Knowledge Graph Embedding & KG enrichment

LGSI, Banglore

Progress from Anindya

09-Nov-2020 onwards

26-Feb-2021

Activities Done

- Improvement and debugging on xml to dict

Pending Action Items

ToDo

- Need to test, the extraction system

DEBUGGING:
Analysis: Mismatch between the PDF data and extracted data:

"#text": "Follow package or container instructions for proper freezing methods."
Line 7340: "#text": "Do not use"
Line 7352: "#text": "Rigid plastic containers with tight-fitting lids"
Line 7360: "#text": "Straight-sided canning/freezing jars"
Line 7368: "#text": "Heavy-duty aluminum foil"
Line 7376: "#text": "Plastic-coated paper"
Line 7384: "#text": "Non-permeable plastic wraps"
Line 7392: "#text": "Specified freezer-grade self-sealing plastic bags"
Line 7405: "#text": "Bread wrappers"
Line 7413: "#text": "Non-polyethylene plastic containers"
Line 7421: "#text": "Containers without tight lids"
Line 7429: "#text": "Wax paper or wax-coated freezer wrap"
Line 7437: "#text": "Thin, semi-permeable wrap"

Packaging recommendations

- Rigid plastic containers with tight-fitting lids
- Straight-sided canning/freezing jars
- Heavy-duty aluminum foil
- Plastic-coated paper
- Non-permeable plastic wraps
- Specified freezer-grade self-sealing plastic bags

Follow package or container instructions for proper freezing methods.

Do not use

- Bread wrappers
- Non-polyethylene plastic containers
- Containers without tight lids
- Wax paper or wax-coated freezer wrap
- Thin, semi-permeable wrap

Reason: XMLtodict combines two tags(listitem) with same name inside a dictionary into a list, resulting in wrong placement of the data

Solution: Most of the tags have ids in the xml tag, utilize the ids to create unique tags,
Eg. listitem -> listitem_id_t_00000096_r1_20181224152325789_80

Unique tags ensure tags are not merged hence avoiding wrong placement of data block

After changes:

"#text": "Packaging recommendations"

Line 7378: "#text": "Rigid plastic containers with tight-fitting lids"
Line 7386: "#text": "Straight-sided canning/freezing jars"
Line 7394: "#text": "Heavy-duty aluminum foil"
Line 7402: "#text": "Plastic-coated paper"
Line 7410: "#text": "Non-permeable plastic wraps"
Line 7418: "#text": "Specified freezer-grade self-sealing plastic bags"
Line 7425: "#text": "Follow package or container instructions for proper freezing methods."
Line 7430: "#text": "Do not use"
Line 7439: "#text": "Bread wrappers"
Line 7447: "#text": "Non-polyethylene plastic containers"
Line 7455: "#text": "Containers without tight lids"
Line 7463: "#text": "Wax paper or wax-coated freezer wrap"
Line 7471: "#text": "Thin, semi-permeable wrap"

25-Feb-2021

Activities Done

- Both the changes coref and discourse changes are integrated into the pipeline

Pending Action Items

ToDo

- Need to test, the extraction system

Allen Coref and Improvement on the system are integrated into the pipeline.

However to test the system, significant memory is required by both the OLLIE and IE tool(10GB).

.95 server memory error

OSError: [Errno 12] Cannot allocate memory

There is insufficient memory for the Java Runtime Environment to continue.

Native memory allocation (mmap) failed to map 758300672 bytes for committing reserved memory.

An error report file with more information is saved as:

/home/lg/Anindya/dev/kg/New_IE_integrated/hs_err_pid21953.log

24-Feb-2021

Activities Done

- Worked on AllenNLP based coref module

Pending Action Items

ToDo

- Integrate into the pipeline

Enter the text
This section talks about Ice Plus. This function increases both ice making and freezing capabilities.
This section talks about **Ice Plus**. **Ice Plus** increases both ice making and freezing capabilities.

Enter the text
Putting food in the refrigerator before it has cooled could cause the food to spoil, or a bad odor to remain inside the refrigerator.
Putting **food** in the refrigerator before **food** has cooled could cause food to spoil, or a bad odor to remain inside the refrigerator.

Enter the text
Press the Ice Plus button to illuminate the icon and activate the function for 24 hours.
Press the Ice Plus button to illuminate the icon and activate the function for 24 hours.

Enter the text
This section talks about Smart Grid. Press the Smart Grid button to turn the function On/Off. When the function is on, the icon illuminates. The function automatically turns on when the refrigerator is connected to the Wi-Fi network.
This section talks about Smart Grid. Press the Smart Grid button to turn the function On / Off. When the function is on, the icon illuminates. the function automatically turns on when the refrigerator is connected to the Wi-Fi network.

Enter the text
Ice is made in the automatic in-door icemaker and sent to the dispenser. The icemaker produces 70 - 182 cubes in a 24-hour period, depending on freezer compartment temperature, room temperature, number of door openings and other operating conditions. It takes about 12 to 24 hours for a newly installed refrigerator to begin making ice.

Ice is made in the **automatic in-door icemaker** and sent to the dispenser. the **automatic in-door icemaker** produces 70-182 cubes in a 24-hour period, depending on freezer compartment temperature, room temperature, number of door openings and other operating conditions. It takes about 12 to 24 hours for a newly installed refrigerator to begin making ice.
Enter the text
Carefully insert the in-door ice bin, slanting the top slightly to fit it under the icemaker.
Carefully insert the **in-door ice bin**, slanting the top slightly to fit the **in-door ice bin** under the icemaker.

23-Feb-2021

Activities Done

- Extracted relations had some issues(fixed)

Pending Action Items

ToDo

- AllenNlp coref model integration

The following relations are wrongly extracted:

'Do not apply excessive force while detaching or assembling the storage bins.'
Rel:
Do not apply excessive force Alternative while detaching assembling the storage bins

'Never use a glass that is exceptionally narrow or deep.'

REL:
Never use a glass Alternative that is exceptionally narrow deep

Turn off the icemaker if the refrigerator is not yet connected to the water supply.

PDTB manual:
https://www.cis.upenn.edu/~pdtb/PDTBAPI/pdtb-annotation-manual.pdf page 75

Based on the following relations are mapped:

while ---> Synchronous
when ----> Synchronous

Some relations are extracted had empty tails:

With either refrigerator door opened, press and hold the Refrigerator and Ice Plus buttons at the same time for 5 seconds.

either refrigerator door opened Conjunction

After replacing the air filter, press and hold the Fresh Air Filter button for three seconds to turn the icon light off.

replacing the water filter Conjunction

Storage times will vary according to the quality and type of food, the type of packaging or wrap used (how airtight and moisture-proof) and the storage temperature.

Storage times will vary according to the quality and type of food , the type of packaging or wrap used and the storage temperature Conjunction how airtight moisture-proof

16-Feb-2021

Activities Done

- Manual data formatted in CeSi input file format

Pending Action Items

ToDo

- Graph Creation

From Manual data input file created in CeSI format (in progress). Triple norm: Spacy for lemmatize, lower case, Triples, entity link, KBP information (["PERSON", "per:city_of_birth", "LOCATION"]) (in progress)

Observation:

1. Entity linking is NULL for most entities
2. KBP information is also empty.

Data Analysis :

1. Stanford corenlp coref module is not able to correctly do coref, allennlp is better.
   Eg.
   Ice Plus
   This function increases both ice making and freezing capabilities.
   Press the Ice Plus button to illuminate the icon and activate the function for 24 hours. The function automatically shuts off after 24 hours. Stop the function manually by pressing the button once more.

   The refrigerator makes a loud noise after initial operation.
   This is normal. The volume will decrease as the temperature decreases.

Integrate allennlp coref to the pipeline.

Rewrite manuals?--> Declarative sentences

Hierarchical label -->what is acceptable(‘start’, ‘stop’)

Property in terms of dictionary

Ice Plus
This function increases both ice making and freezing capabilities.

Ice Plus
This function increases both ice making and freezing capabilities.
Press the Ice Plus button to illuminate the icon and activate the function for 24 hours. The function automatically shuts off after 24 hours. Stop the function manually by pressing the button once more.

This section talks about Ice Plus.
This function increases both ice making and freezing capabilities.

coreference:bad output

The refrigerator makes a loud noise after initial operation.
This is normal. The volume will decrease as the temperature decreases.

allen SRL better

10-Feb-2021

Activities Done

- Graph Embedding (CaRE paper) for OPenKG, reproduce author's results (in progress)

Pending Action Items

ToDo

- Graph Embedding CaRe paper

12-Feb-2021

Activities Done

- Second KT sharing on Graph creation and embedding.

Pending Action Items

ToDo

- On Manual data to input format

CaRE


```
  File "CaRe_main.py", line 211, in <module>
    main(args)
  File "CaRe_main.py", line 92, in main
    model = TransEParam(args,data.embed_matrix,data.rel2word)
  File "CaRe_main.py", line 16, in __init__
    super(ConvEParam, self).__init__()
NameError: name 'ConvEParam' is not defined
```

Soln: typos in author provided code, it should be TransEParam

```
File "CaRe_main.py", line 211, in <module>
    main(args)
  File "CaRe_main.py", line 86, in main
    data = load_data(args)
  File "/home/anindya.sundardas/Development/CaRE/CaRe(B=TransE)/data.py", line 10, in __init__
    self.fetch_data()
  File "/home/anindya.sundardas/Development/CaRE/CaRe(B=TransE)/data.py", line 126, in fetch_data
    self.get_edges(self.canon_clusts)
  File "/home/anindya.sundardas/Development/CaRE/CaRe(B=TransE)/data.py", line 105, in get_edges
    if self.args.model_variant=='CaRe' and len(ent_clusts[ent])==1:
```

~~Soln: There is no such argument as model_variant, seems like, authors did not update the working module.~~

```
#if self.args.model_variant=='CaRe' and len(ent_clusts[ent])==1:
```

```
  utilizing pre-trained word embeddings
Traceback (most recent call last):
  File "CaRe_main.py", line 211, in <module>
    main(args)
  File "CaRe_main.py", line 100, in main
    optimizer = torch.optim.SGD(model.parameters(), lr=params["lr"], nesterov=True, momentum=0.9,
dampening=0)
NameError: name 'params' is not defined

No dict with such name, define params=vars(args)

File "CaRe_main.py", line 233, in <module>
    main(args)
  File "CaRe_main.py", line 156, in main
    loss = model.get_loss(samples,edges,node_id)
  File "CaRe_main.py", line 98, in get_loss
    score = self.get_score(sub_embed, obj_embed, rel_embed)
  File
"/home/anindya.sundardas/miniconda3/envs/py368/lib/python3.6/site-packages/torch/nn/modules/module.py", line
539, in __getattr__
    type(self).__name__, name))
AttributeError: 'TransEParam' object has no attribute 'get_score'
```

Soln:
There is not get_score function defined, from analysis of transE algorithm it seems like, the actual function name should be self._calc not self.get_score

```
File "CaRe_main.py", line 211, in <module>
    main(args)
  File "CaRe_main.py", line 86, in main
    data = load_data(args)
  File "/home/anindya.sundardas/Development/CaRE/CaRe(B=TransE)/data.py", line 10, in __init__
    self.fetch_data()
  File "/home/anindya.sundardas/Development/CaRE/CaRe(B=TransE)/data.py", line 130, in fetch_data
    self.entid2clustid)
ValueError: too many values to unpack (expected 3)
```

Soln:
The function returns one argument.
Comment out

```
#self.train_trips,self.rel2id,self.label_graph = self.get_train_triples(self.data_files["train_trip_path"],
                                                                        self.entid2clustid)

self.train_trips= self.get_train_triples(self.data_files["train_trip_path"],
                                                                        self.entid2clustid)
```

CaRE

```
if self.args.CN=='LAN':
   self.cn = CaRe(self.args.nfeats, self.args.nfeats)
elif self.args.CN=='GCN':
   self.cn = CaReGCN(self.args.nfeats, self.args.nfeats)
else:
     self.cn = CaReGAT(self.args.nfeats, self.args.nfeats//self.args.nheads,
  heads=self.args.nheads, dropout=self.args.dropout)

Traceback (most recent call last):
  File "CaRe_main.py", line 237, in <module>
    main(args)
  File "CaRe_main.py", line 114, in main
    model = ConvEParam(args,data.embed_matrix,data.rel2word)
  File "CaRe_main.py", line 23, in __init__
    self.cn = CaRe(self.args.nfeats, self.args.nfeats)
  NameError: name 'CaRe' is not defined
```

Soln: Redefine as mentioned in TransE part, as the underlying algorithm is same , inspite of what the base model is:

```
if self.args.CN=='LAN':
   self.cn = LAN(self.args.nfeats, self.args.nfeats)
elif self.args.CN=='GCN':
   self.cn = GCN(self.args.nfeats, self.args.nfeats)
else:
     self.cn = GAT(self.args.nfeats, self.args.nfeats//self.args.nheads,
  heads=self.args.nheads, dropout=self.args.dropout)
```

Code Run Log for TRansE: 234 epochs completed

Mean Rank: Head = 1894.5911617565314 Tail = 3174.0769872151195 Avg = 2534.3340744858256

MRR: Head = 0.1423031476867858 Tail = 0.13235962564506312 Avg = 0.13733138666592448

Hits@10: Head = 0.25152862701500833 Tail=0.23846581434130074 Avg = 0.24499722067815452

Hits@30: Head = 0.30322401334074484 Tail=0.28515842134519176 Avg = 0.2941912173429683

Hits@50: Head = 0.3265703168426904 Tail=0.2998888271261812 Avg = 0.3132295719844358

Best Valid MRR: 0.13733138666592448, Best Valid MR: 2534.3340744858256, Best Epoch: 230

epoch 231/500 total epochs, epoch_loss: 0.27973644599795766

epoch 232/500 total epochs, epoch_loss: 0.2796595214312611

epoch 233/500 total epochs, epoch_loss: 0.2793956553808735

epoch 234/500 total epochs, epoch_loss: 0.2793349908976368

05-Feb-2021

Activities Done

- Stanford Corenlp entity linking.
- Sample entities are linked to wikidata

Pending Action Items

ToDo

- Recreate clustering for a given dataset

CoreNLP Entity Link

Download the jar file with core wikidict.tab.gz. Put it inside the path folder containing all jar files

[pool-1-thread-2] INFO edu.stanford.nlp.pipeline.WikidictAnnotator - Loaded 11000000 entries from Wikidict [2937MB memory used; 45.513 seconds elapsed]
[pool-1-thread-2] INFO edu.stanford.nlp.pipeline.WikidictAnnotator - Loaded 12000000 entries from Wikidict [3008MB memory used; 56.153 seconds elapsed]
[pool-1-thread-2] INFO edu.stanford.nlp.pipeline.WikidictAnnotator - Loaded 13000000 entries from Wikidict [3191MB memory used; 01:07.011 minutes elapsed]
[pool-1-thread-2] INFO edu.stanford.nlp.pipeline.WikidictAnnotator - Loaded 14000000 entries from Wikidict [3314MB memory used; 01:28.286 minutes elapsed]

java.lang.OutOfMemoryError:

Taking **a lot more time** and stops after loading 14000000 entries, with out of memory error

Solution:
Allot more memory for java solves the problem.
nlp = StanfordCoreNLP(stanford_core_nlp_path, memory='8g', quiet = not verbose)

CoreNLP Entity Link

It is linked to 21 million entities from wikidata dict.

doc='India, officially the Republic of India, is a country in South Asia. It is the second-most populous country, the seventh-largest country by land area, and the most populous democracy in the world.'

```
for item in result['sentences'][0]['entitymentions']:
    print(item['text'],'-', item['entitylink']) #for only ner

##eg. output:
'''
India - India
Republic of India - India
South - Southern_United_States
Asia - Asia
'''
```

Entity linking for sample entries from Manual: Link (Excluding the sentences and phrases)

CoreNLP Entity Link

Comment:

1) No linking was found for most of the entities .
2) There are also wrong entries as well .

04-Feb-2021

Activities Done

- PPT updated with APPENDIX and shared
- Entity Linking stanford corenlp(Progress)

Pending Action Items

ToDo

- Recreate clustering for a given dataset

CoreNLP Entity Link

{"triple_norm": ["frederick", "have reach", "alessandria"], "true_link": {"object": "/m/02bb_4", "subject": "/m/09w_9"}, "src_sentences": ["Frederick had reached Alessandria", "By late October, Frederick had reached Alessandria."], "triple": ["Frederick", "had reached", "Alessandria"], "kbp_info": [], "entity_linking": {"object": "Alessandria", "subject": "Frederick,_Maryland"}, "_id": 36952}

Analysis:

While true link is for evaluation, entity_link is used by cesi to compute the clusters.

Code is not available which mentions how entity_link is obtained.

Paper mentions entity_link is obtained from stanford core nlp.

Entity link is done by comparison with a dict on wikidict

English (kbp) model downloaded Model-kbp.jar downloaded which contains wikidict

Python wrapper for corenlp for entity linking is not working

CoreNLP Entity Link

Eg. doc="*India, officially the Republic of India, is a country in South Asia. It is the second-most populous country, the seventh-largest country by land area, and the most populous democracy in the world.*"

```
nlp = StanfordCoreNLP(stanford_core_nlp_path, quiet =  not verbose)
props = {'annotators': 'entitylink', 'pipelineLanguage': 'en'}
annotated = nlp.annotate(doc, properties=props)
```

```
Caused by: class java.io.IOException: Unable to open
"edu/stanford/nlp/models/kbp/english/wikidict.tab.gz" as class path, filename or URL

edu.stanford.nlp.io.IOUtils.getInputStreamFromURLOrClasspathOrFileSystem(IOUtils.java
:501)
  edu.stanford.nlp.io.IOUtils$GetLinesIterable.getStream(IOUtils.java:767)
  edu.stanford.nlp.io.IOUtils$GetLinesIterable.access$000(IOUtils.java:736)
  edu.stanford.nlp.io.IOUtils$GetLinesIterable$1.getReader(IOUtils.java:811)
```

03-Feb-2021

Activities Done

- Setup of Cesi(done)
- Executed for author provided dataset. Results reported

Pending Action Items

ToDo

- Recreate clustering for a given dataset

Steps to solve error:

1)Install mariadb

2) sudo apt-get install -y libmariadbclient-dev (installed by sysadmin)

3)pip install pattern

4)environment should be python 3.6

5)pip install -r requirement.txt

6)use numpy version 1.16.0

7)run setup.sh (Make sure glove embeddings have all the 400000 vectors, else NaN values might appear)

CeSi


```
2021-02-02 17:31:55,901 - [INFO] - Running reverb45_test_run
2021-02-02 17:31:55,901 - [INFO] - Reading Triples
2021-02-02 17:32:19,182 - [INFO] -      Cached triples
2021-02-02 17:32:19,415 - [INFO] - Side Information Acquisition
2021-02-02 17:32:27,464 - [INFO] -      Entity Linking Side info
2021-02-02 17:32:27,571 - [INFO] -      PPDB Side info
2021-02-02 17:32:27,821 - [INFO] -      Word Sense Disamb Side info
2021-02-02 17:32:36,212 - [INFO] -      Morphological Normalization Side info
2021-02-02 17:32:40,088 - [INFO] -      Token Overlap Side info
2021-02-02 17:32:54,223 - [INFO] -      AMIE Side info
2021-02-02 17:33:31,665 - [INFO] -      KBP Side info
2021-02-02 17:33:36,222 - [INFO] -      Cached Side Information
2021-02-02 17:33:36,222 - [INFO] - Embedding NP and relation phrases
2021-02-02 17:36:55,815 - [INFO] -      Epochs: 1
2021-02-02 17:38:36,173 - [INFO] -      Epochs: 2
2021-02-02 17:40:16,314 - [INFO] -      Epochs: 3
2021-02-02 17:41:56,101 - [INFO] -      Epochs: 4
2021-02-02 17:43:36,483 - [INFO] -      Epochs: 5
2021-02-02 17:45:16,341 - [INFO] -      Epochs: 6
2021-02-02 17:46:56,118 - [INFO] -      Epochs: 7
2021-02-02 17:48:35,891 - [INFO] -      Epochs: 8
2021-02-02 17:50:16,180 - [INFO] -      Epochs: 9
2021-02-02 17:51:55,807 - [INFO] -      Epochs: 10
2021-02-02 17:51:56,183 - [INFO] - Clustering NPs and relation phrases
2021-02-02 17:52:48,983 - [INFO] - NP Canonicalizing Evaluation
2021-02-02 17:52:49,534 - [INFO] - Macro F1: 0.6252, Micro F1: 0.844, Pairwise F1: 0.8204
2021-02-02 17:52:49,535 - [INFO] - CESI: #Clusters: 6234, #Singletons 1776
2021-02-02 17:52:49,536 - [INFO] - Gold: #Clusters: 6039, #Singletons 186
```

Ent_cluster: https://drive.google.com/file/d/1VLiD5xQZxYnT-qNkjjornqa4Dg2OtzXv/view?usp=sharing
Rel_cluster: https://drive.google.com/file/d/1H6ZSTefMUavv6s9UrOfsSyZV7Nyf0q8J/view?usp=sharing

02-Feb-2021

Activities Done

- Setup of Cesi

Pending Action Items

ToDo

- Recreate clustering for a given dataset

Issue:
scipy==0.19.0
1) raise NotFoundError('no lapack/blas resources found')
numpy.distutils.system_info.NotFoundError: no lapack/blas resources found

Requirements can’t get installed.

Resolve:
1)Downgraded from python 3.7 to python 3.6

Separate environment created and Requirements were installed successfully.
However it needs a package dependency pattern.

Issue:
scipy==0.19.0
1) raise NotFoundError('no lapack/blas resources found')
numpy.distutils.system_info.NotFoundError: no lapack/blas resources found

Requirements can’t get installed.

Resolve:
1)Downgraded from python 3.7 to python 3.6

Separate environment created and Requirements were installed successfully.
However it needs a package dependency pattern.

01-Feb-2021

Activities Done

- Research Paper on Canonicalization of entities(CeSi)

Pending Action Items

ToDo

- Recreate clustering for a given dataset

CeSi

Approach:

1)Side information acquisition:

- **Entity Linking:**Stanford CoreNLP Entity Linker
- **PPDB:**PPDB 2.0 , a large collection of paraphrases in English, for identifying equivalence relation among NPs.
- WordNet: Identify possible Synsets
- IDF TOKEN Overlap: NPs sharing infrequent terms give a strong indication of them referring to the same entity. For example, it is very likely for Warren Buffett and Buffett to refer to the same person.
- Morph Normalization: Tense, Pluralization, Capitalization.

2) **Embedding:** Ho1E algorithm

3)**CLUSTERING EMBEDDINGS AND CANONICALIZATION:** HAC as number of clusters are not known before hand.

29-Jan-2021

Activities Done

- Analysis of code for Open KG embedding in progress

Pending Action Items

ToDo

- Research Paper on Canonicalization of entities(CeSi)

21-Jan-2021

Activities Done

- Discourse Relations are integrated to current system

Pending Action Items

ToDo

- Integrate to current pipeline

20-Jan-2021

Activities Done

- Complex and Compound sentences are broken down to constituent clauses (13 Cases)
- Connectives are converted to Discourses Relation(completed)

Pending Action Items

ToDo

- Integrate to current pipeline

Clause Extraction(ASER)

- All 13 types of relations are covered (to break sentences into clauses):
- Case 1: Precedence

  I eat apple, before I go to swim.

  ['before', 'I go to swim', 'I eat apple ,']

  relations: [('I eat apple ,', 'Precedence', 'I go to swim')]
- Case 2 Succession

  I eat apple, after I go to swim.

  ['after', 'I go to swim', 'I eat apple ,']

  relations: [('I eat apple ,', 'Succession', 'I go to swim')]
- case 3: Synchronous

  I eat apple, meantime I go to swim.

  ['meantime', 'I go to swim', 'I eat apple']

  relations: [('I eat apple', 'Synchronous', 'I go to swim')]
- Case4: Reason

  I eat apple, because I go to swim.

  ['because', 'I go to swim', 'I eat apple ,']

  relations: [('I eat apple ,', 'Reason', 'I go to swim')]

Clause Extraction(ASER)

- All 13 types of relations are covered (to break sentences into clauses):
- Case 5: Result

  I eat apple, so that I go to swim.

  ['so', 'that I go to swim', 'I eat apple ,']

  relations: [('I eat apple ,', 'Result', 'that I go to swim')]
- Case 2 Condition

  I eat apple, if I go to swim.

  ['if', 'I go to swim', 'I eat apple ,']

  relations: [('I eat apple ,', 'Condition', 'I go to swim')
- case 7: Contrast

  I go to swim, but I eat apple

  ['but', 'I eat apple', 'I go to swim']

  relations: [('I go to swim', 'Contrast', 'I eat apple')]
- case 8: Concession

  I like to swim, although I hate water.

  ['although', 'I hate water', 'I like to swim ,']

  relations: [('I like to swim ,', 'Concession', 'I hate water')]

Clause Extraction(ASER)

- All 13 types of relations are covered (to break sentences into clauses):
- Case 9: Conjunction

  I eat apple, and I like to go swimming.

  ['and', 'I like to go swimming', 'I eat apple']

  relations: [('I eat apple', 'Conjunction', 'I like to go swimming')]
- Case 10:Instantiation

  I like food, for example I love noodles.

  ['for example', 'I love noodles', 'I like food ,']

  relations: [('I like food ,', 'Instantiation', 'I love noodles')]
- case 11:Restatement

  I like Mathematics, in other words I like anything related to computation.

['in other words', 'I like anything related to computation', 'I like Mathematics,']
relations: [('I like Mathematics ,', 'Restatement', 'I like anything related to computation')]

- Case 12 Alternative:

  I eat apple, as an alternative I go to school.

  ['as an alternative', 'I go to school', 'I eat apple ,']

  relations: [('I eat apple ,', 'Alternative', 'I go to school')]

Clause Extraction(ASER)

- All 13 types of relations are covered (to break sentences into clauses):
- Case 13: Exception

  I do not go outside, except I go to school.

  ['except', 'I go to school', 'I do not go outside ,']

  relations: [('I do not go outside ,', 'Exception', 'I go to school')]

19-Jan-2021

Activities Done

- Convert discourse connectives to discourse relations (In progress)

Pending Action Items

ToDo

- Integrate to current pipeline

Clause Extraction(ASER)

- 9 types of relations are covered (to break sentences into clauses):
- Case 1: Precedence
  - I eat apple, before I go to swim.
  - I eat apple then I go to swim.
  - I eat apple, till I go to swim.
  - I eat apple, until I go to swim
- Case 2 Succession
  - I eat apple, after I go to swim
  - I eat apple, once I go to swim
- case 3: Synchronous
  - I eat apple, meantime I go to swim.
  - I eat apple, meanwhile I go to swim.
- Case4: Reason

  I eat apple, because I go to swim.
- Case 5: Result
  - I eat apple, so that I go to swim.

Discourse Relations

I eat apple, thus I go to swim.
I eat apple, therefore I go to swim.
I eat apple, so that I go to swim.

- Case 6:Condition
  I eat apple, if I go to swim.
  if I go to swim, I eat apple.
  when I go to swim, I eat apple.
- case 7: Contrast
  I go to swim, but I eat apple.
  I go to swim, however I eat apple.
  I eat apple, by contrast I go to school.
  I eat apple, in contrast I go to school.
  I eat apple, on the contrast I go to school.
- Case 8: Concession
  I like to swim, although I hate water.
- Case 9: Conjunction

I go to swim, and I eat apple.
I go to swim, also I eat apple.

18-Jan-2021

Activities Done

- OpenIE5.1 integrated to Current extraction pipeline

Pending Action Items

ToDo

- Clauses to Relations(ASER paper)

OpenIE 5.1

Different Mode of Running, for Batch processing the following method works best:

1)First Run the server locally:
java -Xmx10g -XX:+UseConcMarkSweepGC -jar openie-assembly-5.0-SNAPSHOT.jar -s -b --httpPort 8000

-b: Binary Relations/ Otherwise N-ary relations will be extracted
-s : Sentence-wise split

2)Run the python wrapper

Observations:
1)Duplicate relations are coming (solved)
2)Lesser relations are extracted
3) Does not work good for Imperative Sentences (A phenomenon observed in almost all IE, Dependency parse might help, but it also extracts unnecessary relations)

15-Jan-2021

Activities Done

- Set up OpenIE5.1

Pending Action Items

ToDo

- Integrate to current pipeline

OpenIE 5.1

Ollie used bootstrapped dependency parse.
It improves extractions from noun relations(RelNoun), numerical sentences(BONIE) and conjunctive sentences(CALMIE).

https://github.com/dair-iitd/OpenIE-standalone

OpenIE5.1 reqires huge memory, it actually depends on **Berkley Language** model

Berkeley Language Model :6.5GB
openie jar :1.8GB , pre-compiled jar for linux can be downloaded, or it can be compiled to create a new jar using instructions mentioned in the link.

Download **verbnet folder** from the project

OpenIE 5.1

java -Xmx10g -XX:+UseConcMarkSweepGC -jar openie-assembly-5.0-SNAPSHOT.jar

The U.S. president Barack Obama gave his speech on Tuesday and Wednesday to thousands of people.

List(0.92 (The U.S. president Barack Obama; gave; his speech; T:on Wednesday; to thousands of people), 0.38 (Barack Obama; [is] president [of]; United States), 0.92 (The U.S. president Barack Obama; gave; his speech; T:on Tuesday; to thousands of people))

Jack and Jill visited India, Japan and South Korea.

List(0.88 (Jill; visited; South Korea), 0.88 (Jill; visited; Japan), 0.88 (Jill; visited; India), 0.88 (Jack; visited; South Korea), 0.88 (Jack; visited; Japan), 0.88 (Jack; visited; India))

13-Jan-2021

Activities Done

- Read Research Paper on OpenKG

Pending Action Items

ToDo

- Replace OLLIE with OpenIE5.1

Proposed Method

OpenKGs do not require **pre-specified ontology**, making them highly adaptable.

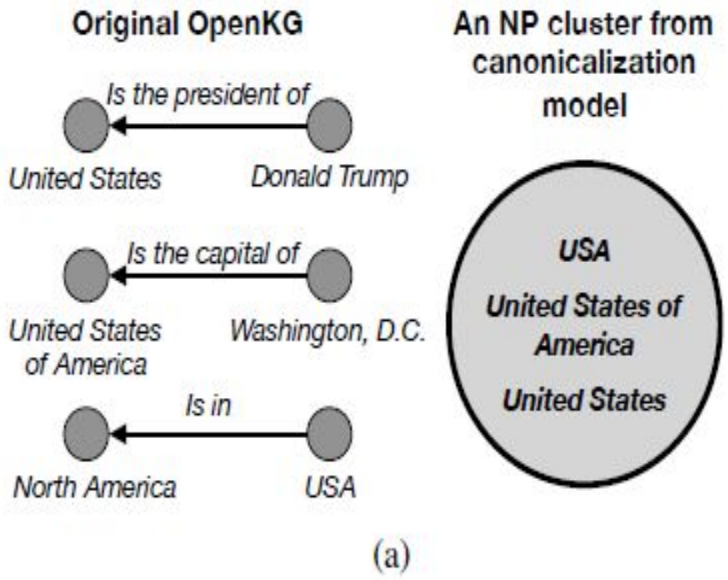
However, for downstream tasks such as Question Answering, Document Classification it is necessary to learn embeddings of NPs and RPs present as nodes and edges in an OpenKG

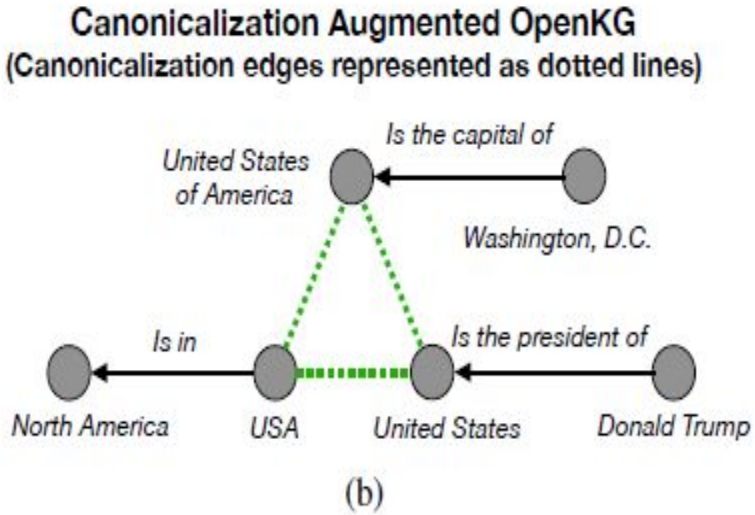
Possible Solution: Canonicalize OpenKGs, merge nodes and assigning them unique ids. However , this approach, results in often wrong node and relation merging, might result in accumulation of errors.

Proposed Method

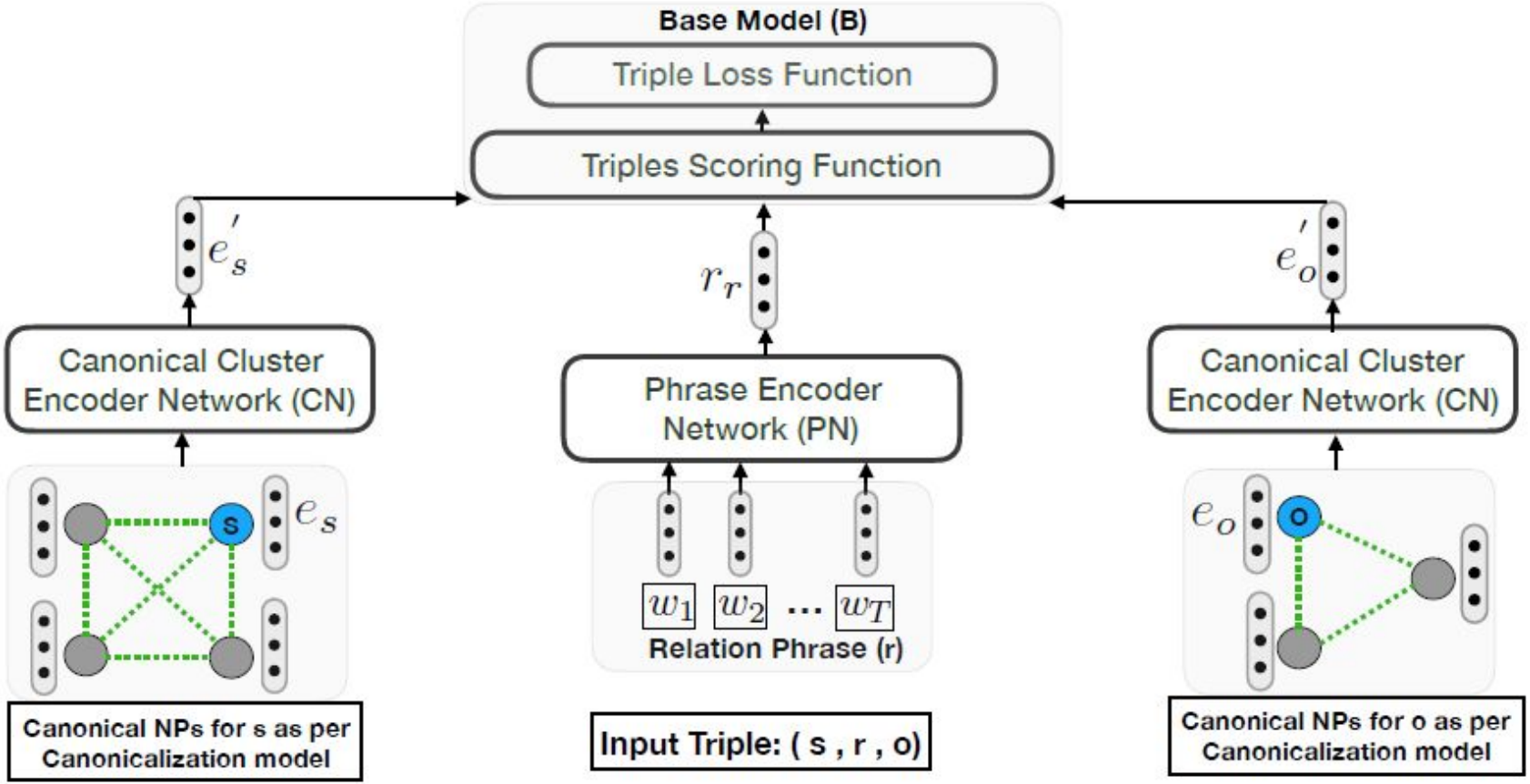
1) Run CESI **clustering algorithms** on the datasets to generate the NP canonical clusters. Add **unlabelled undirected edges between two NPs** , if they are canonical as per algorithm instead of merging. We get **Augmented OpenKG**
2) Run Canonical **Cluster Encoding Network** (a non-parametric message passing and update network, R-GCN like architecture) to obtain **contextvector** of Each NP. Which is passed to Basemodel Decoding Stage.
3) For RP, use **Phrase Encoder Network (**leverage rich semantic of pre trained word embeddings, followed by Bidirectional GRUs)
4) Base model (TransE, TransE, ConvE)

Augmented KG





CaRe



Knowledge Embeddings for Open KGs

CaRe: Open Knowledge Graph Embeddings

OpenKGs (Relations extracted from IE tools): (NPs as nodes, RPs as edges)

Even though several Knowledge Graph (KG) embedding methods have been recently proposed, all of those methods have targeted Ontological KGs, as opposed to OpenKGs.

Existing KG embedding methods are ineffective on OpenKGs as they are not **canonicalized**.
(Obama, Barack Obama, President Obama; took birth in, was born in)

Canonicalization of OpenKGs has received some attention lately, output of such methods has not been used to improve OpenKG embeddings.

Evaluation:

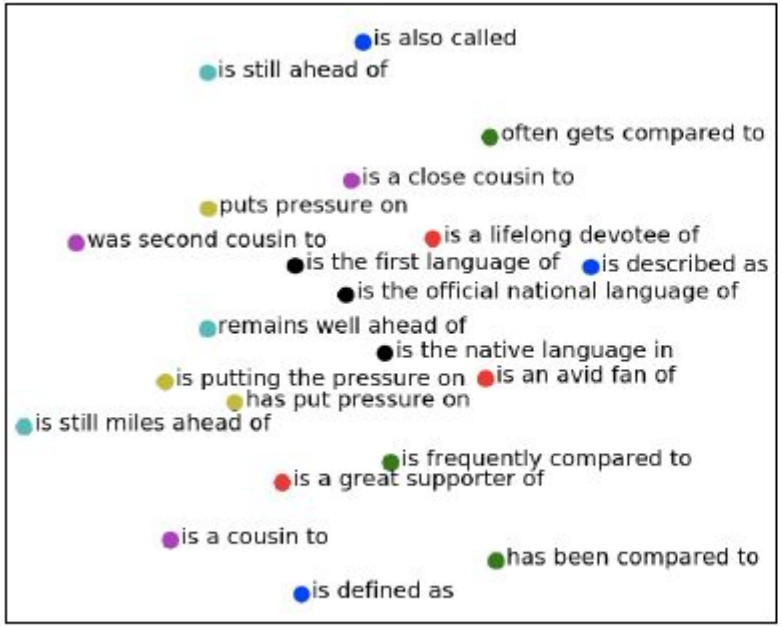
On an average, the number of train triples for each NP and RP is less than 2. In contrast, FB15k , an ontological KG, has on an average 32 triples for each entity and 360 triples for each Relation. Yet this paper achieves superior performance.

In the typical link prediction evaluation, an unseen triple is taken (s; r; o) and partial triples (s; r; ?) and (?; r; o) are shown to the model. It ranks all the entities in the graph for their likelihood to be the missing entity and the rank assigned to the true missing entity is considered.

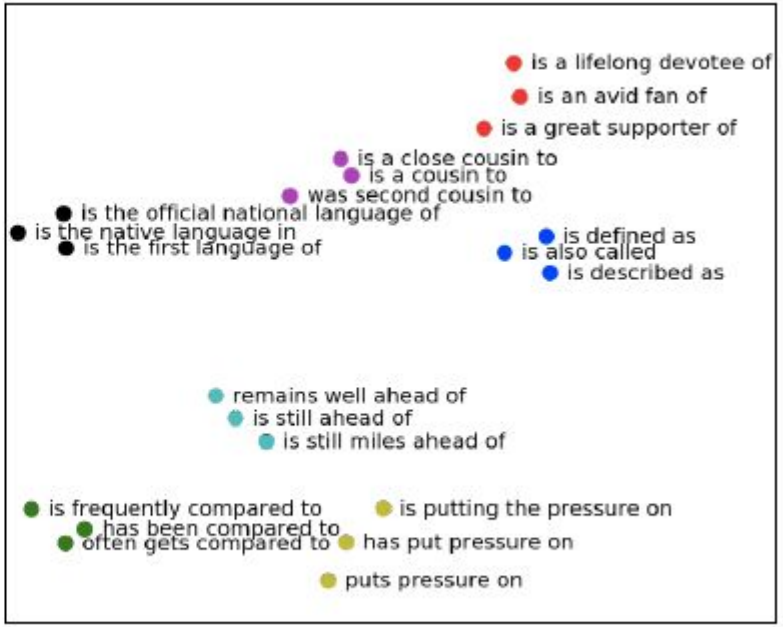
Result:

Method	ReVerb45K					ReVerb20K				
	MR	MRR	Hits@10	Hits@30	Hits@50	MR	MRR	Hits@10	Hits@30	Hits@50
TransE	2955.8	.193	.361	.446	.478	1425.8	.126	.299	.411	.468
TransH	2998.2	.194	.362	.442	.478	1464.4	.129	.303	.409	.467
DistMult	8988.8	.051	.051	.052	.065	6260.0	.033	.044	.055	.060
ComlEx	7786.5	.047	.047	.048	.073	5502.2	.037	.058	.075	.085
R-GCN	2866.8	.042	.046	.091	.113	1204.3	.122	.187	.263	.305
ConvE	2650.8	.233	.338	.401	.429	1014.5	.294	.402	.491	.541
CaRe(B=ConvE)	**1308.0**	**.324**	**.456**	**.543**	**.579**	**973.2**	**.318**	**.439**	**.525**	**.566**

tSNE plot of Relations:



(a) ConvE



(b) CaRe(B=ConvE)

12-Jan-2021

Activities Done

- Proposed KG architecture
- Relation Extraction done for operation Section, based on proposed architecture

Pending Action Items

ToDo

- Need to check suitability of open ended relations

Relation Extraction Operation Section

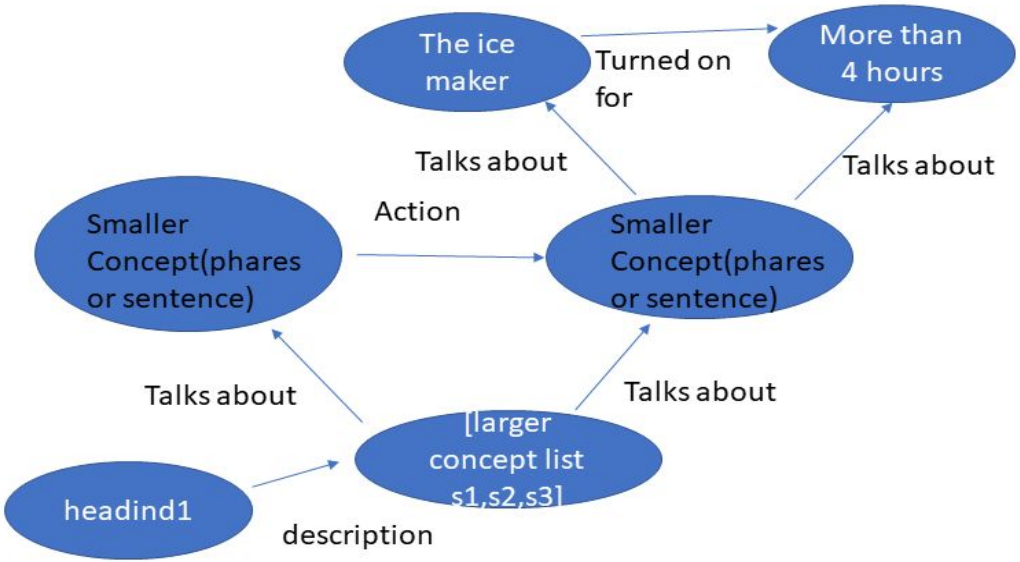
Analysis shows Granular information extraction does not make much sense:

Eg.
The control panel beeps and the temperature settings , display to confirm , that Display Mode
The inside of the refrigerator , smell like , plastic
The inside of the refrigerator ,smell at , first
the refrigerator ,has not been used for ,a long time
The icemaker ,begins producing ice , approximately 48 hours

Context is needed.

Proposed Architecture

Eg: {heading1:{'description':[sentnce1,sentence2,sentence3]}}



Proposed IE extraction

The output file

The output file has some duplicate relations, need to be fixed.

Challenges:

1) *Normalize Entities at leaf Nodes (NER is dropping a lot of these 'entities')*
2) *IE is failing to extract some relations correctly*
   Eg. Do not place any items on the Glide'N'Serve cover .
   0.797: (any items; 'N'Serve; cover)
   How to Prune?
3) *Normalize Relations(Defining Ontologies is cumbersome ,time-consuming) or Can we work with open Relations?*

11-Jan-2021

Activities Done

- Input file created for operation Section
- Relation Extraction done for operation Section

Pending Action Items

ToDo

-
-

Relation Extraction Operation Section

Input file: Link

Extracted Relations:
Link

8-Jan-2021

Activities Done

- Analysis of Discourse Connectives
- Work on Operation Section

Pending Action Items

ToDo

- Work on operation section

Discourse Connectives to Binary Relations

Discourse connectives are words or phrases that connect or relate two coherent sentences or phrases and indicate the presence of discourse relations.

*Taking **discourse connectives** to be the predicates of **binary discourse relations**.*

The ASER paper already shortlists 15 different patterns.

Constituency Parser already breaks the sentences into clauses.

Take Each type as an example, investigate the Constituency parse tree.

Discourse Connectives to Binary Relations

Relation Type	Seed Patterns
Precedence	E_1 **before** E_2; E_1 , **then** E_2; E_1 **till** E_2; E_1 **until** E_2
Succession	E_1 **after** E_2; E_1 **once** E_2
Synchronous	E_1, **meanwhile** E_2; E_1 **meantime** E_2; E_1, **at the same time** E_2
Reason	E_1, **because** E_2
Result	E_1, **so** E_2; E_1, **thus** E_2; E_1, **therefore** E_2; E_1, **so that** E_2
Condition	E_1, **if** E_2; E_1, **as long as** E_2
Contrast	E_1, **but** E_2; E_1, **however** E_2; E_1, , **by contrast** E_2; E_1, , **in contrast** E_2; E_1, , **on the other hand,** E_2; E_1, , **on the contrary,** E_2
Concession	E_1, **although** E_2
Conjunction	E_1 **and** E_2; E_1, **also** E_2;
Instantiation	E_1, **for example** E_2; E_1, **for instance** E_2
Restatement	E_1, **in other words** E_2
Alternative	E_1 **or** E_2; E_1, **unless** E_2; E_1, **as an alternative** E_2; E_1, **otherwise** E_2
ChosenAlternative	E_1, E_2 **instead**
Exception	E_1, **except** E_2

Discourse Connectives to Binary Relations

E1: I eat apple
E2: I go to swim

Case 1: Precedence
I eat apple before I go to swim

```
(S
  (NP (PRP I))
  (VP
    (VBP eat)
    (NP (NN apple))
    (SBAR
      (IN before)
      (S (NP (PRP I)) (VP (VBP go) (S (VP (TO to) (VP (VB swim)))))))))
```

['before I go to swim', 'I eat apple']

Discourse Connectives to Binary Relations

E1: I eat apple
E2: I go to swim

Case 1:
I eat apple then I go to swim

```
(S
  (NP (PRP I))
  (VP
    (VP (VBP eat) (NP (NN apple)))
    (ADVP (RB then))
    (NP (PRP I))
    (VP (VBP go) (S (VP (TO to) (VP (VB swim)))))))
```

mark

['I eat apple then I go to swim']

Discourse Connectives to Binary Relations

Case 2 Succession

I eat apple after I go to swim

```
(S
  (NP (PRP I))
  (VP
    (VBP eat)
    (NP (NN apple))
    (SBAR
      (IN after)
      (S (NP (PRP I)) (VP (VBP go) (S (VP (TO to) (VP (VB swim))))))))))
```

['after I go to swim', 'I eat apple .']

6-Jan-2021

Activities Done

- Research paper on COVID-KG
- Research paper on ASER

Pending Action Items

ToDo

- Read research paper
- Work on operation section

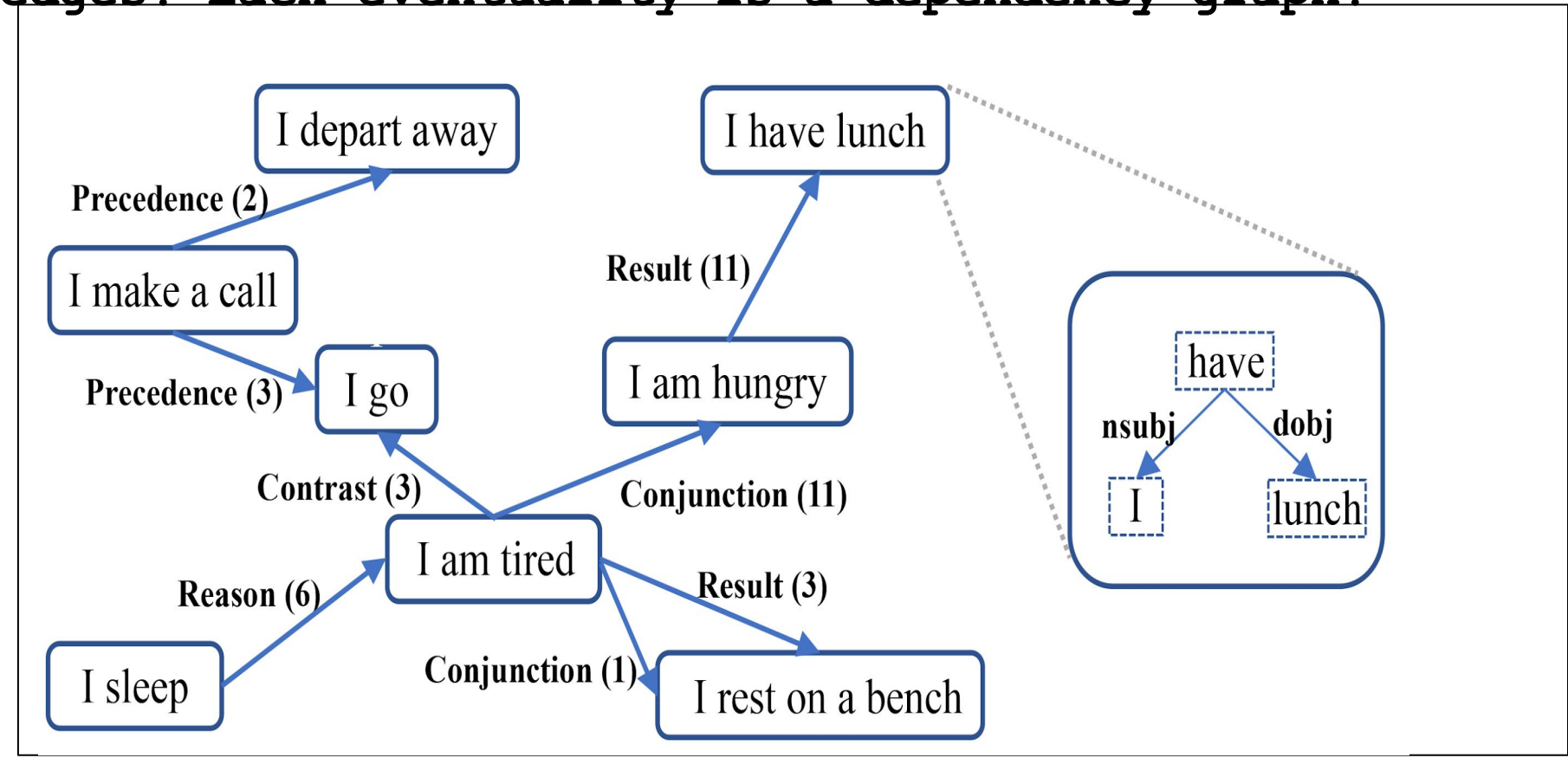
Research papers on KG construction

1)ASER: A Large-scale Eventuality Knowledge Graph

Existing large-scale knowledge graphs mainly focus on knowledge about entities while ignoring knowledge about **activities('I sleep'), states('I am hungry'), or events('I make a call") or their relations('I am hungry', may result in, 'I have lunch')**, which are used to describe how entities or things act in the real world.

ASER contains 15 relation types belonging to five categories, 194-million unique eventualities, and 64-million unique edges among them.

Eventualities are connected with weighted directed edges. Each eventuality is a dependency graph.



ASER

Storage:
Eventualities are connected with weighted directed edges. Each eventuality is a dependency graph.

In the eventuality table, record information about event ids, all the words, dependencies edges between words, and frequencies. In the relation table, we record ids of head and tail eventualities and relations between them.

KNOWLEDGE EXTRACTION
1)first preprocess the texts with the dependency parser. Then filter out all the sentences that contain Clauses. Extract verbs. Eventuality extracting using pattern matching

Pattern	Code	Example
n_1-nsubj-v_1	s-v	'The dog barks'
n_1-nsubj-v_1-dobj-n_2	s-v-o	'I love you'
n_1-nsubj-v_1-xcomp-a	s-v-a	'He felt ill'
n_1-nsubj-(v_1-iobj-n_2)-dobj-n_3	s-v-o-o	'You give me the book'
n_1-nsubj-a_1-cop-be	s-be-a	'The dog is cute'
n_1-nsubj-v_1-xcomp-a_1-cop-be	s-v-be-a	'I want to be slim'
n_1-nsubj-v_1-xcomp-n_2-cop-be	s-v-be-o	'I want to be a hero'
n_1-nsubj-v_1-xcomp-v_2-dobj-n_2	s-v-v-o	'I want to eat the apple'
n_1-nsubj-v_1-xcomp-v_2	s-v-v	'I want to go'
(n_1-nsubj-a_1-cop-be)-nmod-n_2-case-p_1	s-be-a-p-o	'It's cheap for the quality'
n_1-nsubj-v_1-nmod-n_2-case-p_1	s-v-p-o	'He walks into the room'
(n_1-nsubj-v_1-dobj-n_2)-nmod-n_3-case-p_1	s-v-o-p-o	'He plays soccer with me'
n_1-nsubjpass-v_1	spass-v	'The bill is paid'
n_1-nsubjpass-v_1-nmod-n_2-case-p_1	spass-v-p-o	'The bill is paid by me'

ASER

Eventuality Relation Extraction

Extract seed relations from the corpora by using the unambiguous connectives obtained from Penn Discourse Treebank(PDTB).

Relation	Explanation
<E_1, 'Precedence', E_2>	E_1 happens before E_2.
<E_1, 'Succession', E_2>	E_1 happens after E_2.
<E_1, 'Synchronous', E_2>	E_1 happens at the same time as E_2.
<E_1, 'Reason', E_2>	E_1 happens because E_2 happens.
<E_1, 'Result', E_2>	If E_1 happens, it will result in the happening of E_2.
<E_1, 'Condition', E_2>	Only when E_2 happens, E_1 can happen.
<E_1, 'Contrast', E_2>	E_1 and E_2 share a predicate or property and have significant difference on that property.
<E_1, 'Concession', E_2>	E_1 should result in the happening of E_3, but E_2 indicates the opposite of E_3 happens.
<E_1, 'Conjunction', E_2>	E_1 and E_2 both happen.
<E_1, 'Instantiation, E_2>	E_2 is a more detailed description of E_1.
<E_1, 'Restatement', E_2>	E_2 restates the semantics meaning of E_1.
<E_1, 'Alternative', E_2>	E_1 and E_2 are alternative situations of each other.
<E_1, 'ChosenAlternative', E_2>	E_1 and E_2 are alternative situations of each other, but the subject prefers E_1.
<E_1, 'Exception', E_2>	E_2 is an exception of E_1.
<E_1, 'Co-Occurrence', E_2>	E_1 and E_2 appear in the same sentence.

ASER

Seed Connectives:

Neural network based approach to bootstrap. The general steps of bootstrapping are as follows.

• Step 1: Use the extracted seed training instances as the initial labeled training instances.

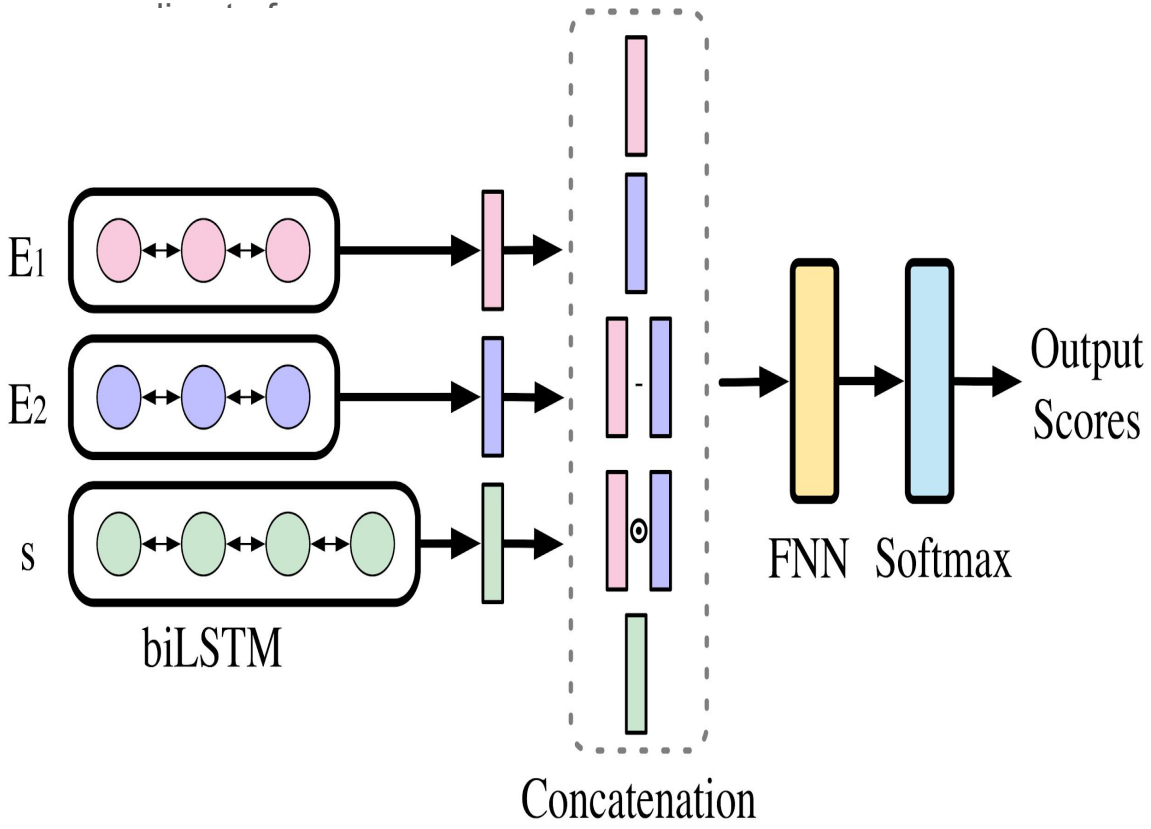
• Step 2: Train a classifier based on labeled training instances.

• Step 3: Use the classifier to predict relations of each training instance. If the prediction confidence of certain relation type T Is higher than the selected threshold, label this instance withT and add it to the labeled training instances

Relation Type	Seed Patterns
Precedence	E_1 before E_2; E_1 , then E_2; E_1 till E_2; E_1 until E_2
Succession	E_1 after E_2; E_1 once E_2
Synchronous	E_1, meanwhile E_2; E_1 meantime E_2; E_1, at the same time E_2
Reason	E_1, because E_2
Result	E_1, so E_2; E_1, thus E_2; E_1, therefore E_2; E_1, so that E_2
Condition	E_1, if E_2; E_1, as long as E_2
Contrast	E_1, but E_2; E_1, however E_2; E_1, , by contrast E_2; E_1, , in contrast E_2; E_1, , on the other hand, E_2; E_1, , on the contrary, E_2
Concession	E_1, although E_2
Conjunction	E_1 and E_2; E_1, also E_2;
Instantiation	E_1, for example E_2; E_1, for instance E_2
Restatement	E_1, in other words E_2
Alternative	E_1 or E_2; E_1, unless E_2; E_1, as an alternative E_2; E_1, otherwise E_2
ChosenAlternative	E_1, E_2 instead
Exception	E_1, except E_2

ASER

Bootstrapping:
four different classifiers are trained c
categories (Temporal, Contingency,
Comparison, Temporal).
Each classifier predicts the types bel
of each instance



ASER

Pattern	Code	Example
n_1-nsubj-v_1	s-v	'The dog barks'
n_1-nsubj-v_1-dobj-n_2	s-v-o	'I love you'
n_1-nsubj-v_1-xcomp-a	s-v-a	'He felt ill'
n_1-nsubj-(v_1-iobj-n_2)-dobj-n_3	s-v-o-o	'You give me the book'
n_1-nsubj-a_1-cop-be	s-be-a	'The dog is cute'
n_1-nsubj-v_1-xcomp-a_1-cop-be	s-v-be-a	'I want to be slim'
n_1-nsubj-v_1-xcomp-n_2-cop-be	s-v-be-o	'I want to be a hero'
n_1-nsubj-v_1-xcomp-v_2-dobj-n_2	s-v-v-o	'I want to eat the apple'
n_1-nsubj-v_1-xcomp-v_2	s-v-v	'I want to go'
(n_1-nsubj-a_1-cop-be)-nmod-n_2-case-p_1	s-be-a-p-o	'It's cheap for the quality'
n_1-nsubj-v_1-nmod-n_2-case-p_1	s-v-p-o	'He walks into the room'
(n_1-nsubj-v_1-dobj-n_2)-nmod-n_3-case-p_1	s-v-o-p-o	'He plays soccer with me'
n_1-nsubjpass-v_1	spass-v	'The bill is paid'
n_1-nsubjpass-v_1-nmod-n_2-case-p_1	spass-v-p-o	'The bill is paid by me'

Research papers on KG construction

1)COVID-19 Literature Knowledge Graph Construction and Drug Repurposing Report Generation

Extract fine grained multimedia knowledge elements (entities, relations and events) from scientific literature.

Construction:
(1) coarsegrained entity extraction and entity linking for four entity types: **Gene nodes, Disease nodes, Chemical nodes, and Organism**.
2)Entity ontology defined in the Comparative Toxicogenomics Database (CTD) , and obtain a Medical Subject Headings (MeSH) Unique ID for each mention.
3)Based on the MeSH Unique IDs, link all entities to the CTD and extract 133 subtypes of relations (GeneChemicalInteraction Relationships, ChemicalDisease Associations, GeneDisease Associations, ChemicalGO Enrichment Associations and ChemicalPathway Enrichment Associations.)
4) Event extraction(Transcription, localiation, protein modification etc)
5)we apply our fine-grained entity extraction system CORD-NER.
6)**Deep figure** to automatically detect and extract figures from each PDF document, pipeline to segment individual subfigures and then align each subfigure with its corresponding sub-caption. (employ **Figure-separator** to detect and separate all non-overlapping image regions.)**OCR**
7)KG can be constructed that links specific molecular structure images to corresponding drug entities in the KG.

Research papers on KG construction

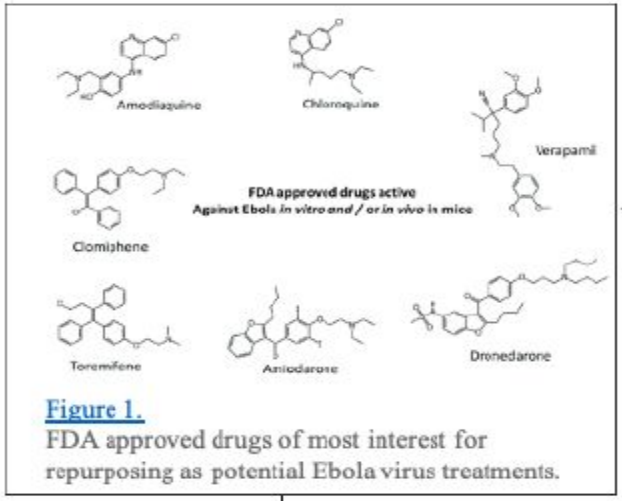
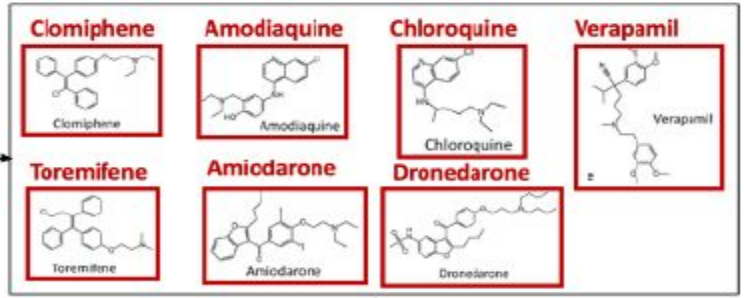
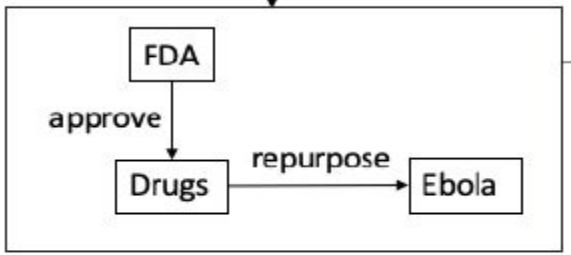


Figure 1.
FDA approved drugs of most interest for repurposing as potential Ebola virus treatments.

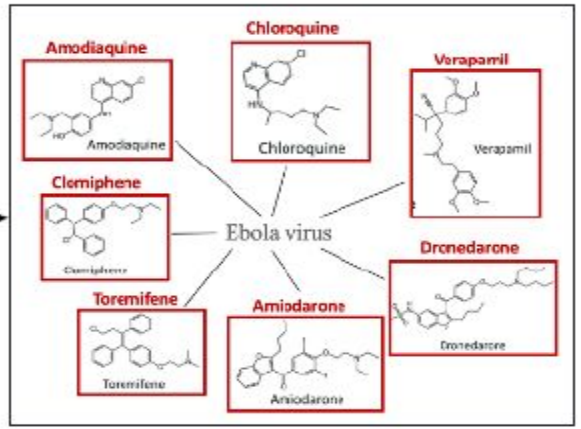
Entity Grounding for Drug Molecular Structure Image



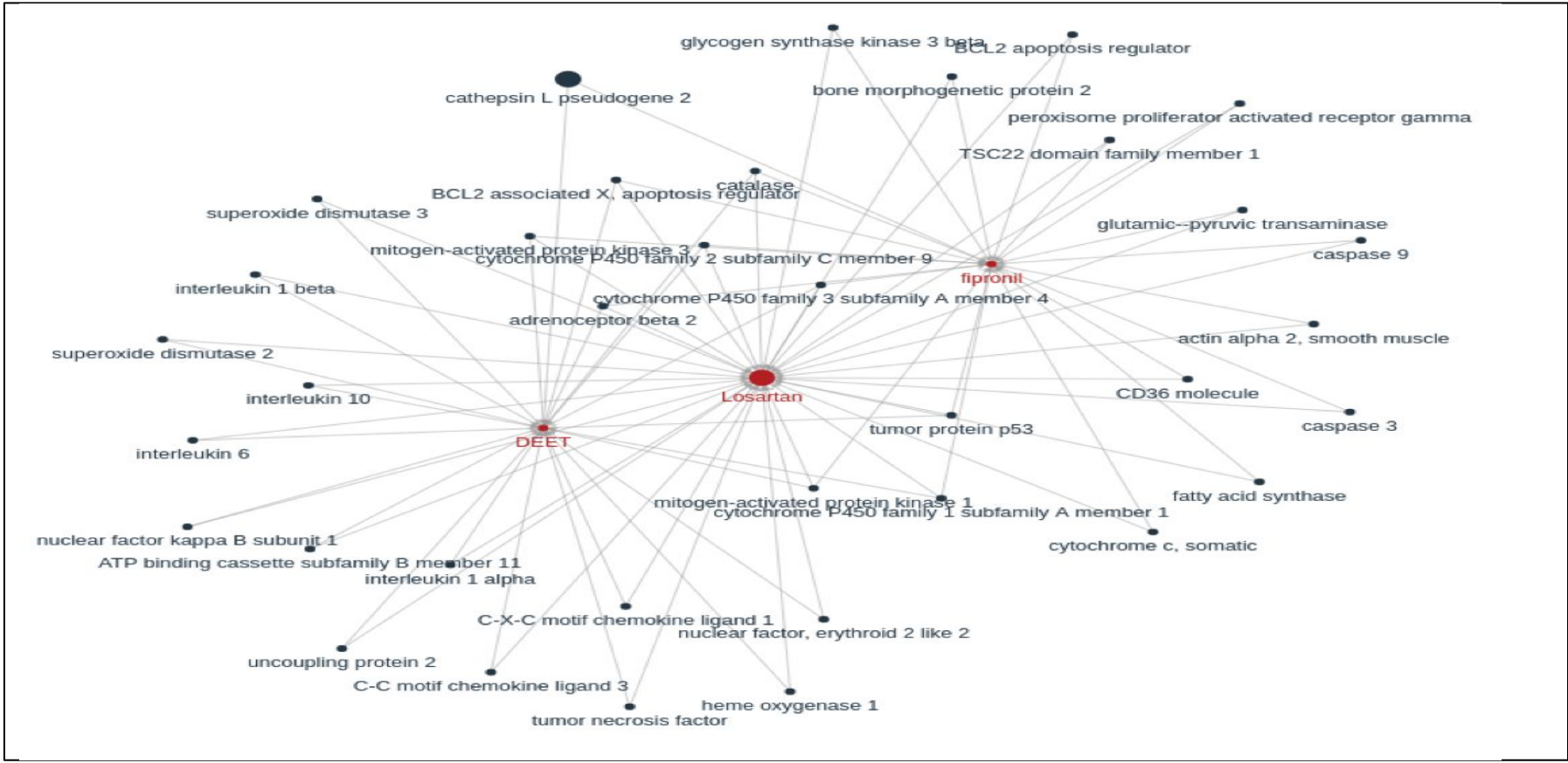
KG from caption text



Multimedia Knowledge Graph Expansion



Research papers on KG construction



5-Jan-2021

Activities Done

- Literature survey of research papers on KG construction
- Designing json for operation section in progress

Pending Action Items

ToDo

- Read research paper
- Work on operation section

Research papers on KG construction

1)ASER: A Large-scale Eventuality Knowledge Graph
2)COVID-19 Literature Knowledge Graph Construction and Drug Repurposing Report Generation
3)Learning a Health Knowledge Graph from Electronic Medical Records
4)KnowIME: A System to Construct a Knowledge Graph for Intelligent Manufacturing Equipment
5)Robustly Extracting Medical Knowledge from EHRs: A Case Study of Learning a Health Knowledge Graph

Revevent:
1)ASER: A Large-scale Eventuality Knowledge Graph
2)COVID-19 Literature Knowledge Graph Construction and Drug Repurposing Report Generation

Operation section(Design)
Analyze and identify on higher as well as granular level
New relations: use, start, stop, change mode, replacement information, has default value, 'replacement period'

4-Jan-2021

Activities Done

- Research paper Multi-hop QA

Pending Action Items

ToDo

Summary

- AllenNLP constituency Parser to break Complex sentences into constituent clauses
- Create json dataset in dictionary format for safety instruction/ define relations
- Create python wrapper for java based OLLIE tool
- Integrated IE tools (Extract dictionary defined relation->Constituency-->OLLIE-->Dependency parse)
- Extract Triples for Created Safety Instruction for each steps
- Research paper on Simple QA answering using character and word level embeddings
- Research paper on multi-hop KGQA using graph embeddings.

Research Paper

Paper Titled: Improving Multi-hop Question Answering over Knowledge Graphs using Knowledge Base Embeddings

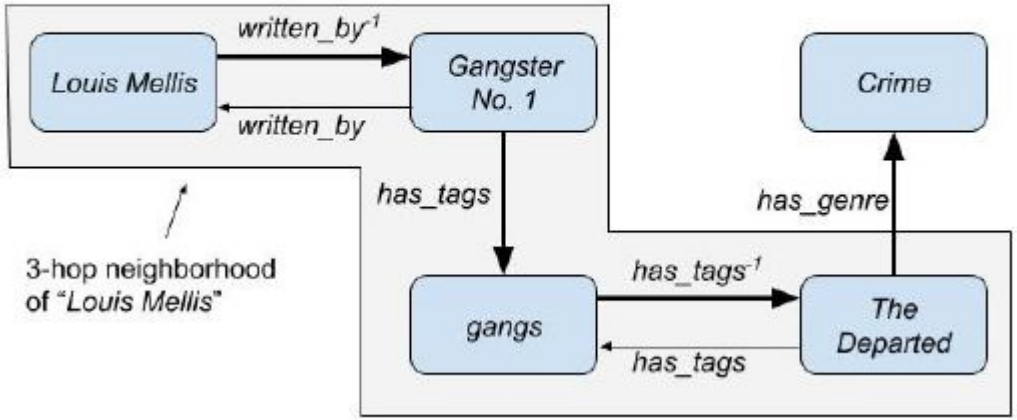
Solution for:
1)Multi-hop KGQA
2)KGs are often incomplete with missing links, additional challenges for multi-hop KGQA
3) Sparse KGQA
4)it relaxes the requirements of heuristic neighborhood limit by previous multi-hop KGQA methods.

Contribution:
1) EmbedKGQA is the first method to use KG embeddings for multi-hop.
2) EmbedKGQA relaxes the requirement of answer selection from a pre-specified local neighborhood

Research Paper

1)Absence of the edge has genre(Gangster No. 1, Crime)in the incomplete KG makes it much harder to answer the input NL question(Reason over a longer path)
2)Heuristic limit

Question: *What are the genres of movies written by Louis Mellis?*
Answer : *Crime*



Research Paper

1)**Problem statement:** Given a natural language question q and a topic entity **eh** present in the question, the task is to extract an entity **et** that correctly answers the question **q**.
2)**Settings:** No fine grained annotations, only Question and Answer
It consists of 3 modules:

KGEmbedding Module creates embeddings for all entities in the KG.

Question Embedding Module finds the embedding of a question

Answer Selection Module reduces the set of candidate answer entities and selects the final answer.

Research Paper

Question Embedding Module :This is done using a feed-forward neural network that first embeds the question q using RoBERTa into a 768-dimensional vector. This is then passed through 4 fully connected linear layers with ReLU activation and finally projected onto the complex space.

$$\phi(e_h, e_q, e_a) > 0 \quad \forall a \in \mathcal{A}$$

$$\phi(e_h, e_q, e_{\bar{a}}) < 0 \quad \forall \bar{a} \notin \mathcal{A}$$

Each question, the score is calculated with all the candidate answer entities . The model is learned by minimizing the binary cross entropy loss between the sigmoid of the scores and the target labels, where the target label is 1 for the correct answers and 0 otherwise.

Research Paper

Answer Selection Module:At inference, the model scores the (head, question) pair against all possible answers. For relatively smaller KGs like MetaQA, simply select the entity with the highest score.

$$e_{ans} = \arg\max_{a' \in \mathcal{E}} \phi(e_h, e_q, e_{a'})$$

if the knowledge graph is large, pruning the candidate entities can significantly improve the performance of EmbedKGQA.

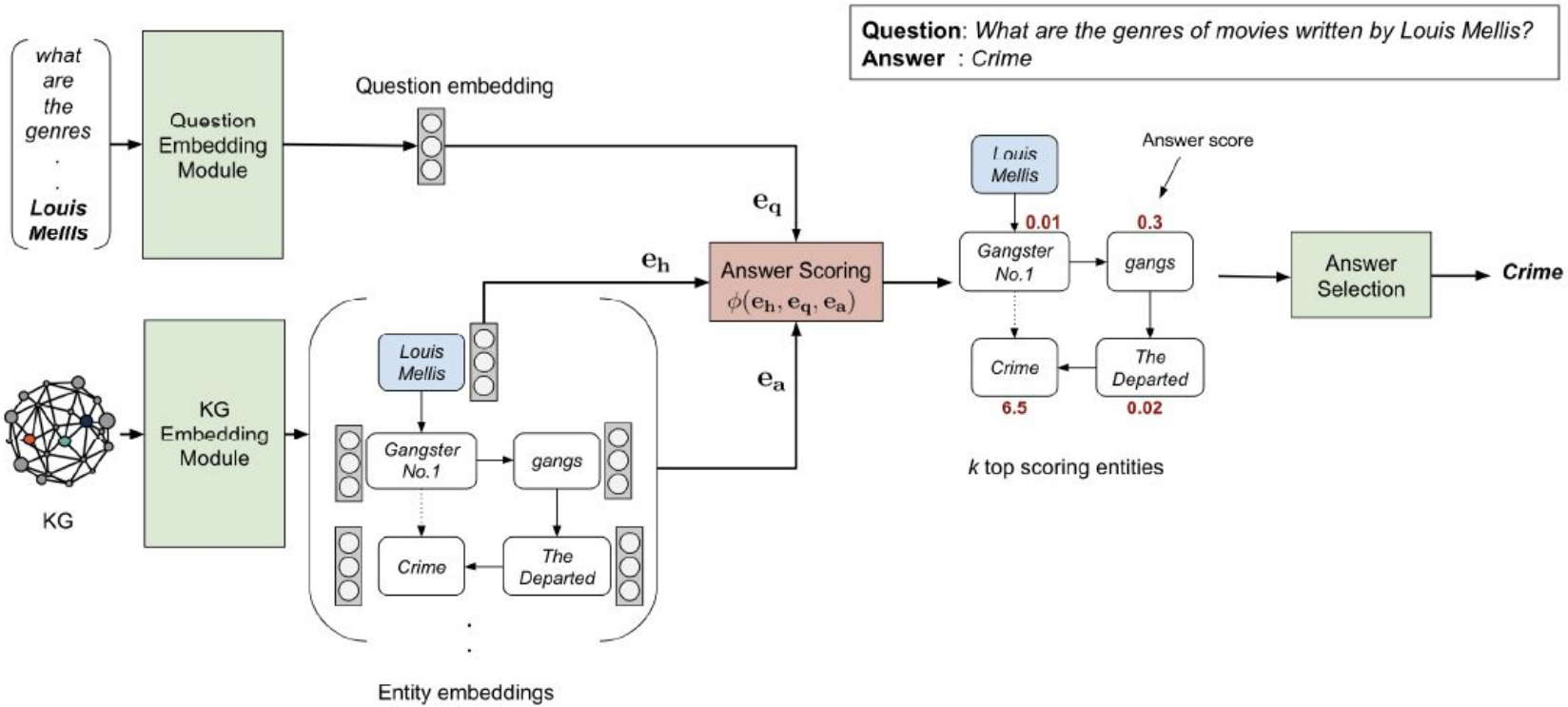
$$h_q = \text{RoBERTa}(q')$$

$$S(r, q) = \text{sigmoid}(h_q^T h_r)$$

$$\text{RelScore}_{a'} = |\mathcal{R}_a \cap \mathcal{R}_{a'}|$$

$$e_{ans} = \arg\max_{a' \in \mathcal{N}_h} \phi(e_h, e_q, e_{a'}) + \gamma * \text{RelScore}_{a'}$$

Research Paper



Research Paper

Model	MetaQA KG-Full			MetaQA KG-50		
	1-hop	2-hop	3-hop	1-hop	2-hop	3-hop
VRN	**97.5**	89.9	62.5	-	-	-
GraftNet	97.0	94.8	77.7	64.0 (91.5)	52.6 (69.5)	59.2 (66.4)
PullNet	97.0	**99.9**	91.4	65.1 (92.4)	52.1 (90.4)	59.7 (85.2)
KV-Mem	96.2	82.7	48.9	63.6 (75.7)	41.8 (48.4)	37.6 (35.2)
EmbedKGQA (Ours)	**97.5**	98.8	**94.8**	**83.9**	**91.8**	**70.3**

Model	WebQSP KG-Full	WebQSP KG-50
KV-Mem	46.7	32.7 (31.6)
GraftNet	66.4	48.2 (49.7)
PullNet	**68.1**	50.1 (51.9)
EmbedKGQA	66.6	**53.2**

31-Dec-2020

Activities Done

- Analysis of Open challenges from ‘Safety’ extraction
- Research paper

Pending Action Items

ToDo

SAFETY INSTRUCTION--Open challenges

1. Open Challenges:
   Relation extraction extracts meaningful relations but the entities are not noun-phrases.
   Noun phrases can be extracted using **TextBlob** package
   Eg **332** Noun Phrases are extracted for safety section.
   Eg. of Extracted relations:
   (If a leak is detected , **do** , avoid any naked flames or potential sources of ignition and air out the room in which the appliance is standing for several minutes .)
   Question: What should I do, if leak is detected?
   But neither of the head and tail is **Noun Phrase**

- **Should we consider all extracted heads and tails as entities?**
- **Challenge: Entity Space Unbounded; pressure on subsequent Question Answering module.**

Research Paper

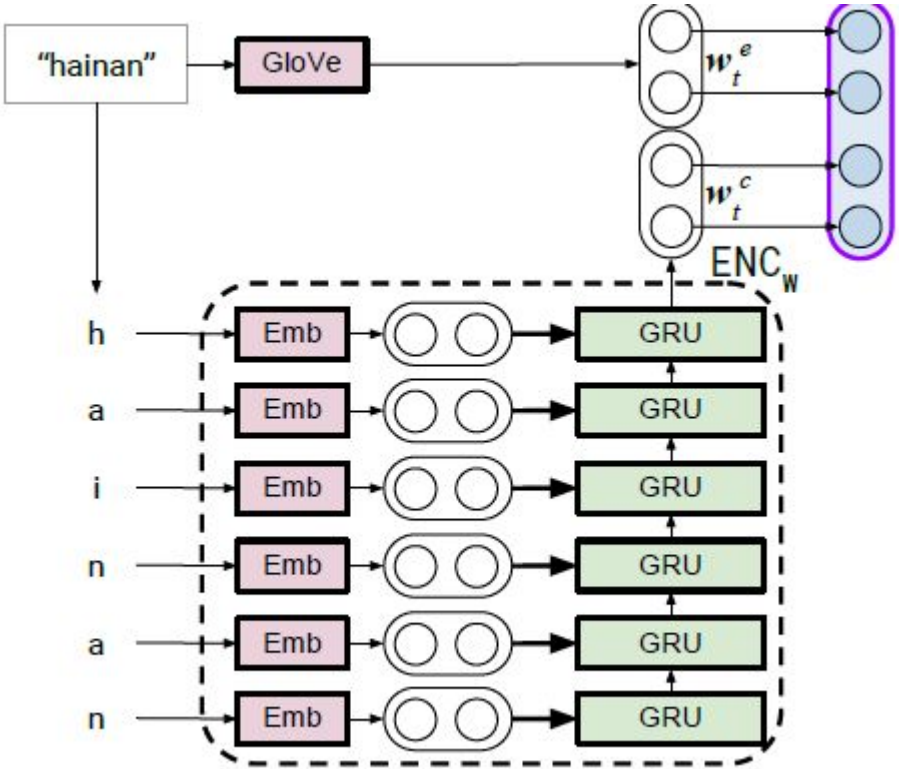
Paper Titled: Neural Network-based Question Answering over Knowledge Graphs on Word and Character Level

Solution for:
1)The traditional paradigm for QA approaches Complex NLP pipeline(POS tagger, template-tting, relation extraction, token merging and entity mapping.)
2)Error propagation
3) The traditional paradigm for QA approaches are domain specific.(KG independent Entity and Relation representation)

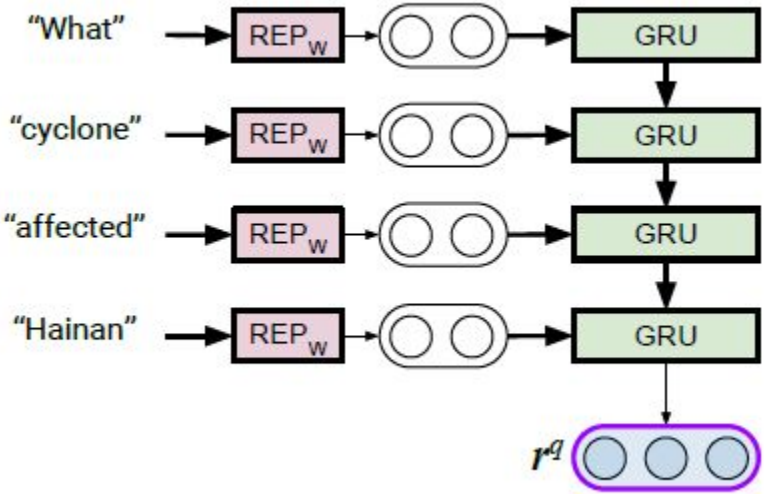
Contribution:
1)merges word- and character-level representations of words modelling of the question to exploit the advantages of both.
2)Knowledge based independent representation for both Entity and Predicate only from textual information.
Dataset:Freebase

Research Paper

Word Representation:

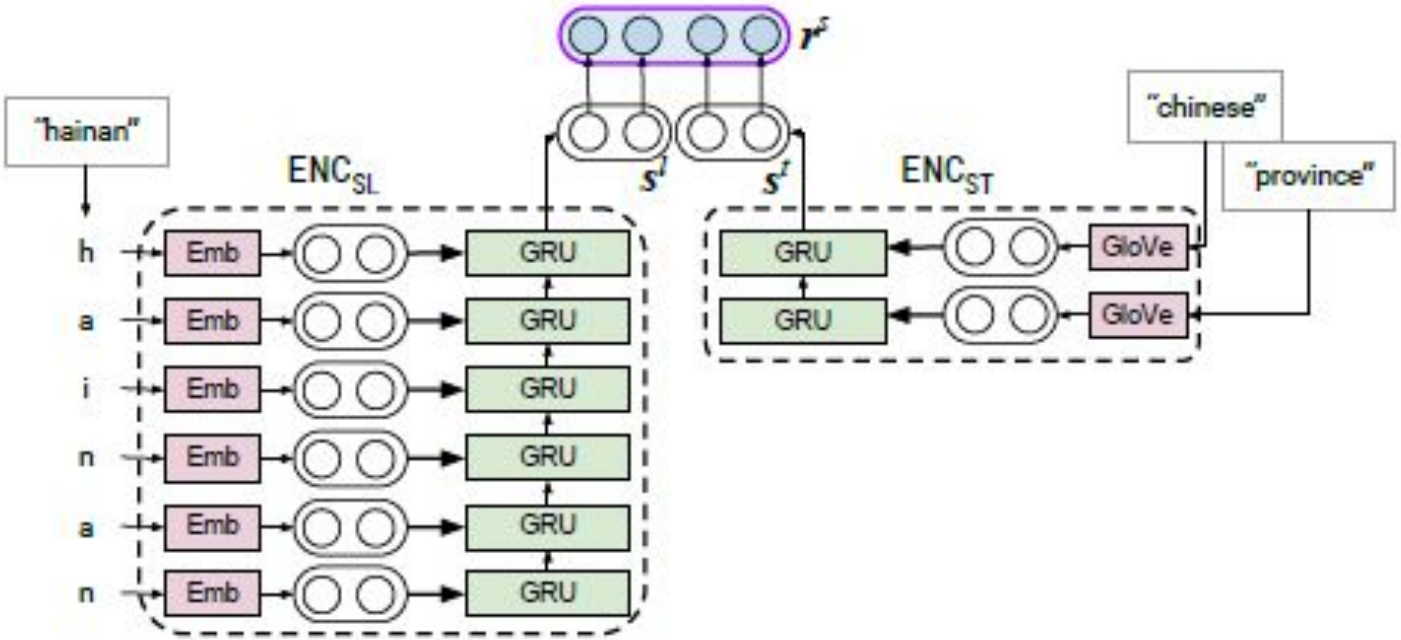


Question Representation(For both subject and Predicate)



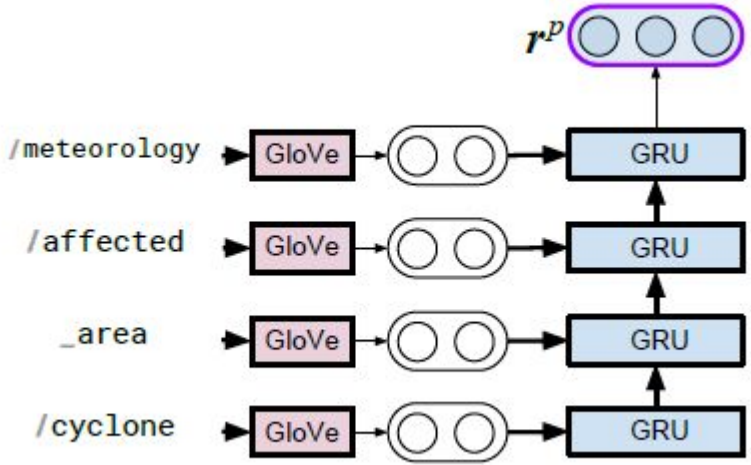
Research Paper

Entity Encoder: Entity Label(Character Encoder), Entity Type(Word vector). Concatenation final vector



Research Paper

Predicate Encoder:



Inference

$$S_s(q,s) = \cos(r_q^s, r_s)$$
$$S_p(q,p) = \cos(r_q^p, r_p) \ ,$$

$$\hat{s} = \underset{s_i \in \mathcal{C}_s}{\operatorname{argmax}} S_s(q, s_i) \ , \quad (1)$$

$$\hat{p} = \underset{p_j \in \mathcal{C}_p}{\operatorname{argmax}} S_p(q, p_j) \ . \quad (2)$$

Approach	Setting	Test Accuracy %
Bordes et al. [4]	end-to-end	62.7
Yin et al. [26]	end-to-end	68.3
Dai et al. [8]	end-to-end	62.6*
Golub and He [12]	end-to-end	70.9
Our approach	end-to-end	71.2
Dai et al. [8]	active linking	75.7*
Yin et al. [26]	focused pruning	76.4

30-Dec-2020

Activities Done

- Python Wrapper for java based OLLIE
- Integrate IE tool(OLLIE+ Dep parse)

Pending Action Items

- 1) Research Papers for QA using knowledge embeddings, specifically Multi-hop
- 2) Analysis of other section question, to design

ToDo

SAFETY INSTRUCTION

1. Open Language Learning for Information Extraction is java based, python wrapper is created to call the OLLIE module
2. For any extraction if confidence score is >0.62, relation is included
3. Dependency Parse based extraction is also incorporated
4. The overall Integration as follows:
    a. Extract defined relations from input JSON
    b. Break the complex sentences in (Cause1, ‘do’, Clause 2) using allennlp constituency parser
    c. Apply OLLIE to all sentences and pick up relations if confidence score>0.62
    d. If OLLIE fails, to extract any relations, apply dependency parsing to those sentences.
    e. The final file contains relations extracted from steps(a., b., c., d.)
5. This file needs to be reviewed . link

29-Dec-2020

Activities Done

- Analysis of Data format for operation section and dict/json file creation
- Extracted structured relation from json to triple format
- Extracted Clausal relations

ToDo

- Add OLLIE IE

Pending Action Items

- 1) Research Papers for QA using knowledge embeddings, specifically Multi-hop
- 2) Integrated IE tool (Constituency-->OLLIE-->Dependency parse)
- 3) Build triple Graph for Safety section --> Inspect how it is coming for a single section.
- 4) Analysis of other section question, to design

SAFETY INSTRUCTION

Input file in JSON Format:

-

1)Each dict key is either a relation or Node(The nodes need not be entity, it could be phrases or block of data)
2) Some relations are predefined by the structure. eg 'description', 'mean', 'is_for', 'be_avoided_by', 'purpose'
3)as USER need to follow the instruction, so in triple format , the following instruction should be

(USER, fllow_instruction_during_installation, [block of instruction])
so a new default entity 'USER is created.

Input file link

- Input json file link
- Extracted basic Relations from dict format
- After Clausal Relation Extraction format

28-Dec-2020

Activities Done

- Create dataset in dictionary format for safety instruction/ Define relations
- Coding for Integration of three IE tools in progress

ToDo

- Identify relations (pattern) from LG Manual Questions

Pending Action Items

- 1) Research Papers for QA using knowledge embeddings, specifically Multi-hop
- 2) Integrated IE tool (Constituency-->OLLIE-->Dependency parse)
- 3) Build triple graphs for Safety section --> Inspect how it is coming for a single section.
- 4) Analysis of other section question, to design

24-Dec-2020

Activities Done

- Coded the module the break complex declarative/imperative sentences to simple sentence
- Improved Dependency Parsing based IE

ToDo

- Identify relations (pattern) from LG Manual Questions

Pending Action Items

- 1) Research Papers for QA using knowledge embeddings, specifically Multi-hop
- 2) Integrated IE tool (Constituency-->OLLIE-->Dependency parse)
- 3) Build Graph for Safety section --> Inspect how it is coming for a single section.
- 4) Analysis of other section question, to design

Constituency Parsing: AllenNLP

Scope:

1) Complex Sentences
2) Declarative and Imperative sentences

Out of scope:

1) Interrogative Sentence
2) Inverted Declarative Sentence

For details Clause Tag

Eg. '(S (ADVP (RB Never)) (VP (VB attempt) (S (VP (TO to) (VP (VB operate) (NP (DT this) (NN appliance))))) (SBAR (IN if) (S (NP (PRP it)) (VP (VBZ is) (VBN damaged) (, ,) (VBG malfunctioning) (, ,) (ADVP (RB partially)) (VBN disassembled) (, ,) (CC or) (VP (VBZ has) (NP (NP (ADJP (JJ missing) (CC or) (JJ broken)) (NNS parts)) (, ,) (PP (VBG including) (NP (DT a) (VBN damaged) (NN cord) (CC or) (NN plug)))))))) (. .))'

Constituency Parsing: AllenNLP

Logic:
Recursively traverse subtrees, if ‘SBAR’ is at root of any subtree, leaves form sub_ordinate clause.
Truncate the subtree from main tree, remaining tree forms Principal Clause
Never attempt to operate this appliance if it is damaged, malfunctioning, partially disassembled, or has missing or broken parts, including a damaged cord or plug.

['if it is damaged , malfunctioning , partially disassembled , or has missing or broken parts , including a damaged cord or plug', 'Never attempt to operate this appliance .']
When moving the refrigerator, be careful not to roll over or damage the power cord.
['When moving the refrigerator', ', be careful not to roll over or damage the power cord .']

Contact an authorized service centre when installing or relocating the refrigerator.
['when installing or relocating the refrigerator', 'Contact an authorized service centre .']

Dependency Parsing: IE

for some examples it's not extracting any relations
This product is not to be used for special purposes such as the storage of medicine or test materials or for use on ships, etc.

[]

Reason:Object to Open Causal Complement

Now
[('This product',
'not is used for',
'special purposes such as the storage of medicine or test materials'),
('This product', 'not is used for', 'use on ships etc')]

- Add subject to imperative sentences

Imperative sentence : NO IE

- Do not modify or extend the power cord.
- Do not operate the refrigerator or touch the power cord with wet hands.

No relations are getting extracted as No Subjects, add a dummy subject

23-Dec-2020

Activities Done

- Analysis, Break complex sentences into constituent clauses using constituency parsing

Pending Action Items

-

ToDo

- Identify relations (pattern) from LG Manual Questions

Constituency Parsing: Break Complex sentences

Many Relations are in form of Conditional Relations

If X happens , Do the Y || Do the Y, If X happens
When X happens Do the Y II Do the Y, when X happens

Eg.
Never attempt to operate this appliance if it is damaged, malfunctioning, partially disassembled, or has missing or broken parts, including a damaged cord or plug.

Question?
What should i do if the appliance is damaged, malfunctioning,partially disassembled?

Answer?
Never attempt to operate this appliance

Constituency Parsing: Break Complex sentences

Issue?
1)No extraction for (OLIE)
2)[('it is damaged ,', 'not attempt disassembled', 'partially'), ('it is damaged ,', 'not attempt', 'has missing broken parts , including a damaged cord plug'), ('has missing broken parts , including a damaged cord plug', 'not attempt disassembled', 'partially')]
3) SRL gives too many argument

Solutions:

1) Break the Complex and Compound sentences , into simple sentences
2) It was also observed that OLLIE performs better on simple sentences, but extract no relations on complex sentences.

How?

1) Apply **Constituency parsing** to convert Complex sentences to simple sentence
2) Apply Relation (Simple Cause 1-Conditions-Simple Clause 2)
3) Apply relations extractions on individual simple clauses to extract further relations

Constituency Parsing: Break Complex sentences

sentence :
Never attempt to operate this appliance if it is damaged, malfunctioning, partially disassembled, or has missing or broken parts, including a damaged cord or plug.
<class 'nltk.tree.ParentedTree'>
['if it is damaged malfunctioning partially disassembled', 'if it has missing or broken parts including a damaged cord', 'Never attempt to operate this appliance', 'Never plug']

When moving the refrigerator, be careful not to roll over or damage the power cord.
<class 'nltk.tree.ParentedTree'>
['not to roll over', 'When be careful or damage the power cord moving the refrigerator', 'When be careful or damage the power cord be careful', 'When be careful or damage the power cord damage the power cord']
sentence :
Contact an authorized service centre when installing or relocating the refrigerator.
<class 'nltk.tree.ParentedTree'>
['when installing or relocating the refrigerator']
Note: For some format of the sentences , the clauses are incorrect

22-Dec-2020

Activities Done

- Annotations of Head Entities and Relations from the given questions and analysis

Pending Action Items

- Identify relations (pattern) from LG Manual Questions

ToDo

- Read Research paper on automatic kg creation

Analysis/Observations from Questions from Manuals

1) Most of the questions are Descriptive types, which does not have any entity, but describe a method or process or action!
2) Parse/ Entity Detection will not work in many cases!
   - This appliance is intended to be used in household and similar applications such as:
   - **staff kitchen areas in shops, offices and other working environments;**
   - **farm houses and by clients in hotels, motels and other residential type environments;**
   - **bed and breakfast type environments;**
   - **catering and similar non-retail applications.**
3) Questions: Where the refrigerator appliance is intended to be used?
4) Yes/No questions needs to be addressed separately, as It's a separate case of Inference.
5) ‘Wh’- Questions have (Head, Predicate) and Tail as Answers. Yes/No Questions, are structurally different have a triple.

Analysis/Observations from Questions from Manuals

1) Eg. (Can i install the refrigerator in a damp or dusty place?)
(the refrigerator, install in, a damp or dusty place)

Eg. if I ask “Should I install in sunny and bright places”? What should be the answer?

2)**Control Panel**

Sets the refrigerator te

temperature and freezer temperature, the water filter condition and the dispenser mode.

Questions: What is the use of the control panel?
Answer: Sets the refrigerator temperature and freezer temperature, the water filter condition and the dispenser mode.
(Control Panel, function, Sets the refrigerator temperature and freezer temperature, the water filter condition and the dispenser mode.)

Question: Where we will set the refrigerator temperature?
(refrigerator temperature, is set in , control panel)
This relations can’t be captured from information extraction! Or Parsing. Rules can work but it changes in every section(Eg Operation, Control Panel)

21-Dec-2020

Activities Done

- Download and Installed OLLIE, used in the paper
- Analysis of Questions from Manuals and identification of relation

Pending Action Items

- Identify relations (pattern) from LG Manual Questions

ToDo

- Read Research paper on automatic kg creation

Analysis from Questions from Manuals

1) A lot questions are asking about ‘mean/display/tell’. This are similar relations

2) **WARNING**

   You may be killed or seriously injured if you do not follow instructions.

   The question is “What the warning messages say”? Such relations can’t be extracted from parsing or information extraction

3) A vast number of answers are not Named entities or Noun phrases, rather subordinate clauses.

4) How do we handle negative sentences? No proper mention of handling of negative relations . Open IE can’t extract negative relations. OLLIE can extract relations.

Open Language Learning for Information Extraction examples:

Download from Link . It outputs Relations with confidence score 1.0 being the highest
Mumbai is commercial capital of India
>
0.931: (Mumbai; is commercial capital of; India)
0.795: (Mumbai; is; commercial capital of India)
0.517: (commercial; is capital of; India)

Manual sentences:
Do not install the refrigerator in a damp or dusty place where insulation on electrical parts may deteriorate.

>0.895: (the refrigerator; Do not install in; a damp or dusty place where insulation on electrical parts may deteriorate)

Open Language Learning for Information Extraction examples:

Never attempt to operate this appliance if it is damaged, malfunctioning, partially disassembled, or has missing or broken parts, including a damaged cord or plug.

>No extractions found.

When moving the refrigerator, be careful not to roll over or damage the power cord.

>No extraction found

Contact an authorized service center when installing or relocating the refrigerator.

>No extractions found.

Open Language Learning for Information Extraction examples:

Replace the water filter approximately every six months.

>No extractions found.

18-Dec-2020

Activities Done

- Read research paper on Automatic Knowledge Graph Creation Framework

Pending Action Items

- Identify relations (pattern) from LG Manual Questions

ToDo

- Read Research paper on automatic kg creation

An Automatic Knowledge Graph Creation Framework from Natural Language Text

Research paper: Link

Predicate mapping is essential because it can reduce the heterogeneity of the data

A hybrid combination of a rule-based approach and a similarity-based approach is presented for mapping a predicate to its corresponding predicate in a KG.

KG creation is conducted in open domains, in which prior knowledge is not provided, F1 score 50%

An Automatic Knowledge Graph Creation Framework from Natural Language Text

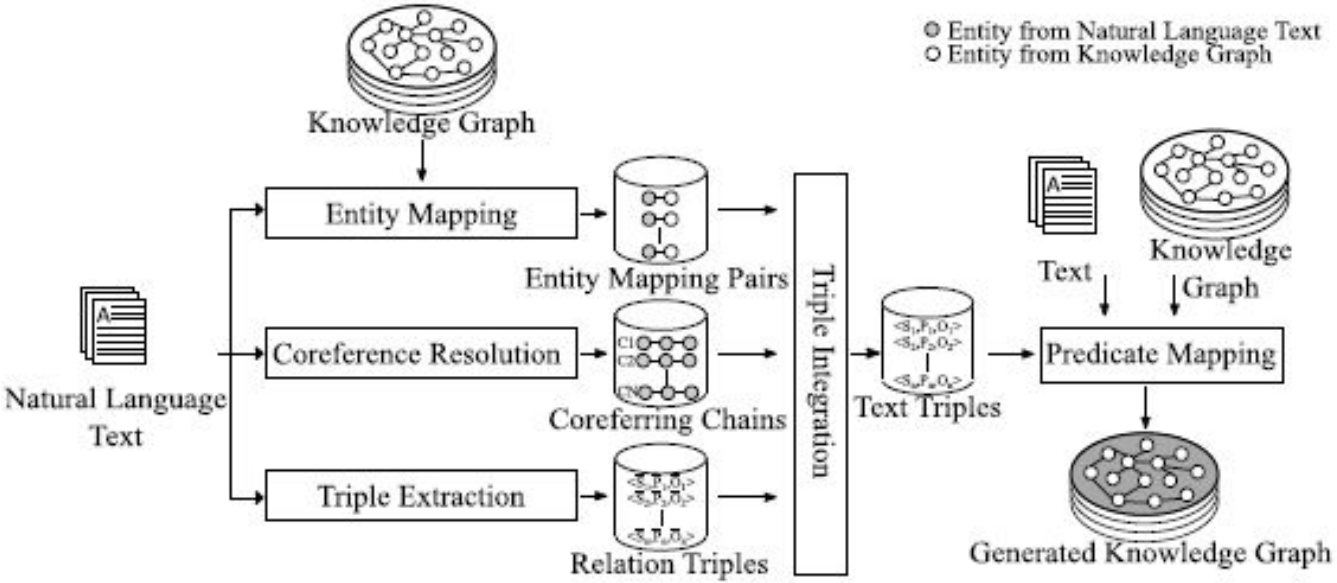


Fig. 1 Architecture of the T2KG framework.

An Automatic Knowledge Graph Creation Framework from Natural Language Text

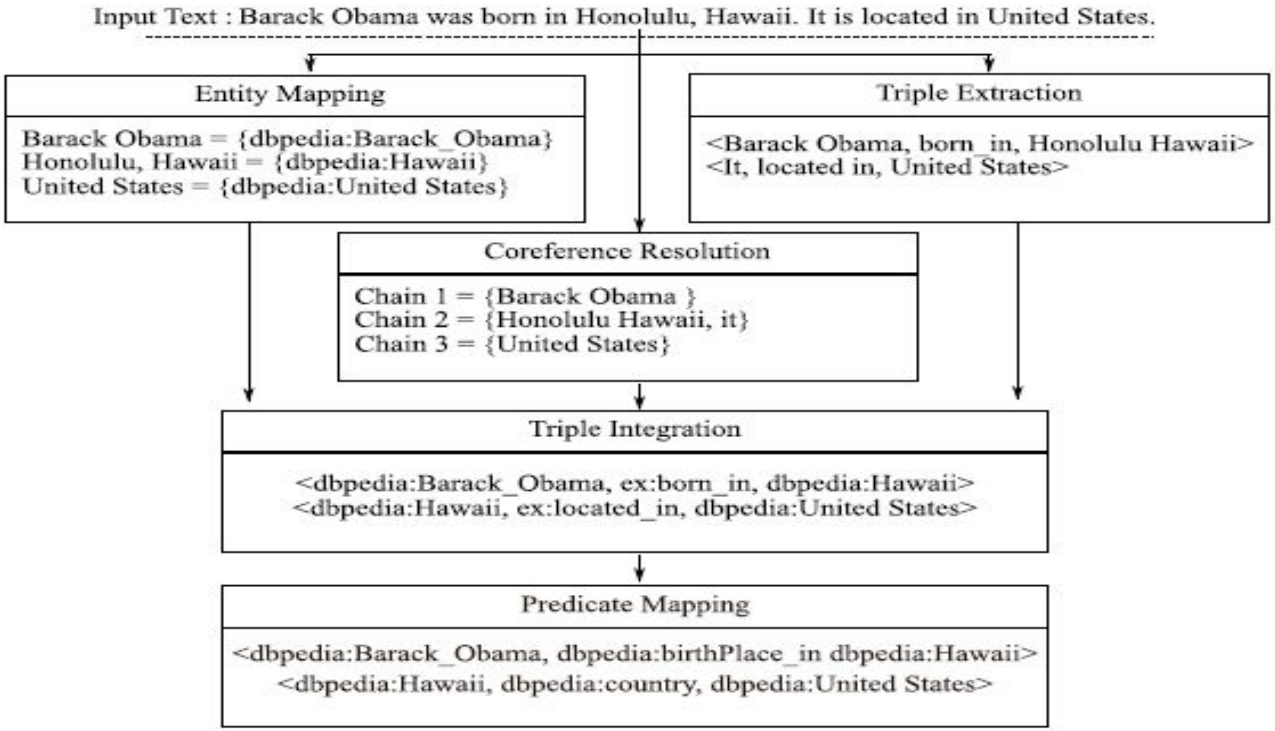


Fig. 2 Example of data flow in the T2KG framework.

An Automatic Knowledge Graph Creation Framework from Natural Language Text

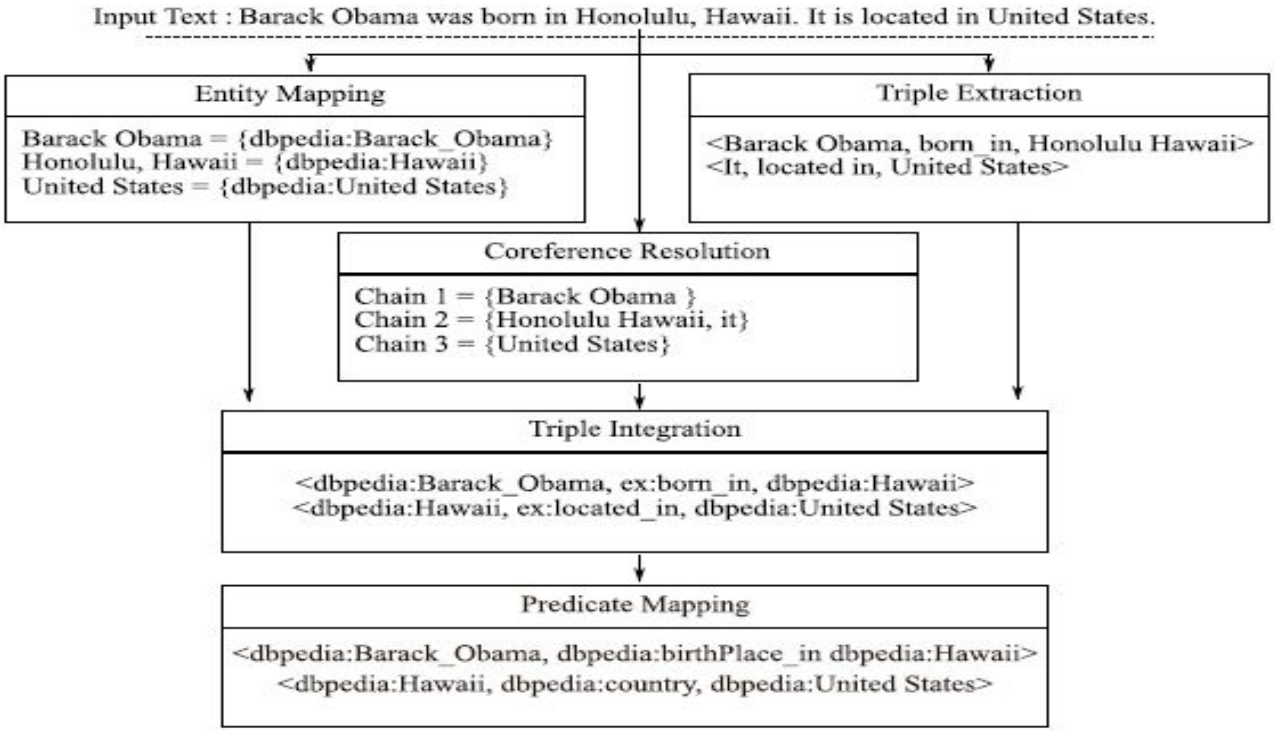


Fig. 2 Example of data flow in the T2KG framework.

An Automatic Knowledge Graph Creation Framework from Natural Language Text

Predicate Mapping:

1. Triple Enrichment (with class and data dype)
   Eg. <dbpedia: Barack Obama, dbpedia: birthPlace, dbpedia: Hawaii>
   To
   dbpedia: Person, dbpedia: birthPlace, <dbpedia: Location>. (class)
   To

   dbpedia: Person, dbpedia: birthPlace, <dbpedia: Location>. (dtype) else 'string’

Rule-Based Candidate Generation:

1. if the subject and the object of the text triple are similar to the subject and the object of the KG triple, respectively, it is assumed that the predicate of the text triple and the predicate of the KG triple are equivalent.

An Automatic Knowledge Graph Creation Framework from Natural Language Text

Similarity-Based Candidate Generation: elements of triples that have a similar context should be embedded more closely with each other in the vector space than dissimilar elements

Objective function:

$$L(\theta) = \arg\max_{\theta} \sum_{(e,c)\in BT} \Big(\log \sigma(\bar{v}_c \cdot v_e) + \sum_{(neg,c)\in BT'_{(e,c)}} \log \sigma(-\bar{v}_c \cdot v_{neg}) \Big)$$

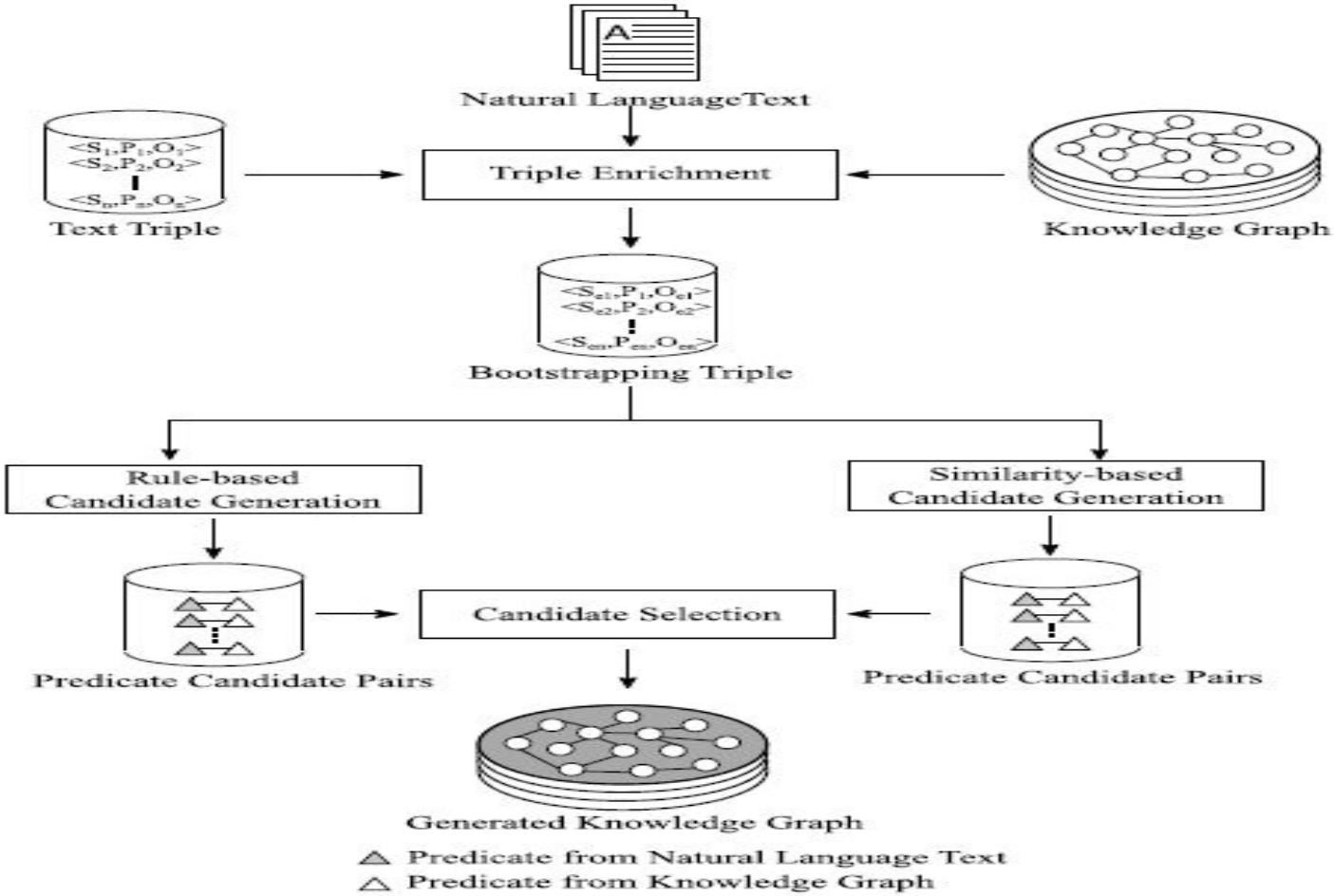


Fig. 3 Diagram of the predicate mapping component.

17-Dec-2020

Activities Done

- Completed Coding for module 3

Pending Action Items

- Identify relations (pattern) from LG Manual Questions

ToDo

- Read Research paper on automatic kg creation

Result (proposed methodology)

Relation extraction:
https://drive.google.com/file/d/1g6Dy6afJxPwX8BX7Itq1NBWdX1kGSP6a/view?usp=sharing

Final triples:
https://drive.google.com/file/d/1-YviExyIMzzg0eOxRPwlnK-vUSbDLU1I/view?usp=sharing

16-Dec-2020

Activities Done

- Completed coding for module 1 and module 2

Pending Action Items

ToDo

- Complete coding for module 3

15-Dec-2020

Activities Done

- Convert Design requirements to coding requirements
- Coding in Progress

Pending Action Items

ToDo

- Develop modules

From Design to Coding Requirements:

- Step_1:(Module 1) Entity Extraction and Coreference Resolution
- Input : Json nested key values
- Output:
  a. File1.pkl: Pickle file containing Entities extracted from text and Main_Entity from dictionary
  b. File2.txt:Intermediate file :Text file with coreference resolved for each part.
     Eg. `{Main_Entity:{Rel_1:[Coreference resolved unstructured text 1],`
     `Rel_2:[Coreference resolved unstructured text 2],`
     `Rel_3:[Coreference resolved unstructured text 3]}}`
  C. FILE3.txt: Text file with Dependency parse on co-reference resolved with Intermediate file:
     Eg.`{Main_Entity:{Rel_1:[Dependency parse from coreference resolved text 1],`
     `Rel_2:[Dependency parse from coreference resolved text 2],`
     `Rel_3:[Dependency parse from coreference resolved text 3]}}`

From Design to Coding Requirements:

- Step_2:(Module 2) :Relation Extraction
- Input : File3.txt generated from second step
- Output:
  a. Intermediate file: Process the file and extract relations for each key.
  Eg:`{Main_Entity:{Rel_1:[(e1,r1,e'1),(e2,r2,e'2),(e3,r3,e'3)...],`
  `Rel_2:[(e_1,r_1,e'_1),(e_2,r_2,e'_2),(e_3,r_3,e'_3)],`
  `Rel_3:[(e-1,r-1,e'-1),(e-2,r-2,e'-2),(e-3,r-3,e'-3)]}}`
  B. Process intermediate file (Build relation with main entity and entity within list) File4.csv
  Eg. (Main_Entity,Rel_1,e1) ,(Main_Entity,Rel_1,e'1), (Main_Entity,Rel_1,e2), (Main_Entity,Rel_1,e'2),`(e1,r1,e'1),(e2,r2,e'2),(e3,r3,e'3)`

From Design to Coding Requirements:

- Step_3:(Module 3) :Triples Creation
- Input : File4.csv
- Output:

  For every relation in File4.csv, if head_entity and tail_entity are present in Entity list, include or appropriate logic for entity match

  Else exclude

14-Dec-2020

Activities Done

- Analysis of Data
- Design to convert unstructured data to structured data
- Design for Entities/Relationships

Pending Action Items

1. Identify a end to end pipeline by PropBank SRL whch can convert text to KG to Q&A

ToDo

- Convert the design on coding requirements

Design of KG

- Design Questions
- Focus:
  1. What should be our entity , What are the relations ?
     To determine our entity we have to ask questions from
     User's point of view and what questions we are trying to solve?
     Eg. How I can download a cycle? What application I need? Questions need different approaches!
- Observations:
  - Addressing problem on sentence level, yields granular relations, the bigger picture is lost

- Use the existing structure in manuals, convert semi-structured data to fully structured data either manually, or coding of mixture of both.

Semi-structured to Structure(Eg. Main washer/Operation

{'**MAIN WASHER**': {'OPERATION':{'Using the Washer':{'WARNING': ['**To reduce the risk of fire, electric shock, or injury to persons, read the SAFETY INSTRUCTIONS before operating this appliance.**'],
'STEPS':{'Sort Laundry and Load the Washer ':['Sort laundry by fabric type, soil level, color and load size, as needed. Open the
door and load items into the washer.'],' Cleaning Products':['Add the proper amount of HE (High-Efficiency) detergent and liquid fabric softener to the Auto Dispense reservoir or the manual dispenser. If desired, use the manual dispenser to add powdered oxygen-based bleach. Do not use liquid chlorine bleach.'], ' Turn on the Washer' :['Press the Power button to turn on the washer. After a short delay, the display will illuminate and a chime will sound'],
'Cycle Buttons' :['Touch a cycle icon in the display to select a cycle. Swipe left in the display to see additional cycles.'], ' Adjust Settings':[Default settings for the selected cycle can now be changed, if desired, by selecting the options in the display. See the Control Panel section for more details.]
'Cleaning Products' :[Add the proper amount of HE (High-Efficiency) detergent and liquid fabric softener to the Auto Dispense reservoir or the manual dispenser. If desired, use the manual dispenser to add powdered oxygen-based bleach. Do not use liquid chlorine bleach']

},'Sorting Laundry': {}, 'Loading the Washer' :{}, 'Adding Cleaning Products' :{},
'Control Panel' :{}, 'Wash Cycles' :{}, 'Cycle Guide' :{}, 'Cycle Modifier Buttons' :{},
'Special Cycles/Features' :{},
'Drying Tips' :{}, 'Settings' :{}, 'Download Cycles' :{}}}}

Design

Approach:

1) Treat the top level key as entity, and rest as nested relations, followed by entities from unstructured data contained within list.
2) The entities extracted from unstructured data of a particular section, must have relations with head entity.

Advantage:

Using inherent structure present in the manual, make uses wrong entities are not linked as scope is defined.

Eg. Main washer, the top key becomes Entity, nested keys becomes relation, and entities from unstructured data , has relation with main entity.

Main Washer/Operation/Using the Washer/WARNING

“· To reduce the risk of fire, electric shock, or injury to persons, read the SAFETY INSTRUCTIONS before operating this appliance.”

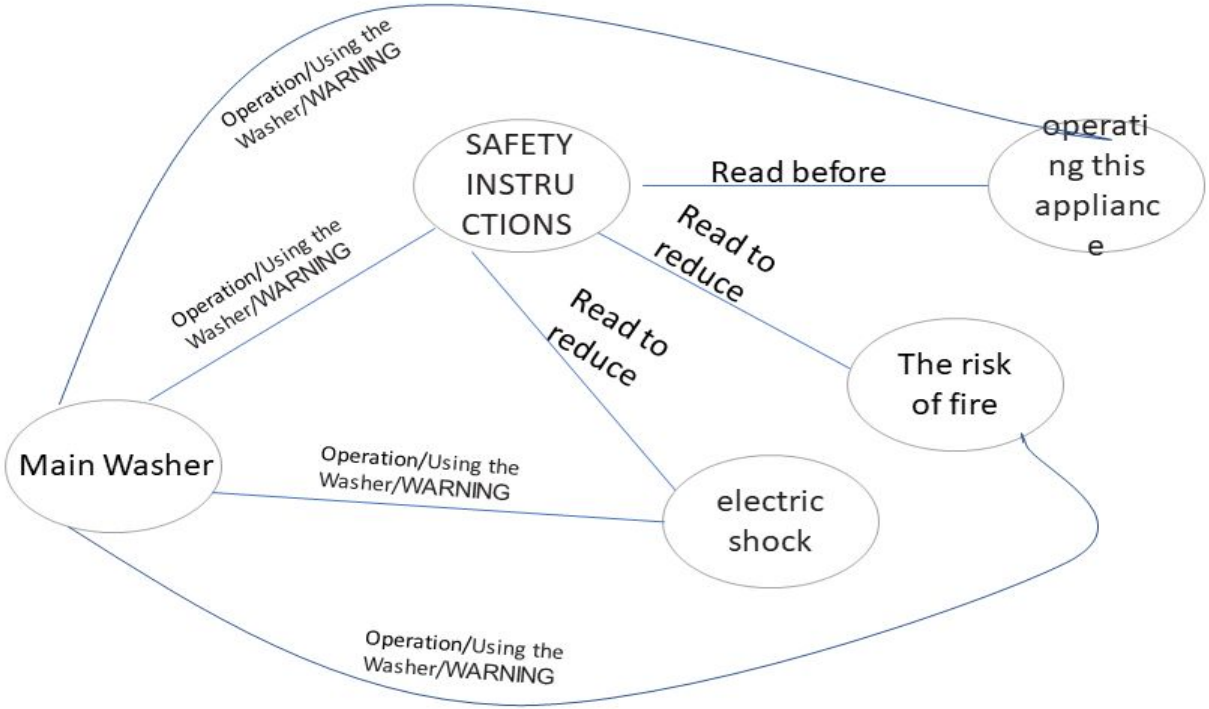
Main Washer/Operation/Using the Washer/Cleaning Products

“Add the proper amount of HE (High-Efficiency) detergent and liquid fabric softener to the Auto Dispense reservoir or the manual dispenser. If desired, use the manual dispenser to add powdered oxygen-based bleach. Do not use liquid chlorine bleach.”

Main Washer/Operation/Using the Washer/Turn on the Washer

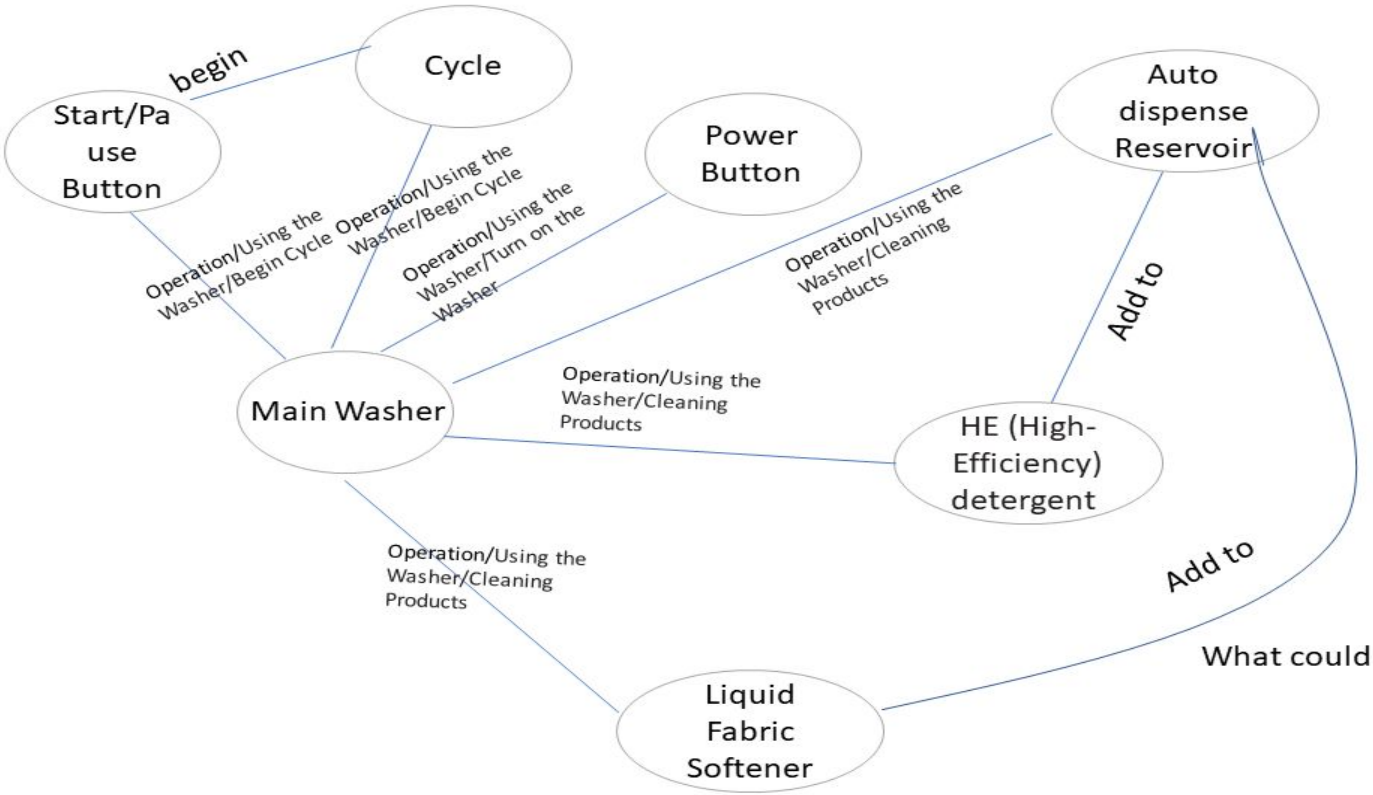
Press the Power button to turn on the washer.

Design : Plan



What could be relevant questions?

Design : Plan



What could be relevant questions?

11-Dec-2020

Activities Done

- Chunking data and apply coreference
- Triples for Operation section of Washer Manual
- Freebase paper

Pending Action Items

1. Identify a end to end pipeline by PropBank SRL whch can convert text to KG to Q&A

ToDo

- CoreNLP Co-reference can’t handle some of the hex values

Relation extraction(IE) LG Manuals

- Issue:
  - Coreference resolution was not working, for large data.
- Solution:
  - Divide the data into small chunks and apply co-reference
- From data, sentence needs to be parsed, and token get replaced with detected reference.
- Issue faced:
  - there is a mismatch of sentence parsing between coreNLP and spacy,
  - Solution:
  - so for each chunk coreNLP parser is used to parse sentences. However, still we have to use spacy sentence splitter as CoreNLP can’t work with large data, so chunk the data using spacy splitter, then for each chunk, apply coreNLP to split and for coreference resolution.
- Issues:
  - It still can’t handle @™, some hexadecimal values, currently these exceptions are by passed

Relation extraction(IE) LG Manuals

- For washer manual Operation part triples are created for KG:
- https://drive.google.com/file/d/1-YviExyIMzzg0eOxRPwlnK-vUSbDLU1I/view?usp=sharing

FREE BASE

- Freebase database by Metaweb was a large collaborative knowledge base consisting of data composed mainly by its community members.
- It was an online collection of structured data harvested from many sources, including individual, user-submitted wiki contributions.
- Google's Knowledge Graph was powered in part by Freebase
- Object, Type, Properties

10-Dec-2020

Activities Done

- Dependency parse based IE enriched for relation extraction from user manual

Pending Action Items

1. Identify a end to end pipeline by PropBank SRL whch can convert text to KG to Q&A

ToDo

- Co-reference for longer sentences not working/Need to check

Relation extraction(IE) LG Manuals

- Apart from openie based IE, dependency parsing based Information extraction is enriched to extract relations from unstructured data.
- It treats verb with preposition as relations, and align information around it as entities
- Need to investigate how to handle manners of the verb such as ‘always’ in the graph
- Results are available at :
- https://drive.google.com/file/d/100QnPnC8qobJvUQPMryfHNktN0CCNYEj/view
- Dependency based IE yields better relations than opnie

9-Dec-2020

Activities Done

- Unstructured data extracted from LG Manual
- And KG constructed for LG manuals
- Extraction requirements are shared with Senthil

Pending Action Items

1. Check the end to end pipeline for SRL to KG to Q&A

ToDo

- Co-reference for longer sentences not working/Need to check
- Dependency parsing based IE

Relation extraction(IE) LG Manuals

- Unstructured Data extracted using apache tika
- Basic preprocessing (remove \n\n\n\n… , symbols • etc)
- Split text into lines using spacy tokenizer
- Available at : https://drive.google.com/file/d/1ItAMT6wZx3B4ASVcq8Rf1MTFw_bOAFTA/view?usp=sharing
- Total number of lines 1778
- Issues faced
  - Stanford coreNLP can’t process large data file(connection error file file with more than 50 sentence and “can't handle incoming annotation”
  - So temporarily Core-NLP co-occurrence resolution turned off, till we find better way
- The results (KG triples) are stored in https://drive.google.com/file/d/1-YviExyIMzzg0eOxRPwlnK-vUSbDLU1I/view?usp=sharing

Relation extraction(IE) LG Manuals: Requirements

Observation: IE tool does not work for Imperative sentence:
Eg:
“Periodically check the hoses for cracks, leaks, and wear, and replace the hoses every five years.”
:No information extracted

It is also difficult to parse imperative sentence

Annotations needed

8-Dec-2020

Activities Done

- Analysis LG Manuals Data
- Explore some tools to parse PDF to structured format

Pending Action Items

1. Check the end to end pipeline for SRL to KG to Q&A

ToDo

- Requirements needs to be discussed and specified

Relation extraction(IE) LG Manuals: Requirements

1) Data is not Unstructured , rather Semi-structured
2) Section-subsection-subsubsection (Hierarchical)/Tables/Unstructured
   Document Manual:
   http://gscs-b2c.lge.com/downloadFile?fileId=fQ3Fk73pg7m3CDOaYXczCg
   Eg. Page 14. **INSTALLATION->Connecting the Water Lines** could be a relation, each subsection could represent an entity.Each entity might have some description.
   Eg. Page 9: **PRODUCT OVERVIEW->Parts** could be relations, each part could be an entity
   Accessories->Included Accessories could be a relationship as well
   Eg. Page 8: Unstructured text

3)Data also has tables: Page 11; Extract relations from tables are separate set of problems
4) Manual has images, so preprocessing required
5)Requirement needs to be clearly specified eg. what Questions are we expecting to solve from this manual.
6) May be some annotations on human level, for better specification of requirements, as the manual is a mixture of different mixture of structured and unstructured data

Relation extraction(IE) LG Manuals

7) Based on the requirements, different problems need to be dealt separately, eg. Unstructured data, Relations in the form of structured hierarchy, Tables, Images

- The semi-structured data need to be converted to structured data (JSON) example
- Some Tools:

pip install **PyPDF2**

```
f='/work/LGSI/KG/Manuals/MFL69475588_G+Best_HP_EUS_170221_v2_p.pdf'
reader = PyPDF2.PdfFileReader(f)
reader.documentInfo
{'/CreationDate': "D:20170221195631+09'00'", '/Creator': 'Adobe InDesign CC 2017 (Windows)', '/ModDate': "D:20170221200451+09'00'", '/Producer': 'Adobe PDF Library 15.0', '/Trapped': '/False'}
```

to extract text from specified page:

```
reader.getPage(10-1).extractText()
```

- Not that good, tab separated strings are getting combined into one

Relation extraction(IE) LG Manuals

Another tool is:
Apache Tika
Pip install tika #to parse a PDF
From tika import parser
parsed= parser.from_file(f)
Parsed['content']

7-Dec-2020

Activities Done

- Developed New IE from parse tree (To prune relations)

Pending Action Items

1. Check the end to end pipeline for SRL to KG to Q&A

ToDo

- Prune/Standardise Relations

Relation extraction(IE) from dependency parse tree

Algorithm to :

For sentenece_parse_sub_tree in parse_tree:
    for token in sentence_parse_sub_tree:
        if no_child is 'nsubj': subject
            skip the token
        else:
            recursive traverse child with nsubj for subject
            find_predicates and tail pair(for a verb_token)

def find_predicate_and_tail pair(for a verb token):
    if children has dobj/'attr/pobj', verb is relation, dobj/'attr' is object
    if children has 'advcl'/'prep' keep recursively adding children of verb to get the relation till only 'pobj'/'ccomp' #
    track pobj /'ccomp' recursively, this is tail
Now include (sub, prel/drel/arel/advcl_rel, pobj/dobj/attr/advcl)
Also include (attr/dobj, prel/arel, advcl/pobj)

IE

Link for IE:

https://drive.google.com/file/d/16I_kHY3LJOfgRUnvaYCOFHsedQoo5KbG/view?usp=sharing

4-Dec-2020

Activities Done

- Reduced number of entities by Head word matching
- Entity,Entity Alias dictionary created

Pending Action Items

1. Check the end to end pipeline for SRL to KG to Q&A

ToDo

- Prune/Standardise Relations

Text to KG

Challenges:

- including dependency parsing adds more entities .[A elation is added only if there is intersection between extracted tails by IE and entities!]
- it was observed that dependency parsing creates a vaste number of redundant entities as well. Eg. , Tag ‘Other’ means extracted through Dependency parsing, not using NER
- "India 's": 'Other',
- 'operating': 'Other',
- "India 's oldest operating port": 'Other',
- 'riverine': 'Other',
- 'its sole major riverine port': 'Other',
- 'the " cultural capital of India " for the city \'s historical architectural significance': 'Other',
- "the city 's": 'Other',
- "the city 's historical architectural significance": 'Other'

Text to KG

Observation:

Some of these phrases have same parent(head_ids), some are (denoted by head ids) are children of other entities.

Solution:

Track head word_ids and child_word_ids, if a string (denoted by head_word_ids) is a children of any other string(denoted by child_ids)

we prune those entity string

Text to KG

Entity Alias dictionary:

- Entities as Keys
- Remove article from keys, that is one alias name
- If there is sufficient overlap (maximum string length, Fuzz ratio) of an IE tail , with an Entity or it's alias, that also becomes alias name of that entity.
- Store it in a dictionary

eg.
```
{
    'the 2011 Indian census': ['the 2011 Indian census', '2011 Indian census'],
    'the " cultural capital of India " for the city \'s historical architectural significance': ['the " cultural capital of India " for the city \'s historical architectural significance', '" cultural capital of India " for city \'s historical architectural significance', 'cultural capital of India '], …..}
```

Output

	Type	Entity 1	Relationship	Type	Entity2
2	PERSON	Kolkata	is city in	GPE	India
3	OTHER	the city	financial hub of	O	eastern India
4	PERSON	Kolkata	is seventh-most populous city in	GPE	India
5	PERSON	Kolkata	is seventh-most populous city Ac	O	2011 Indian census
6	PERSON	Kolkata	is known as	O	cultural capital of India
7	PERSON	Kolkata	population in	O	Kolkata Metropolitan Area
8	PERSON	Kolkata	is	O	seventh-most populous city in India
9	OTHER	the city	is	O	prime business
10	PERSON	Kolkata	is city According to	O	2011 Indian census
11	OTHER	the city	Located on	O	eastern bank of Hooghly River
12	PERSON	Kolkata	is populous city According to	O	2011 Indian census
13	PERSON	Kolkata	is seventh-most city According tc	O	2011 Indian census
14	PERSON	Kolkata	population of	O	over 14.1 million residents
15	PERSON	Kolkata	is seventh-most city in	GPE	India
16	OTHER	the city	hub of	O	eastern India
17	PERSON	Kolkata	is populous city in	GPE	India
18					

Text to KG

Relations need to be standardise

Ideas?

Need access to server

3-Dec-2020

Activities Done

- Dependency Parsing to extract more entities
- Entity Matching and Pruning

Pending Action Items

1. Check the end to end pipeline for SRL to KG to Q&A

ToDo

- Prune Entities

Text to KG

- Develop dependency parse tree, with sentence no, word no
- Space parser output:

token.text, token.i, token.tag_, token.pos_, token.ent_type_, token.dep_, token.head, token.head.text, token.head.pos_ [child for child in token.children]

Eg.

Important 0 JJ ADJ amod advances advances NOUN []
advances 1 NNS NOUN nsubjpass made made VERB [Important]
have 2 VBP AUX aux made made VERB []
been 3 VBN AUX auxpass made made VERB []
made 4 VBN VERB ROOT made made VERB [advances, have, been, in,), .]
in 5 IN ADP prep made made VERB [treatment]
the 6 DT DET det treatment treatment NOUN []

Text to KG

- Develop dependency parse tree, with sentence no, word no, children in form of dictionary
- Dependency Parse Tree:

https://docs.google.com/document/d/1wphQm6TfrR8hTcHoZcq72e35ozPAhxOXXIzNCb7Bcgw/edit?usp=sharing

- Apart from NERs selected by Spacy, to extract additional NER:

1. Target the tag with tag as ‘NN’,
2. Recursively traverse the tree to find out all the children of selected node. Join tokens , to form new entities (Exclue ‘nsubj’ and ‘cc’ tag)
3. Now from Stanford IE relation extraction, for each tail entity of IE tool, compare the tail with entities
4. (longest common subsequence), score them as per the maximum length, sort

Text to KG

Type	Entity 1	Relationship	Type	Entity2
Other	Kolkata	is populous city Accordin	O	2011 Indian census
Other	city	hub of	Other	eastern India
Other	Kolkata	is city According to	O	2011 Indian census
Other	Kolkata	population in	O	Kolkata Metropolitan Area
Other	city	is	O	prime business
Other	city	financial hub of	Other	eastern India
Other	Kolkata	is	O	seventh-most populous city in India
Other	Kolkata	is known as	O	cultural capital of India
Other	Kolkata	is seventh-most populou	O	2011 Indian census
Other	Kolkata	population of	O	over 14.1 million residents
Other	Kolkata	is seventh-most city Acc	O	2011 Indian census
Other	city	Located on	O	eastern bank of Hooghly River

2-Dec-2020

Activities Done

- Coreference Resolution added in unstructured data to text
- Additional requirements for pruning redundancy

Pending Action Items

1. Check the end to end pipeline for SRL to KG to Q&A

ToDo

- Prune Entities

Text to KG

- The current implementation , though incorporates coreference resolution, it was not reflected at output.
- Original: It is the prime business, commercial, and financial hub of eastern India and the main port of communication for the North-East Indian states, as well as having the third-largest urban economy of India.
- To replace: It | at: 1 2 With: the city
- Result: the city is the prime business , commercial , and financial hub of eastern India and the main port of communication for the North-East Indian states , as well as having the third-largest urban economy of India .

Text to KG

At output:

final:

Located on the eastern bank of the Hooghly River , the city is approximately 80 kilometres (50 mi) west of the border with Bangladesh . It is the prime business , commercial , and financial hub of eastern India and the main port of communication for the North-East Indian states , as well as having the third-largest urban economy of India . According to the 2011 Indian census , Kolkata is the seventh-most populous India in India , with a population of 4.5 million residents within the city limits , and a population of over 14.1 million residents in the Kolkata Metropolitan Area , making it the third-most populous metropolitan area in India . The Port of Kolkata is India oldest operating port and its sole major riverine port . Kolkata is known as the " cultural capital of India " for the city 's historical and architectural significance .

Text to KG

Analysis:

When there are multiple corefernces in a single sentence, such problem is occuring

Fix provided

Relation Extraction (stanford IE): Reason for redundancy:

Kolkata	is known as	cultural capital of India for city
Kolkata	is known as	cultural capital of India
Kolkata	is known as	capital of India
Kolkata	is known as	cultural capital for city
Kolkata	is known as	capital for city
Kolkata	is known as	cultural capital
Kolkata	is	known
Kolkata	is known as	capital of India for city
Kolkata	is known as	capital

Requirements

Issue 1: Current KG includes all the relations if there is a partial match between any NER entity and the tail of IE extracted data, if no matched entity then also include results from IE: Results redundancy

Solution:

Include only Entities if there is full match (but that results in very less relations), So increase number of entities and include IE if it's a proper match, order to do so:

1) Include dependency parsing, to track roots of NER
2) Incorporates multiple NER, trimm same relations with alias names

Kolkata	is seventh-most populous city in	India
Kolkata	is seventh-most city According to	2011 census
Kolkata	is seventh-most populous city According to	2011 Indian census

s of

1-Dec-2020

Activities Done

- Knowledge Graph created

Pending Action Items

1. Convert text to frames.

ToDo

- Find some other open source implementation

Text to KG

Further Analysis :

- It turns out, that the shell script intermediate file creation was not working in google colab, but transferring the same code in a linux based system yields expected output.
- Input unstructured file:

> Located on the eastern bank of the Hooghly River, the city is approximately 80 kilometres (50 mi) west of the border with Bangladesh. It is the prime business, commercial, and financial hub of eastern India and the main port of communication for the North-East Indian states, as well as having the third-largest urban economy of India. According to the 2011 Indian census, Kolkata is the seventh-most populous city in India, with a population of 4.5 million residents within the city limits, and a population of over 14.1 million residents in the Kolkata Metropolitan Area, making it the third-most populous metropolitan area in India. The Port of Kolkata is India's oldest operating port and its sole major riverine port. Kolkata is known as the "cultural capital of India" for the city's historical and architectural significance.

Text to KG

Output is available at :https://drive.google.com/file/d/1-YviExyIMzzg0eOxRPwlnK-vUSbDLU1I/view?usp=sharing

Text to KG

Another Implementation

:https://github.com/dstlry/dstlr (scala implementation)

Setting up (java+sbt) issue while setting up

```
[error]         at sbt.io.Using.apply(Using.scala:22)
[error]         at sbt.MainLoop$.runWithNewLog(MainLoop.scala:104)
[error]         at sbt.MainLoop$.runAndClearLast(MainLoop.scala:59)
[error]         at sbt.MainLoop$.runLoggedLoop(MainLoop.scala:44)
[error]         at sbt.MainLoop$.runLogged(MainLoop.scala:35)
[error]         at sbt.StandardMain$.runManaged(Main.scala:138)
[error]         at sbt.xMain.run(Main.scala:89)
[error]         at xsbt.boot.Launch$$anonfun$run$1.apply(Launch.scala:111)
[error]         at xsbt.boot.Launch$.withContextLoader(Launch.scala:131)
[error]         at xsbt.boot.Launch$.run(Launch.scala:111)
[error]         at xsbt.boot.Launch$$anonfun$apply$1.apply(Launch.scala:37)
[error]         at xsbt.boot.Launch$.launch(Launch.scala:120)
[error]         at xsbt.boot.Launch$.apply(Launch.scala:20)
[error]         at xsbt.boot.Boot$.runImpl(Boot.scala:56)
[error]         at xsbt.boot.Boot$.main(Boot.scala:18)
[error]         at xsbt.boot.Boot.main(Boot.scala)
Project loading failed: (r)etry, (q)uit, (l)ast, or (i)gnore? [error] java.io.IOError: java.lang.RuntimeException: /pack
ages cannot be represented as URI
[error] Use 'last' for the full log.
```

30-Nov-2020

Activities Done

- On knowledge graph creation, analysis of bottle neck faced.

Pending Action Items

1. Convert text to frames.

ToDo

- Find some other open source implementation

Text to KG

Analysis regarding `connection error` while using stanfordnlpcore tokenizer:

It seems like, the underlying java program is exhausting the memory when using the tokenizer, switching to spacy tokenizer solves the problem also generate tokens consistent with stanford-core-nlp.

Relation_extractor.py invokes a shell script ./process_large_corpus.sh', with input file name and output file name, which further calls a subprocess main.py. However,the analysis show no file is being created in the process.

The issue also reported to the author of the repository, no response.

27-Nov-2020

Activities Done

- Worked on text to KG(issues)

Pending Action Items

1. Convert text to frames.

ToDo

- Find some other open source implementation

Text to KG

Analysis shows:

The program can't handle high-phens : Our test text has a sentence: It is the prime business, commercial, and financial hub of eastern India and the main port of communication for the North-East Indian states, as well as having the third-largest urban economy of India.

- Stafordcorenlp tokenizer used for coreference resolution which splits around hyphen, while nltk does not, both are used so index mismatch
- Changing the tokenizer raises another connection error which needs to be analysed

Text to KG

Analysis shows:

Removal of tokens with hyphens, let the process flow, but issue in the subsequent steps as it seems , a file is missing.

26-Nov-2020

Activities Done

- Worked on text to KG(issues)

Pending Action Items

1. Convert text to frames.

ToDo

- Find some other open source implementation

Text to KG

From Github implementation

:https://github.com/varun196/knowledge_graph_from_unstructured_text

1. Format:

	Type	Entity 1	Relationship	Type	Entity2
2	ORG	Paramount Pictures	obtained	O	rights to novel for price of $ 80 000
3	ORG	Godfather	is	NORP	American
4	ORG	Godfather	is	DATE	1972 crime film
5	ORG	Paramount Pictures	obtained	O	rights for price of $ 80 000
6	ORG	Paramount Pictures	obtained	O	rights for price
7	ORG	Paramount Pictures	obtained	O	rights to novel
8	ORG	Godfather	is	DATE	1972 American crime film
9	ORG	Paramount Pictures	obtained	O	rights to novel for price
10	ORG	Paramount Pictures	obtained	O	rights
11	ORG	Godfather	is	NORP	1972 American crime film

Text to KG

- Apparently it seems to work! (spacy NER+stanford corenlp Corefernce Resolution + stanford IE)
- However faced issue while tried to reproduce with different data:

```
Traceback (most recent call last):
  File "knowledge_graph.py", line 292, in <module>
    main()
  File "knowledge_graph.py", line 287, in main
    doc =
resolve_coreferences(doc,stanford_core_nlp_path,named_entities,verbose)
  File "knowledge_graph.py", line 217, in resolve_coreferences
    result = coref_obj.resolve_coreferences(corefs,doc,ner,verbose)
  File "knowledge_graph.py", line 200, in resolve_coreferences
    replaced_sent = words[i] + " "+ replaced_sent
IndexError: list index out of range
```

25-Nov-2020

Activities Done

- Installed docker , and built environment and and run the code for Sample reference

Pending Action Items

1. Convert text to frames.

ToDo

- Convert text to KG

Semantic Role Labeling for Knowledge Graph Extraction from Text

- Docker
- Run Docker Instance
- Open terminal(type):
- git clone https://github.com/TakeFiveSRL/TakeFiveSRL.git
- cd TakeFiveSRL
  - docker-compose build
  - docker-compose up corenlpsrl
- The image corenlpsrl is running. Now open another terminal and type:
- Cd SRL_Example
  - Vim Dockerfile   ##put your sentence
  - docker network ls ##to see the available networks
  - docker build -t python-barcode .
  - docker run --network=takefivesrl_srlnet python-barcode

Semantic Role Labeling for Knowledge Graph Extraction from Text

Sentence: “Tom ate an apple” More Examples

Output:

verb: ([u'nsubj', u'ate-2', u'Tom-1'], 'agent',
verb: ([u'dobj', u'ate-2', u'apple-4'], 'undergoer',

Sentence:”New Delhi is the capital of India. India has a population on 1 billion. India has 29 states and 7 union territories.”

Output:

verb: ([u'nsubj', u'has-2', u'India-1'], 'agent'

verb: ([u'dobj', u'has-2', u'states-4'], 'undergoer'

Text to KG

Another implementation from unstructured text to KG is available at link (Analysis required for feasibility)

24-Nov-2020

Activities Done

- Updated code and Data in github/google drive
- Read paper Sample reference. For building knowledge graph
- Installed Docker(faced issue)
- PPT for KT shared

ToDo

- Read paper Sample reference. For building knowledge graph

Pending Action Items

1. Convert text to frames. Sample reference.

Semantic Role Labeling for Knowledge Graph Extraction from Text

This paper introduces TakeFive :A Semantic Role Labeling Algorithm that combines:Github Link (Requires Docker)

1)CoreNLP for dependency parsing

2)VerbNet for verb sense disambiguation, syntactic frame

3) FrameNet: contains frames, which describe a situation, state or action. Each frame has semantic roles (“frame elements”) that are much more semantically detailed than VerbNet ones.

4)Framester: a KG hub between several predicate oriented linguistic resources such as FrameNet, WordNet, VerbNet

Semantic Role Labeling for Knowledge Graph Extraction from Text

Installed Docker. Run the code:

```
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL\SRL_Example> cd ..
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL> docker-compose build
corenlpsrl uses an image, skipping
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL> docker-compose up corenlpsrl
Pulling corenlpsrl (motiz88/corenlp:)...
ERROR: Get https://registry-1.docker.io/v2/motiz88/corenlp/manifests/latest: Get https://auth.doc
ker.io/token?scope=repository%3Amotiz88%2Fcorenlp%3Apull&service=registry.docker.io: net/http: TL
S handshake timeout
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL> cd .\SRL_Example\
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL\SRL_Example> docker build -t python-barcode .
[+] Building 12.9s (5/8)
 => [internal] load build definition from Dockerfile                                    1.0s
 => => transferring dockerfile: 32B                                                     0.0s
 => [internal] load .dockerignore                                                       1.4s
 => => transferring context: 2B                                                         0.1s
 => ERROR [internal] load metadata for docker.io/library/python:2.7                    10.0s
 => ERROR [1/4] FROM docker.io/library/python:2.7                                       1.2s
 => => resolve docker.io/library/python:2.7                                             1.2s
 => [internal] load build context                                                       0.7s
 => => transferring context: 24.56kB                                                    0.0s
------
 > [internal] load metadata for docker.io/library/python:2.7:
------
------
 > [1/4] FROM docker.io/library/python:2.7:
------
failed to solve with frontend dockerfile.v0: failed to build LLB: failed to load cache key: faile
d to do request: Head https://registry-1.docker.io/v2/library/python/manifests/2.7: dial tcp: loo
kup registry-1.docker.io on 192.168.65.1:53: no such host
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL\SRL_Example> docker run --network=srldocker_srlnet py
thon-barcode
Unable to find image 'python-barcode:latest' locally
docker: Error response from daemon: Get https://registry-1.docker.io/v2/: dial tcp: lookup regist
ry-1.docker.io on 192.168.65.1:53: no such host.
See 'docker run --help'.
PS C:\Users\Anindya\Desktop\tt\TakeFiveSRL\SRL_Example>
```

Thematic Roles (Agent, Theme, Experiencer) are hard to create

Alternative to thematic roles more generalized:

1. PropBank(Fewer roles: generalized semantic roles)
2. FRameNet(Define roles specific to a group of predicates)

Roles in PropBank are specific to a verb

Role in FrameNet are specific to a frame: a background knowledge structure that defines a set of frame-specific semantic roles, called frame elements

23-Nov-2020

Activities Done

- Coding for interactive mode completed

Pending Action Items

1. Convert text to frames. Sample reference.

ToDo

- Read paper Sample reference. For building knowledge graph

Demo:

```
Please Ask Question:
who is the president of the united states?
george crum , susanna styron , john hock , bronx style bob  ,
darren manzella
Please Ask Question:
What is the capital of India?
new delhi
Please Ask Question:
what is the capital of england ?
london
Please Ask Question:
who is the director of the movie titanic ?
james cameron

Please Ask Question:
```

20-Nov-2020

Activities Done

- Code Analysis to implement interactive is in progress
- Coding in in progress

Pending Action Items

1. Convert text to frames. Sample reference.

ToDo

- interactive mode_planning to finish it by today
- Read paper Sample reference. For building knowledge graph

Code Analysis_ test_main.py

- Save TEXT,ED, [mid_dic, mid_num_dic ,pre_dic, match_pool, pre_num_dic,index_names, tuple_topic] as pickle file
- TEXT,ED, are fields with vocab information
- Index_names: just like one entity can have multiple aliases, one partial name_match from a question might represent different entities index_names stores key as partial_name, value as list of entities
- head_mid_idx contains detected_named entities, if match found or partial named entity(multiple)
- dete_tokens_list contains detected_head_enty if match found or entire string exept wh-part
- filter_q contains question without wh_part
- head_mid_idx contains [[(entity1:name1),(entity2:name2)],[(entity3:name3)]...]# there could be mulple names/multiple entities corresponding to a combination of names

19-Nov-2020

Activities Done

- Training Complete: Predicate Learning Model
- Training Complete: Head Entity Learning Model
- Test Detection: Head Entity and Predicate
- Result reproduced

Pending Action Items

1. Convert text to frames. Sample reference.
2. Dynamic Graph Convolutional Networks for Entity Linking

ToDo

- Add interactive mode
- Read paper Sample reference. For building knowledge graph

Entity Representation Learning Model

- Model is trained to learn Predicate representation from questions, stored at

  trained_Model_Head_Learning

  ```
  Early Stopping. Epoch: 60, Best Dev Accuracy:
  0.6400479616306954
  ```

  Log

Predicate Representation Learning Model

- Model is trained to learn Predicate representation from questions, stored at

  trained_Model_Predicate_Learning

  ```
  Early Stopping. Epoch: 37, Best Dev accuracy:
  0.8189027201475334, loss: 0.0010696391624631682,
  ```

  Log

Joint detection

Results:

Head_entity: 0.7539539816479919,

JOINT HEAD AND PREDICATE:0.8164799188453913,

all acc 0.67962373772306 (HEAD <PREDICATE> TAIL, joint)

(Paper claims 0.749 HEAD Entity detection)

18-Nov-2020

Activities Done

- Logic added to save embeddings, after TransE Train (Tested for 1 epoch)
- Downloaded pre-trained Graph Embedding (TransE)
- Training Head Entity Detection (HED) model: Completed
- Code uploaded to Github repository

Pending Action Items

1. Convert text to frames. Sample reference.
2. Dynamic Graph Convolutional Networks for Entity Linking

ToDo

- Predicate and headentity learning model needs to be analysed and trained
- Detect pairs to answer questions

Head Entity Detection model

- Model is trained to detect head-entities from questions, stored at

  HED-Trianed-model.ckpt

  ```
  Dev Recall:  94.762902% Precision:  91.411776% F1 Score:
  93.057179%

  Early Stopping. Epoch: 92, Best Dev Recall: 0.950676061702532 8
  ```

  Log

17-Nov-2020

Activities Done

- Code Analysis
- Running transE algorithm to obtain the Graph embeddings (Node embeddings and Predicate Embeddings)

Pending Action Items

1. Convert text to frames. Sample reference.
2. Dynamic Graph Convolutional Networks for Entity Linking

ToDo

- Head Entity Detection (HED) model needs to be trained
- Predicate and headentity learning model needs to be analysed and trained

Graph Embedding TransE Algorithm

- TransE model to train graph embeddings are not present in the mentioned repository,

  Authors suggests to download Pre-Trained embeddings , from:

  https://www.dropbox.com/s/o5hd8lnr5c0l6hj/KGembed.zip

- To train any KG on TransE Graph algorithm ; tensorflow -v1 based implementations are available from original paper at

  https://github.com/ZichaoHuang/TransE (Some modifications required)

  Inputs: relation2id.txt, entity id, transE_train.txt, transE_text.txt, transE_valid.txt

16-Nov-2020

Activities Done

- Working towards replicating authors results from the paper ***Knowledge Graph Embedding Based Question Answering***
- Downloaded freebase KG and Simple question dataset
- Analysis Data and Preprocessed data

ToDo

- Reproduce the results
- Code analysis

Pending Action Items

1. Dynamic Graph Convolutional Networks for Entity Linking

- GitHub implementation paper (***Knowledge Graph Embedding Based Question Answering***)

  https://github.com/xhuang31/KEQA_WSDM19

- Freebase no longer supports API, Data downloaded, Data Preprocessed
- Data Format:
  a. KG: fb2m: 10843106 records/ fb5m:12010500 records , triplet formats with machine id
  b. Sample record: 'www.freebase.com/m/0pbnwg7\twww.freebase.com/people/person/place_of_birth\twww.freebase.com/m/0glcrf\n
  c. 'FB5M.name.txt' (5507279 #5 million entities): Contains name of the entity corresponding to machine ids.
  d. Simple Question datasets: (<head machine id> <predicate> <tail machine id> <question>):"annotated_fb_data_train.txt"(number of records: 75910), "annotated_fb_data_test"(number of records: 21678), "annotated_fb_data_valid.txt" (number of records: 10845)
- Pretrained model is not available, pretrained graph embeddings are available to download

- After Preprocessing :
- CleanedFB.txt (number of records: 7188619, sample data: 'm.0n1vy1h\tm.05zppz\tpeople.person.gender\n'
- Entity2id: (number of records: 647657 sample records: 'm.06x1gc9\t0\n' ,'m.01692xz\t1\n'
- names.trimmed:(number of records: 599887, sample: 'm.0f8vjgd\tskyshaper\n')
- relation2id.txt(number of records: 4641, sample records: 'law.invention.inventor\t0\n', 'sports.sports_league_season.awards\t4640\n'
- transE_train.txt/transE_test.txt, transE_valid.txt (sample:'m.03byqr1\tm.033th\tcvg.computer_videogame.cvg_genre\n', 'm.0drj5kc\tm.01dt_1\tmusic.recording.artist\n')

12-Nov-2020

Activities Done

- Variational Reasoning for Question Answering with Knowledge Graph (Read paper in detail)

Pending Action Items

1. Dynamic Graph Convolutional Networks for Entity Linking

ToDo

- Read Research Papers in detail

- Review of Variational Reasoning for Question Answering with Knowledge Graph:
- Two issues in VRN (Variational Reasoning Network) is addressed in this paper:

  1)Expressions in questions are noisy (for example, typos in texts, or variations in pronunciations)(traditional string matching, rules are not useful or expensive)

  2)Many questions require multi-hop logic reasoning over the knowledge graph to retrieve the answers (“Who wrote the paper titled..?” (paper_title, authored_by, author_name); Who have co-authored the paper with?(author, authored, paper),(paper, authored_by, author)

  3) traditional model is not trained end-to-end and errors may be cascaded.

- To solve the problem:

  Unified deep learning architecture, and an end-to-end variational learning algorithm which can handle noise in questions, and learn multi-hop reasoning simultaneously.

- Typically, the **training data** for QA system is provided as question-answer pairs, where **fine grained annotation (No mention of head entity or predicate)** of these pairs are not available.
- the Exact logic reasoning steps along the knowledge graph leading to the answer.
- Formulation:
  - Since no, topic entity annotation: two probabilistic models, identifying topic entity given a question, and a model for reasoning over knowledge graph:

$$\max_{\theta_1,\theta_2} \quad \frac{1}{N}\sum_{i=1}^{N}\log\left(\sum_{y\in V(\mathcal{G})} P_{\theta_1}(y|q_i)P_{\theta_2}(a_i|y,q_i)\right)$$

$$\begin{aligned} P_{\theta_1}(y|q) &= \text{softmax}\left(W_y^\top f_{\text{ent}}(q)\right) \\ &= \frac{\exp(W_y^\top f_{\text{ent}}(q))}{\sum_{y'\in V(\mathcal{G})}\exp(W_{y'}^\top f_{\text{ent}}(q))}, \end{aligned}$$

- Formulation of first part is straight forward, RNN converts questions to a d-dimensional vector, followed by feed-forward , and softmax for first probabilistic model
- **Probabilistic module for logic reasoning over knowledge graph:**

  1) the knowledge graph can be very large; 2) the required logic reasoning is unknown and can be multi-step

  To solve this, authors proposes a **reasoning graph** (embedding architecture and inference rules in non-linear vector spaces to be learned)

  Scope of entity y (Subgraph Gy)(determined by Maximum hops:Starting from a topic entity y, we perform traverse for all entities within T hops according to the knowledge graph

Sub Graph G_y for Max_hop=2

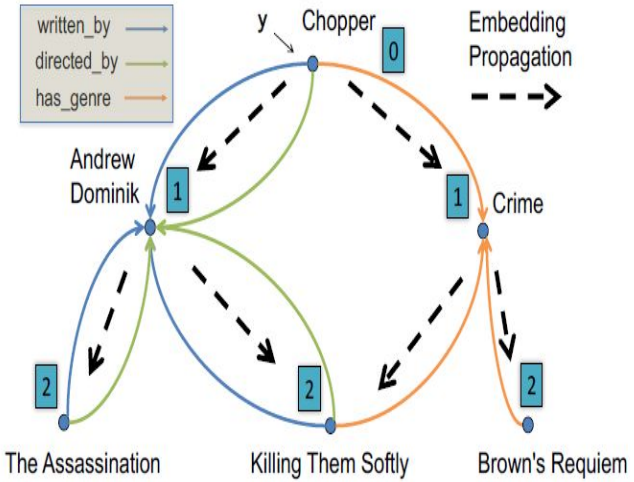
Given a potential answer in a

sub graph G_y, G_(y-->a), denotes

minimum subgraph that contains all the

Path from y->a, and denoted as

d-dimensional vector learned.



$$g(\mathcal{G}_{y\to a}) = \frac{1}{\#\text{Parent}(a)} \sum_{a_j \in \text{Parent}(a),(a_j,r,a) \text{ or } (a,r,a_j)\in\mathcal{G}_y} \sigma(V \times [g(\mathcal{G}_{y\to a_j}), \vec{e}_r]),$$

$$P_{\theta_2}(a|y,q) = \text{softmax}\left(f_{\text{qt}}(q)^\top g(\mathcal{G}_{y\to a})\right) = \frac{\exp(f_{\text{qt}}(q)^\top g(\mathcal{G}_{y\to a}))}{\sum_{a'\in V(\mathcal{G}_y)} \exp(f_{\text{qt}}(q)^\top g(G_{y\to a'}))}.$$

we use variational inference and optimize the negative Helmholtz variational free energy:

$$\max_{\psi,\theta_1,\theta_2} \mathcal{L}(\psi,\theta_1,\theta_2) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{Q_\psi(y|q_i,a_i)}[\log P_{\theta_1}(y|q_i) + \log P_{\theta_2}(a_i|y,q_i) - \log Q_\psi(y|q_i,a_i)], \tag{7}$$

$$Q_\psi(y|q,a) \propto \exp\left(\tilde{W}_y^\top \tilde{f}_{\text{ent}}(q) + \tilde{f}_{\text{qt}}(q)^\top \tilde{g}(G_{a\to y})\right),$$

$$a^* = \underset{a\in\mathcal{G}_y, y\in\{y_1,y_2,\ldots,y_k\}}{\text{argmax}} \log P_{\theta_2}(a|y,q).$$

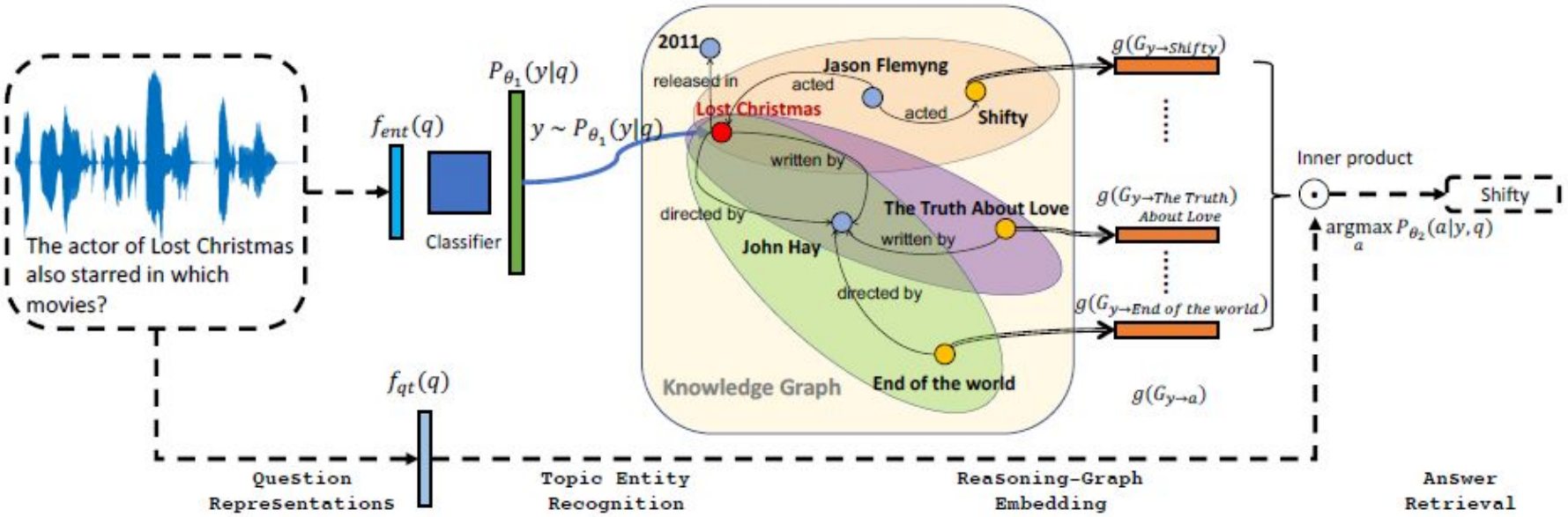


Table 1: Test results (% hits@1) on Vanilla and Vanilla-EU datasets. EU stands for entity unlabeled.

	Vanilla 1-hop	Vanilla 2-hop	Vanilla 3-hop	Vanilla-EU 1-hop	Vanilla-EU 2-hop	Vanilla-EU 3-hop
VRN	**97.5**	**89.9**	**62.5**	**82.0**	**75.6**	**38.3**
Bordes et al. [22]'s QA system	95.7	81.8	28.4	39.5	38.3	26.9
KV-MemNN	95.8	25.1	10.1	35.8	10.3	10.5
Supervised embedding	54.4	29.1	28.9	18.1	23.2	25.3

11-Nov-2020

Activities Done

- Knowledge Graph Embedding Based Question Answering (Read paper in detail)

Pending Action Items

1. Dynamic Graph Convolutional Networks for Entity Linking

ToDo

- Read Research Papers in detail

- Review of ***Knowledge Graph Embedding Based Question Answering***:
- Two issues in KG-QA is addressed in this paper:

  1)a predicate could be expressed in different ways in natural language questions (Unbounded user query).

  Eg. person.nationality can be expressed as “what is ... ’s nationality”, “which country is ... from”, “where is ... from”

  2)**Ambiguity** in the entity names or **partia**l names (eg. Barack Obama, Obama, Victoria as name, Victoria as place)

- To solve the problem:

The knowledge graph embedding to perform question answering as global relations are preserved in KG embeddings

- Task: Given a Question Q as input , outputs the corresponding **head entity** and **predicate**.
- Scope:
  - The paper targets Simple Questions (One fact: (head, predicate, tail)) . Eg. Complicated Question:(A complex question contains more than one fact) :“who is the alumni of Princeton University and Harvard University?”
- Algorithm:
  1. Using existing KG embedding algorithms (TransE /TransR) learn P (embedding representations of all predicates in the Graph) and E (embedding representations of all entities in the Graph)
  2. Obtain the predicate and head embeddings predicate and head entity learning models for a given Q.

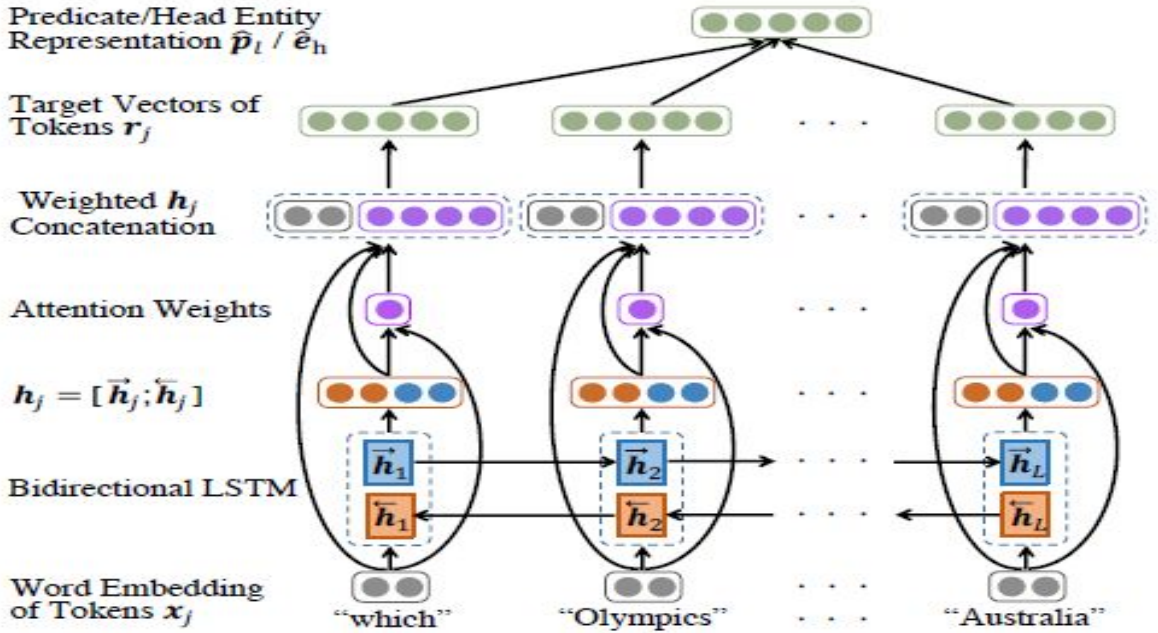


Figure 2: The architecture of the proposed predicate and head entity learning models.

3. Detect HEAD Entities from Head Entity Detection (HED) model to reduce search space/ Expense. Thus, entities that are the same as or contain HEDentity would be included as the **candidate head entities**

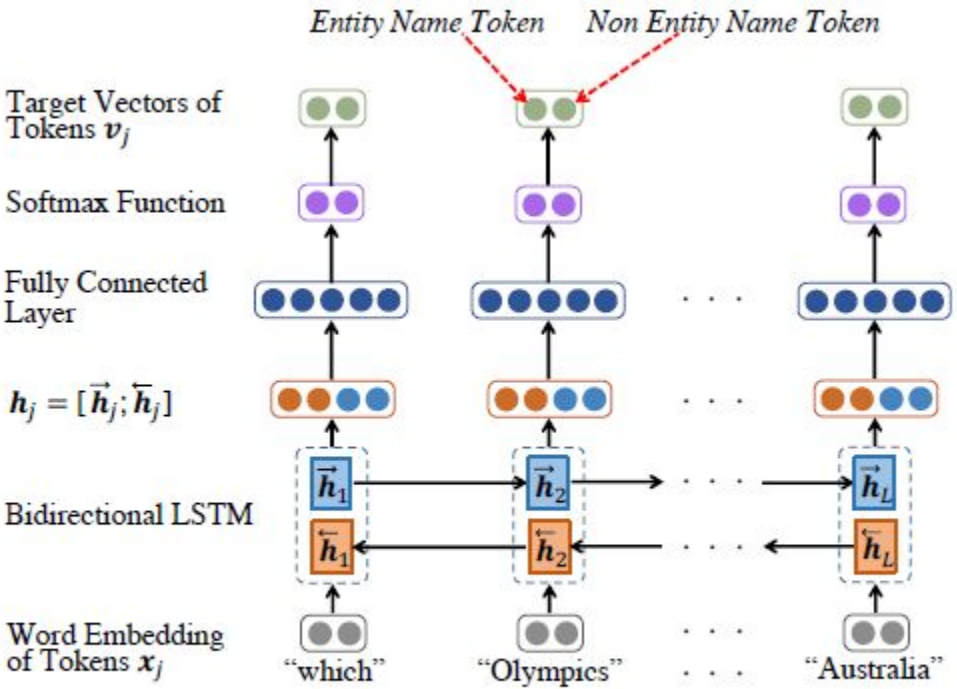


Figure 3: Structure of Head Entity Detection (HED) model.

Joint Search on Embedding Spaces:

For all the candidates in candidate head entities C find out the fact (h*, ℓ*, t*) that minimizes the distance metric between predicted (h', l') and fact (h,l) in Graph G, measured by a metric .

$$\underset{(h,\ell,t)\in C}{\text{minimize}} \; \|\mathbf{p}_\ell - \hat{\mathbf{p}}_\ell\|_2 + \beta_1\|\mathbf{e}_h - \hat{\mathbf{e}}_h\|_2 + \beta_2\|f(\mathbf{e}_h, \mathbf{p}_\ell) - \hat{\mathbf{e}}_t\|_2 - \beta_3 sim[n(h), \text{HED}_{\text{entity}}] - \beta_4 sim[n(\ell), \text{HED}_{\text{non}}], \quad (9)$$

Key Notes:

1) A novel knowledge graph embedding based question answering problem.
2) Targets Simple Questions.
3) Jointly recover the question's head entity, predicate, and tail entity representations in the KG embedding spaces.
4) Reduction of Search space and Cost of searching using HED model.

10-Nov-2020

Activities Done

- Identifying Relevant research Papers
- Publicly available dataset widely used in research

Pending Action Items

1. Knowledge Graph Embedding Based Question Answering (Read paper in detail)
2. Dynamic Graph Convolutional Networks for Entity Linking

ToDo

- Read Research Papers in detail

- Relevant research papers:
  - **Paper on Question Answering**: *Knowledge Graph Embedding Based Question Answering*(WSDM '19: Proceedings of the Twelfth ACM International Conference on Web Search and Data): (Problems in KG-QA: 1)a predicate could have different natural language expressions (semantic information). 2)ambiguity of entity names and partial names) . The paper proposed an approach to solve above problems: 1)By utilizing embedding information in the graph. jointly recover the question's head entity, predicate, and tail entity representations in the KG embedding spaces. It Attention-based bidirectional LSTM models are employed to perform the predicate and head entity representation learning. 2)It focuses on answering simple questions.)
  - *Variational Reasoning for Question Answering with Knowledge Graph*
  - **Paper on Entity Linking:** *Dynamic Graph Convolutional Networks for Entity Linking***(**WWW2020**)**(The graph structure in this model is dynamically computed and modified during training using Graph Convolutional Networks (GCN).)

- Relevant research papers:
  - **Paper on Graph attention**:
    - *Commonsense Knowledge Aware Conversation Generation with Graph Attention*(Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18) : It's a generation task, not search, could be relevant
    - *KGAT: Knowledge Graph Attention Network for Recommendation*(KDD '19): For recommendation
    - *Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs*(ACL): The paper proposes a novel attention-based feature embedding that captures both entity and relation features in any given entity's neighborhood.
  - **Paper on KG Enrichment::** *Entity Alignment between Knowledge Graphs Using Attribute Embeddings*(Third AAAI Conference on Artificial Intelligence (AAAI-19)): Learn embeddings that can capture the similarity between entities in different knowledge graphs. The proposed model helps align entities from different knowledge graphs, and hence enables the integration of multiple knowledge graphs
  -

- Relevant research papers:
  - **Other relevant survey papers:**:
    - *Knowledge Graph Embedding: A Survey of Approaches and Applications*
    - *Knowledge Graphs*: It discusses various graph-based data models and query languages that are used for knowledge graphs. creation, enrichment, quality assessment, refinement, and publication of knowledge graphs
    - *Knowledge Graph Refinement*:A Survey of Approaches and Evaluation Methods : discusses knowledge graph refinement approaches, with a dual look at both the methods being proposed as well as the evaluation methodologies used.
  - **Datasets:** *Publicly available large-scale KG dataset used in Research: DBpedia, Freebase, OpenCyc, Wikidata, YAGO, Webclient, Conceptnet*