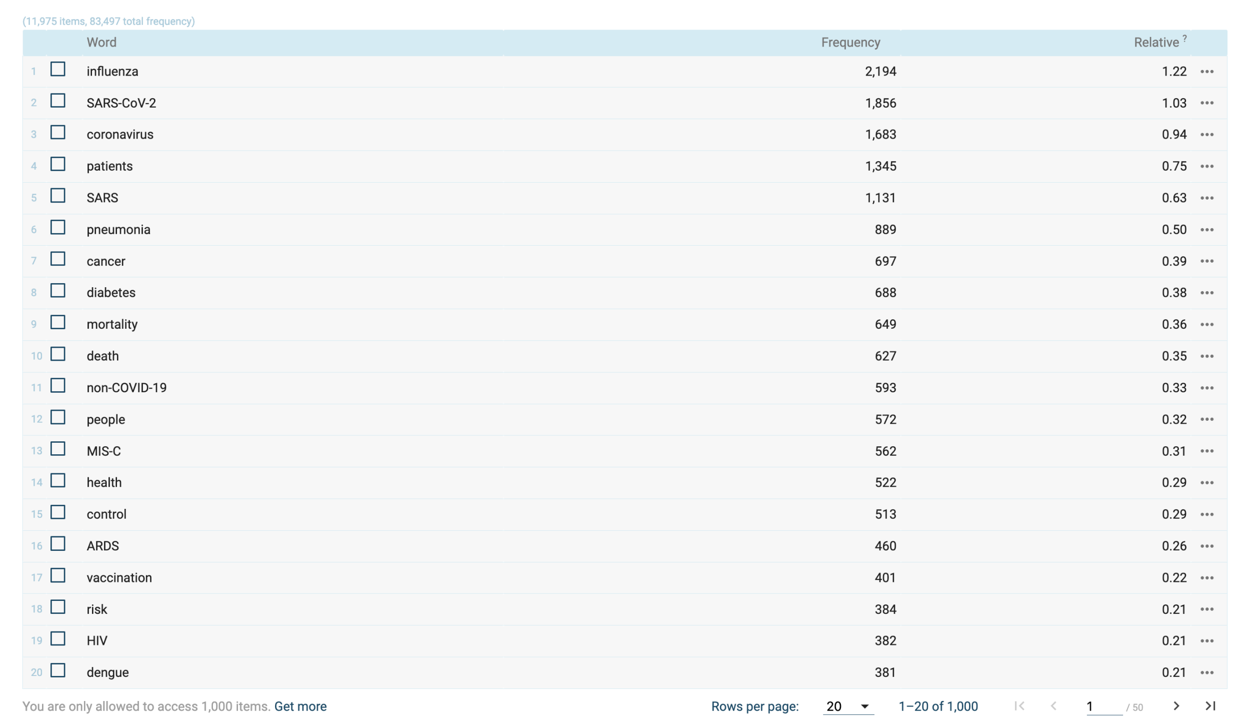
		Word	Frequency	Relative ?	
1	☐	pandemic	488,655	272.30	•••
2	☐	patients	325,972	181.65	•••
3	☐	infection	142,806	79.58	•••
4	☐	cases	106,538	59.37	•••
5	☐	vaccine	67,565	37.65	•••
6	☐	outbreak	59,394	33.10	•••
7	☐	disease	46,089	25.68	•••
8	☐	vaccination	38,590	21.50	•••
9	☐	vaccines	37,396	20.84	•••
10	☐	pneumonia	36,799	20.51	•••
11	☐	crisis	27,514	15.33	•••
12	☐	epidemic	25,882	14.42	•••
13	☐	symptoms	23,973	13.36	•••
14	☐	infections	20,410	11.37	•••
15	☐	diagnosis	19,844	11.06	•••
16	☐	severity	19,317	10.76	•••
17	☐	lockdown	18,516	10.32	•••
18	☐	mortality	18,505	10.31	•••
19	☐	case	16,502	9.20	•••
20	☐	transmission	15,827	8.82	•••

Figure 3: Top 20 Most Frequent Words Modified by COVID

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).

```
[tag="N.*" & (word !="(?i)covid(-|\s)?(\d+)?('s)?")] within
[word="(?i)covid(-|\s)?(\d+)?"] ([tag="CC" | word=","][(tag="N.*")]){0,9999}
```

Include the snapshot of those words:



(11,975 items, 83,497 total frequency)

		Word	Frequency	Relative ?
1	□	influenza	2,194	1.22
2	□	SARS-CoV-2	1,856	1.03
3	□	coronavirus	1,683	0.94
4	□	patients	1,345	0.75
5	□	SARS	1,131	0.63
6	□	pneumonia	889	0.50
7	□	cancer	697	0.39
8	□	diabetes	688	0.38
9	□	mortality	649	0.36
10	□	death	627	0.35
11	□	non-COVID-19	593	0.33
12	□	people	572	0.32
13	□	MIS-C	562	0.31
14	□	health	522	0.29
15	□	control	513	0.29
16	□	ARDS	460	0.26
17	□	vaccination	401	0.22
18	□	risk	384	0.21
19	□	HIV	382	0.21
20	□	dengue	381	0.21

You are only allowed to access 1,000 items. Get more

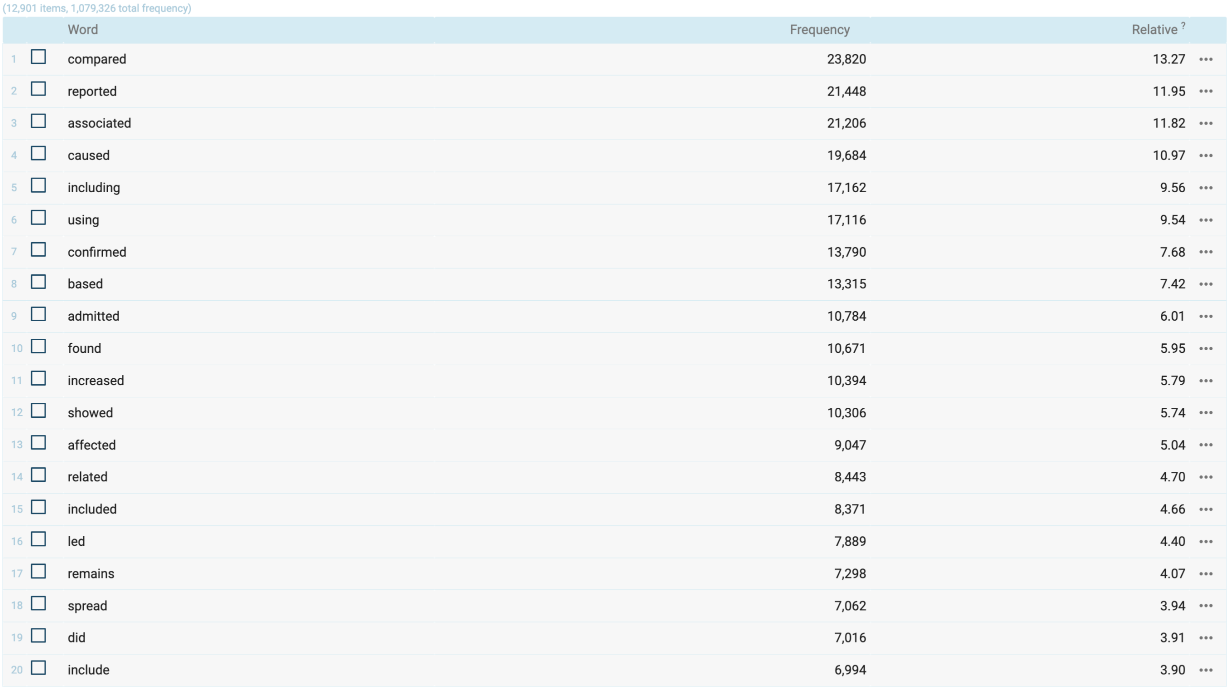
Rows per page: 20 ▾ 1–20 of 1,000 |‹ ‹ 1 / 50 › ›|

Figure 4: Top 20 Most Frequent Right Conjuncts of COVID

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

```
[word="(?i)covid(-|\s)?(\d+)?"][]{0,2}[(tag != "VH.* | VB.*") & (tag = "VV.*")]
```

Include the snapshot of verbs that take Covid as subject:



(12,901 items, 1,079,326 total frequency)

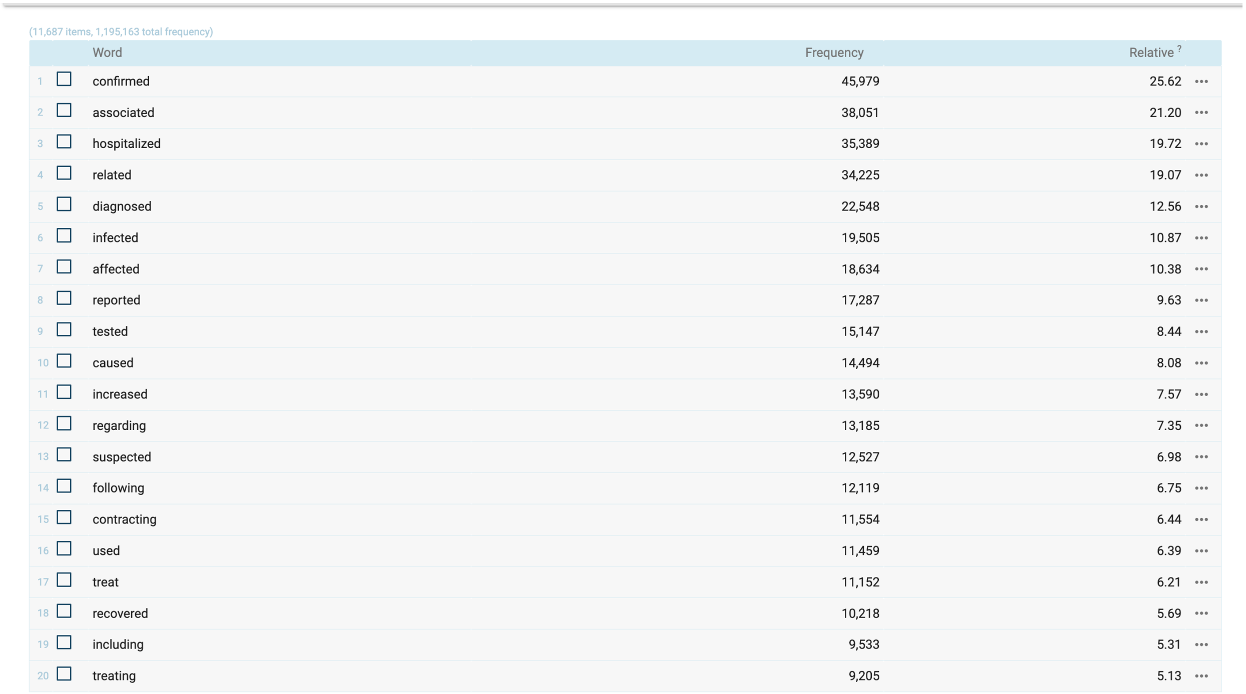
		Word	Frequency	Relative ?	
1	□	compared	23,820	13.27	•••
2	□	reported	21,448	11.95	•••
3	□	associated	21,206	11.82	•••
4	□	caused	19,684	10.97	•••
5	□	including	17,162	9.56	•••
6	□	using	17,116	9.54	•••
7	□	confirmed	13,790	7.68	•••
8	□	based	13,315	7.42	•••
9	□	admitted	10,784	6.01	•••
10	□	found	10,671	5.95	•••
11	□	increased	10,394	5.79	•••
12	□	showed	10,306	5.74	•••
13	□	affected	9,047	5.04	•••
14	□	related	8,443	4.70	•••
15	□	included	8,371	4.66	•••
16	□	led	7,889	4.40	•••
17	□	remains	7,298	4.07	•••
18	□	spread	7,062	3.94	•••
19	□	did	7,016	3.91	•••
20	□	include	6,994	3.90	•••

Figure 5: Top 20 Most Frequent Verbs that take COVID as subject

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

```
[(tag ="V.*")&(tag!="VB.*|VH.*")][]{0,2}[word="(?i)covid(-|\s)?(\d+)?"]
```

Include the snapshot of verbs that take Covid as object:



(11,687 items, 1,195,163 total frequency)

		Word	Frequency	Relative ?	
1	□	confirmed	45,979	25.62	•••
2	□	associated	38,051	21.20	•••
3	□	hospitalized	35,389	19.72	•••
4	□	related	34,225	19.07	•••
5	□	diagnosed	22,548	12.56	•••
6	□	infected	19,505	10.87	•••
7	□	affected	18,634	10.38	•••
8	□	reported	17,287	9.63	•••
9	□	tested	15,147	8.44	•••
10	□	caused	14,494	8.08	•••
11	□	increased	13,590	7.57	•••
12	□	regarding	13,185	7.35	•••
13	□	suspected	12,527	6.98	•••
14	□	following	12,119	6.75	•••
15	□	contracting	11,554	6.44	•••
16	□	used	11,459	6.39	•••
17	□	treat	11,152	6.21	•••
18	□	recovered	10,218	5.69	•••
19	□	including	9,533	5.31	•••
20	□	treating	9,205	5.13	•••

Figure 6: Top 20 Most Frequent Verbs that take COVID as object

Problem 2.3

For this corpus, the LogDice score appears to be the most effective metric for identifying and ranking collocations The ranking from Mutual Information (MI) score is unique but prioritizes rare word combinations ($\leq$ 10 co-occurrences). This characteristic of MI can lead to highlighting less frequent, therefore potentially less relevant collocations in the context of a prevalent and significant term like “COVID.”

The T-Score, places emphasis on more common words such as “of” In this specific corpus, these common words provide relatively minimal informational value about the unique linguistic patterns associated with COVID-19.

The LogDice score offers a more balanced result. It appear to effectively normalize the frequency of word pairs and addresses the biases towards extremely rare or extremely common words. This produced good results, where overly common words and overly scarse words are ranked lower. Among all three rankings, the LogDice score gives more nuanced and contextually relevant ranking of collocations that balances statically significance and content richness well.

Problem 3

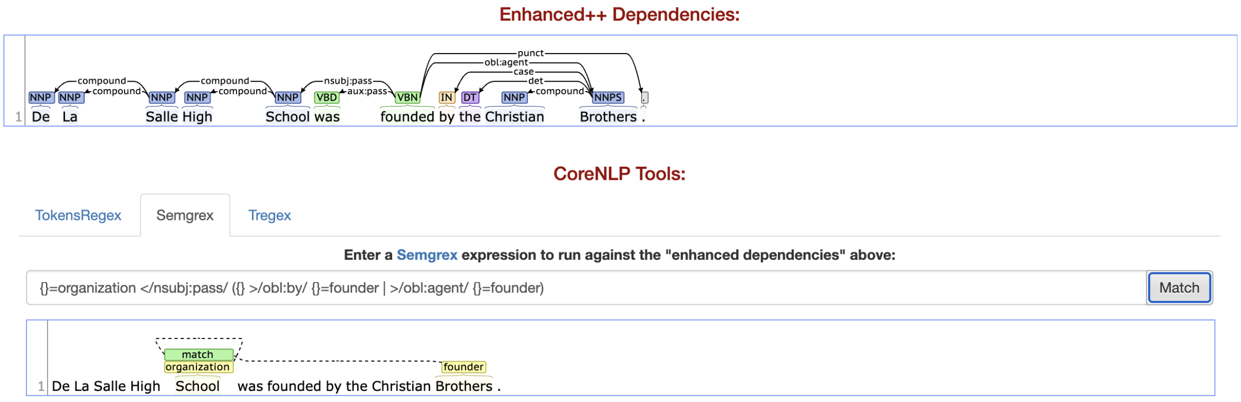
Problem 3.1

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

Semgrex:

```
{}=organization </nsubj:pass/ ({} >/obl:by/ {}=founder | >/obl:agent/ {}=founder)
```

Result:



Enhanced++ Dependencies:

punct
obl:agent
case
det
compound
compound
compound
compound
compound
nsubj:pass
aux:pass
NNP NNP NNP NNP NNP VBD VBN IN DT NNP NNPS .
1 De La Salle High School was founded by the Christian Brothers .

CoreNLP Tools:

TokensRegex
Semgrex
Tregex

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

{}=organization </nsubj:pass/ ({} >/obl:by/ {}=founder | >/obl:agent/ {}=founder)

Match
match
organization
founder
1 De La Salle High School was founded by the Christian Brothers .

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

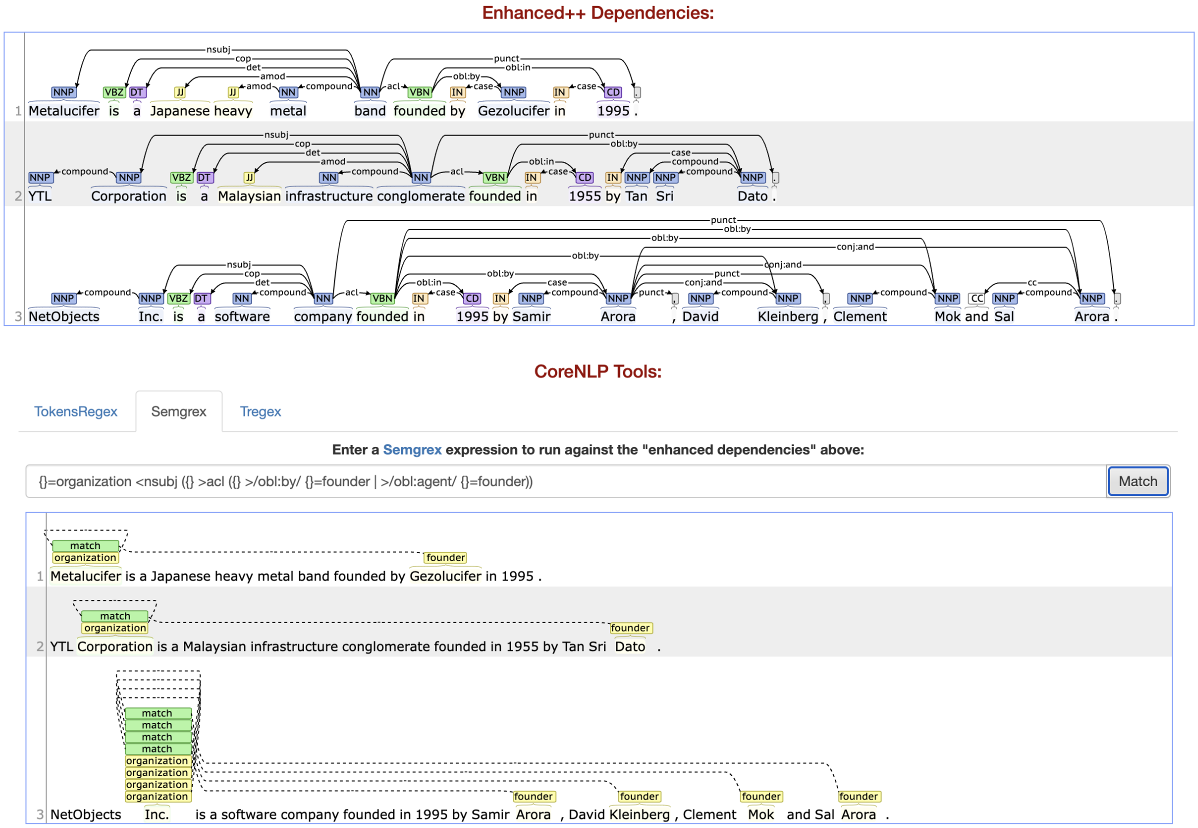
YTL Corporation is a Malaysian infrastructure conglomerate founded in 1955 by **Tan Sri Dato**.

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

Semgrex:

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

Result:



Enhanced++ Dependencies:

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

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

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

CoreNLP Tools:

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

{}=organization <nsubj ({} >acl ({} >/obl:by/ {}=founder | >/obl:agent/ {}=founder))

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

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

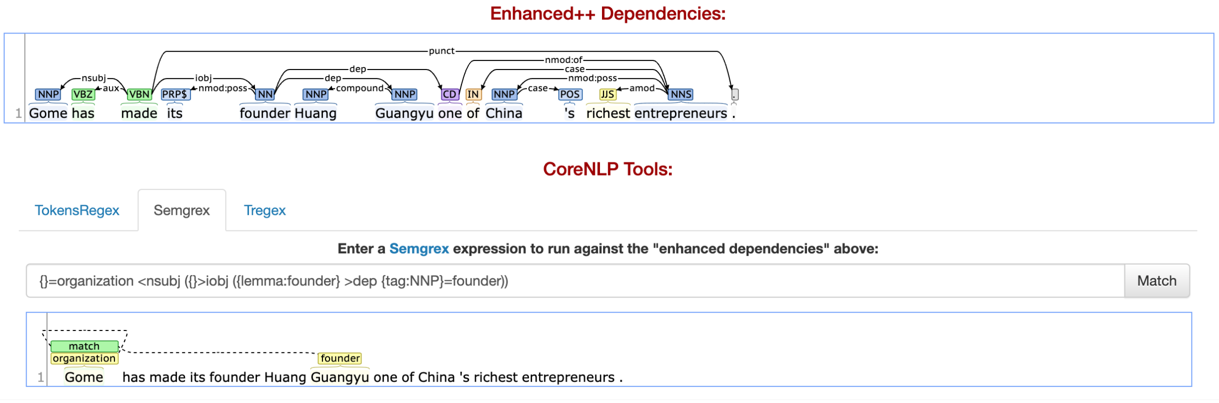
3 NetObjects Inc. is a software company founded in 1995 by Samir Arora , David Kleinberg , Clement Mok and Sal Arora .

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

Semgrex:

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

Result:



Enhanced++ Dependencies:

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

CoreNLP Tools:

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

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

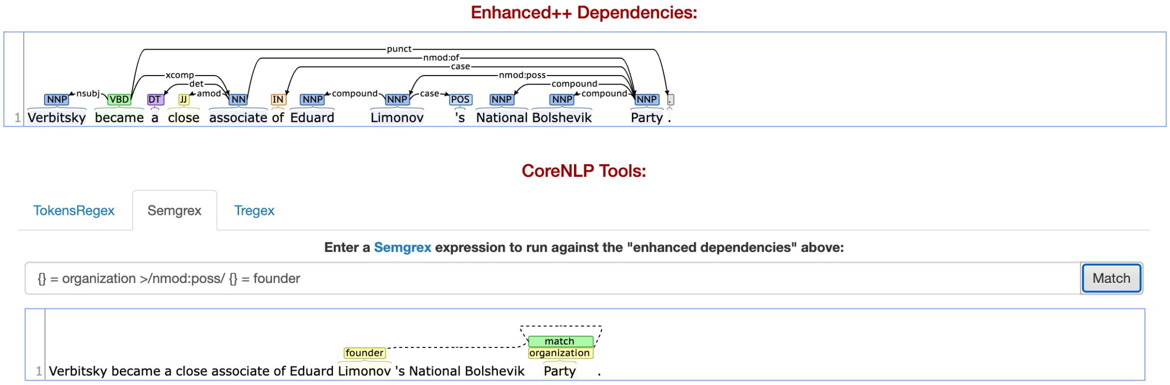
Gome has made its founder Huang Guangyu one of China 's richest entrepreneurs .

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

Semgrex:

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

Result:



Enhanced++ Dependencies:

NNP
VBD
DT
JJ
NN
IN
NNP
NNP
POS
NNP
NNP
NNP
.
nsubj
xcomp
det
amod
punct
nmod:of
case
compound
case
nmod:poss
compound
compound
1 Verbitsky became a close associate of Eduard Limonov 's National Bolshevik Party .

CoreNLP Tools:

TokensRegex
Semgrex
Tregex

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

{} = organization >/nmod:poss/ {} = founder

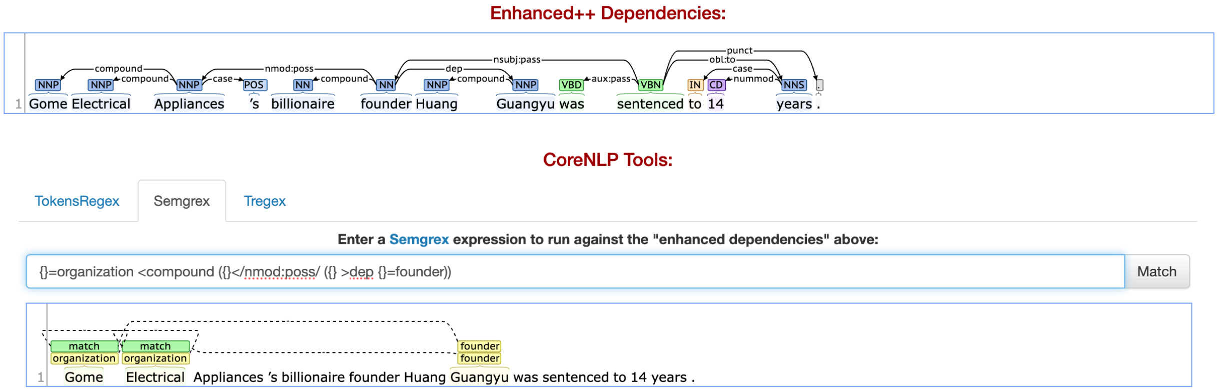
Match
founder
match
organization
1 Verbitsky became a close associate of Eduard Limonov 's National Bolshevik Party .

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

Semgrex:

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

Result:



Enhanced++ Dependencies:

compound
compound
case
nmod:poss
compound
dep
compound
nsubj:pass
aux:pass
punct
obl:to
case
nummod
NNP NNP NNP POS NN NN NNP NNP VBD VBN IN CD NNS .
1 Gome Electrical Appliances 's billionaire founder Huang Guangyu was sentenced to 14 years .

CoreNLP Tools:

TokensRegex
Semgrex
Tregex

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

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

Match
match
organization
match
organization
founder
founder
1 Gome Electrical Appliances 's billionaire founder Huang Guangyu was sentenced to 14 years .