

CS – 6320: Natural Language Processing

Report : Template based Information Extraction System using NLP Techniques

Team Name : PIADA

Prit Thakkar (pvt170000)

Ronit Patel (rrp170002)

A. Problem Description:

Information Extraction (IE) is an important part in the field of Natural Language Processing (NLP) and linguistics. It's widely used for tasks such as Question Answering Systems, Machine Translation, Entity Extraction, Event Extraction, Named Entity Linking, Coreference Resolution, Relation Extraction, etc. The scope of this project is to perform Information extraction using Template (slot) filling on the given unstructured textual data (such as Wikipedia Articles, Articles from Newspaper etc.). The goal was to achieve the task by applying various NLP techniques and creating a pipeline where we input unstructured text and the output we get are extracted templates containing information from the given data.

We were provided with 30 Wikipedia Articles:

- 10 articles related to Organizations
- 10 articles related to Persons
- 10 articles related to Locations

And we were expected to extract information from the above mentioned articles and fill the given 3 templates:

- **Template#1: BUY**(*Buyer, Item, Price, Quantity, Source*)

For this template we need to extract buying event.

- "Buyer", who is an agent (can be a Person or Organization) for the buying activity.
- "Item" that is being bought in the buying activity. It can be a Product, Organization etc.

- “Price”, monetary amount for which the item is being bought.
- “Quantity”, numerical quantity of the item that is being bought.
- “Source”, source from which an item is bought.

Example:

Sentence(s): In 2017, Amazon acquired Whole Foods Market for US\$13.4 billion, which vastly increased Amazon's presence as a brickand-mortar retailer.

Extracted Template: BUY(“Amazon”, “Whole Foods Market”, “US\$13.7 billion”, “”, “”)

- **Template#2: WORK**(*Person, Organization, Position, Location*)

For this template we are expected to extract job title relation between person and organizations.

- “Person”, person entity involved in the event.
- “Organization”, organization for which the person works for.
- “Position”, job title person holds in the organization involved in this template.
- “Location”, location where Person performs the job.

Example:

Sentence(s): Steven Paul Jobs was the chairman, chief executive officer (CEO), and co-founder of Apple Inc.

Extracted Template: WORK(“Steven Paul Jobs”, “Apple Inc.”, “chairman ; chief executive officer (CEO); co-founder”, “”)

- **Template#3: PART**(*Location_1, Location_2*)

For this template we are expected to extract part of relation between two locations.

- “Location_1”, sub-part of Location_2.
- “Location_2”, location where Location_1 is sub-part.

Example:

Sentence(s): Dallas is a technical hub in Texas.

Extracted Template: PART("Dallas","Texas")

B. Proposed Solution:

The task of Information Extraction (IE) based on template filling can be performed using multiple approaches. They are listed below:

- Supervised
- Semi-supervised
- Rule-based Approach

For this project we will be basically focusing on the Rule-based Approach, where we will be defining a set of syntactical and grammatical rules of a natural language and then use them to fill our templates for the problem. After reviewing multiple research paper on Rule- based Information Template extraction, we found multiple techniques to achieve this task. Some of them are listed below.

- Using Wordnet features of tokens from the text
- Semantic-Role Labelling(SRL)
- Dependency Parsing
- Constituency Parsing
- Named-Entity Recognition(NER)
- Co-reference Resolution
- Phrase Matching

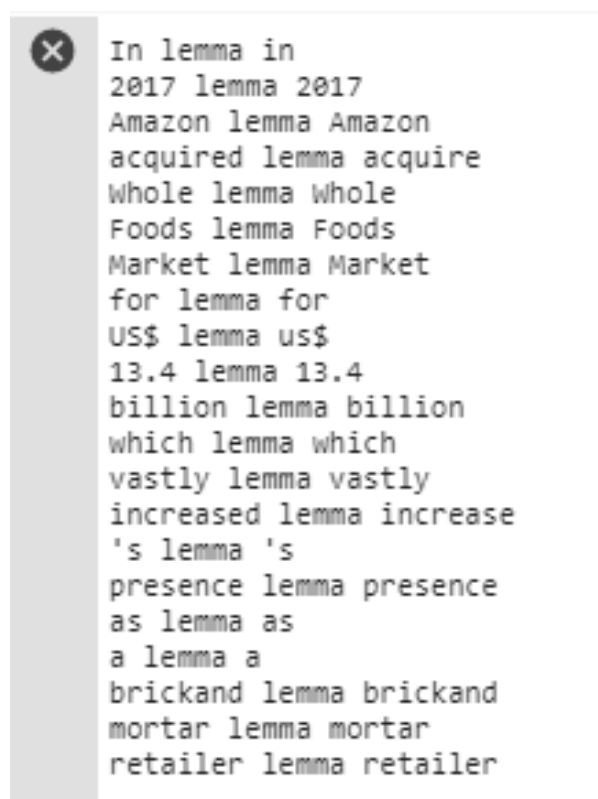
Initially, we apply Named-Entity Recognition and Co-reference Resolution to the given unstructured data in order to resolve pronoun references so that we can extract templates from a single standalone sentence. After that we apply Sentencizer to find sentence boundary to extract a single standalone sentence. Now for extracting information from the extracted sentence to fill Template#1, we use Lemmatization, NER, and SRL. For extracting information to fill Template#2, we use Lemmatization, NER, SRL, Dependency Parsing and Phrase Matching. And for Template#3, we

use Lemmatization and then extract Wordnet Features to find the part-of relation.

For Template#1(BUY):

Example Sentence: In 2017, Amazon acquired Whole Foods Market for US\$13.4 billion, which vastly increased Amazon's presence as a brickand-mortar retailer.

1. **Lemmatization:** We find lemmas of all the tokens in the sentence.



```
In lemma in
2017 lemma 2017
Amazon lemma Amazon
acquired lemma acquire
Whole lemma Whole
Foods lemma Foods
Market lemma Market
for lemma for
US$ lemma us$
13.4 lemma 13.4
billion lemma billion
which lemma which
vastly lemma vastly
increased lemma increase
's lemma 's
presence lemma presence
as lemma as
a lemma a
brickand lemma brickand
mortar lemma mortar
retailer lemma retailer
```

2. **NER:** We will apply NER on the sentence to find out entities and categorize them according to their entity labels.



In 2017 DATE , Amazon ORG acquired Whole Foods Market ORG for US\$13.4 billion MONEY , which vastly increased Amazon ORG 's presence as a brickand-mortar retailer.

3. **SRL:** Now we perform SRL to extract semantic role information about the verb present in the sentence. We perform this step only if

we find that the verb lemma into consideration with a sense similar to “buy”, for this example it is “acquire”.

acquired: [ARGM-DIS: In 2017] , [ARG0: Amazon] [V: acquired] [ARG1: Whole Foods Market] [ARG3: for US\$ 13.4 billion] , [ARGM-ADV: which vastly increased Amazon 's presence as a brickand - mortar retailer] .

Roles:

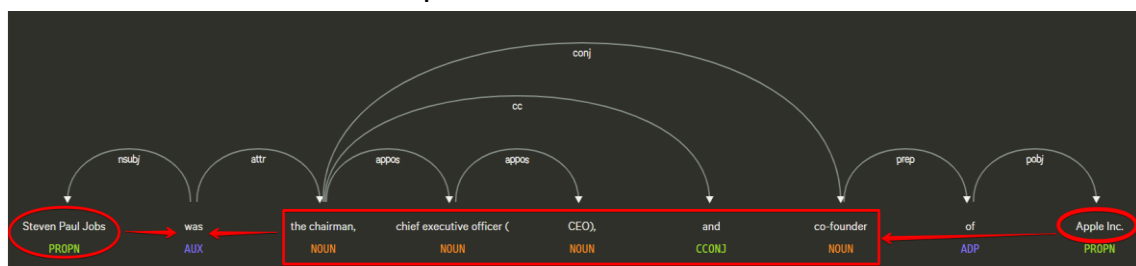
Arg0-PAG: *buyer* (vnrole: 13.5.1-agent)
Arg1-PPT: *thing bought* (vnrole: 13.5.1-theme)
Arg2-DIR: *seller* (vnrole: 13.5.1-source)
Arg3-VSP: *price paid* (vnrole: 13.5.1-asset)
Arg4-GOL: *benefactive* (vnrole: 13.5.1-beneficiary)

Extracted Template: BUY(“Amazon”, “Whole Foods Market”, “US\$13.7 billion”, “”, “”)

For Template#2(WORK):

Example Sentence: Steven Paul Jobs was the chairman, chief executive officer (CEO), and co-founder of Apple Inc.

1. **Lemmatization:** We find lemmas of all the tokens in the sentence similarly as we did for Template#1.
2. **NER:** We would similarly find NER as we did in Template#1.
3. **Dependency Parsing:** We would use the dependency parsing of the sentence to fill this template.



As it is clearly visible that the entity with label “ORG” i.e Apple Inc.(labelled as a part of NER) is related to all the “Position” tokens which is related to the Auxiliary Verb whose child with dependency “nsubj” is an entity labelled as “PERSON”.

Extracted Template: WORK("Steven Paul Jobs", "Apple Inc.", "chairman ; chief executive officer (CEO); co-founder", "")

But there are cases which are not covered with the dependency parsing technique. So, we use the ensemble approach using SRL and Phrase Matching.

Example Sentence: John worked for Amazon as a software engineer since 5 years.

4. **SRL:** Here we would find Semantic Role information by applying SRL for the verbs having a sense of "work" or "become".

worked: [ARG0: John] [V: worked] [ARG2: for Amazon] [ARG1: as a software engineer] [ARGM-TMP: since 5 years].

Roles:

Arg0-PAG: worker
Arg1-PPT: job, project
Arg2-GOL: employer, benefactive
Arg3-COM: coworker
Arg4-MNR: instrumental
Arg5-PRD: secondary-theme

Extracted Template: WORK("John", "Amazon", "software engineer", "")

Example Sentence: Amazon's co-founder Jeff Bezos announced about their acquisition of Whole Food Market.

5. **Phrase Matching:** In this technique, we match phrases based on custom rules like If we have Entity pattern as "ORG" "TITLE" "PERSON", we extract the work template.

Extracted Template: WORK("Jeff Bezos", "Amazon", "co-founder", "")

For Template#3(PART):

Example Sentence: Dallas is a technical hub in Texas.

1. **Lemmatization:** We find lemmas for all the tokens in the sentence similarly as we did for Template#1 and Template#2.

2. **NER:** We would similarly find NER as we did in Template#1 and Template#2.
3. **Wordnet Features:** First of all, we will only trigger this template if we have at least two entities with label “LOC” or “GPE” or “NORP” or “FAC” (using NER). Now we have to find the relation as Loc_1 is a part of Loc_2. So, we apply various rules on its Wordnet features such as:
 - a. Check if Holonyms(Loc_1) \cap Synonyms(Loc_2) \neq NULL
 - b. Check if Synonyms(Loc_1) \cap Meronyms(Loc_2) \neq NULL
 - c. Check if Holonyms(Loc_1) \cap Meronyms(Loc_2) \neq NULL

```

Dallas holonyms ['Texas']
Dallas synonyms ['Dallas']
is holonyms []
is synonyms ['be', 'be', 'be', 'exist', 'be', 'be', 'equal', 'be', 'constitute',
a holonyms ['nanometer', 'abampere']
a synonyms ['angstrom', 'angstrom_unit', 'A', 'vitamin_A', 'antiophthalmic_factor']
technical holonyms []
technical synonyms ['technical', 'technical_foul', 'technical', 'technical', 'pro
hub holonyms ['car_wheel', 'electric_fan', 'propeller']
hub synonyms ['hub', 'hub']
of holonyms []
of synonyms []
Texas holonyms ['Gulf_States', 'Southwest', 'United_States']
Texas synonyms ['Texas', 'Lone-Star_State', 'TX']

```

Extracted Template: PART(“Dallas”, “Texas”)

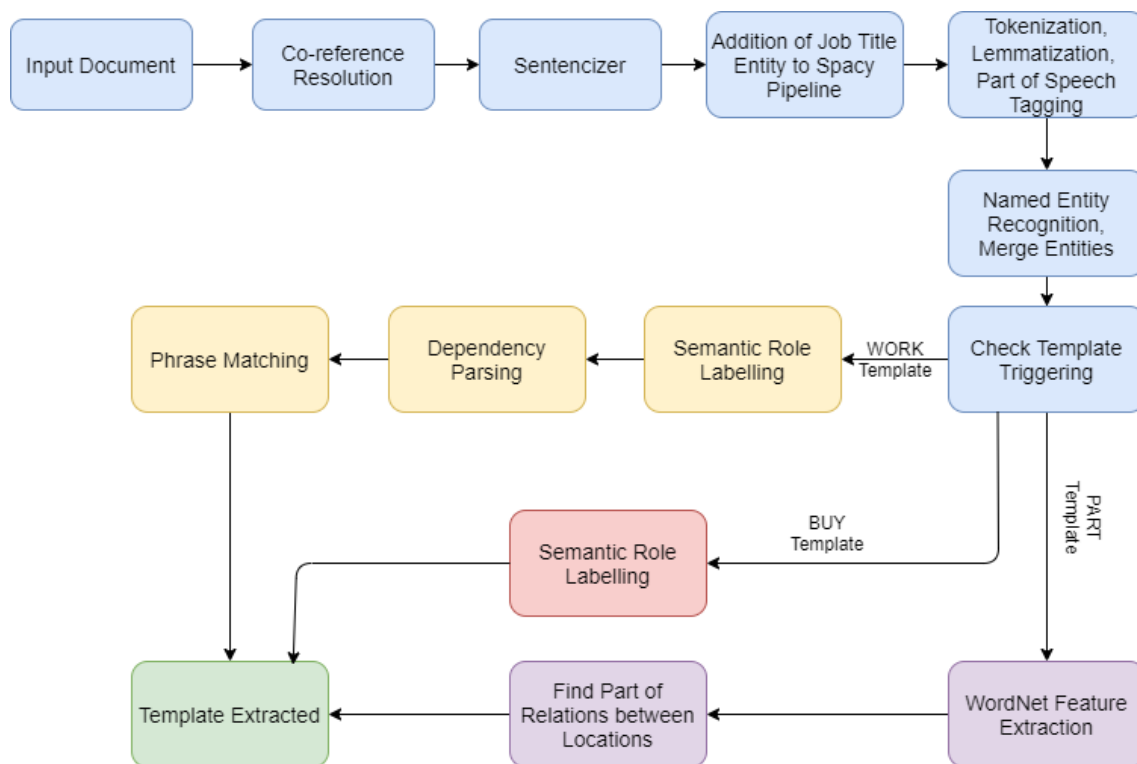
C. Full Implementation Details:

a. Programming Tools

To implement this system, we used python3 as Programming language along with various NLP libraries like spacy, AllenNlp and nltk.

b. Architecture

Below Image represents the Architecture of the NLP Pipeline developed for Template Extraction.

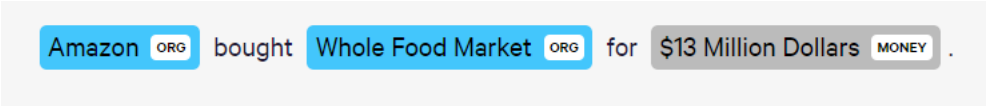


- Input Document:
 - Here, Wikipedia Articles (Unstructured Text) is provided as input to the pipeline.
- Co-reference Resolution:

- Here, the input document is parsed through AllenNlp Co-reference Resolution for resolving pronouns with entities.

0 Steven Paul Jobs (; February 24 , 1955 – October 5 , 2011) was an American business magnate and investor .
 0 He was the chairman , chief executive officer (CEO) , and co - founder of 3 Apple Inc. ; chairman and
 majority shareholder of 2 Pixar ; a member of 1 The Walt Disney Company 's board of directors following
 1 its acquisition of 2 Pixar ; and the founder , chairman , and CEO of NeXT . 0 Jobs is widely recognized as
 a pioneer of the microcomputer revolution of the 1970s and 1980s , along with 3 Apple co - founder Steve
 Wozniak .

- Sentencizer:
 - After co-reference resolution, it is passed into spacy Sentencizer to segment into sentences.
- Addition of Job Title Entities to Spacy Pipeline:
 - Spacy doesn't provide Job Title Entity in NER model. So, we provided available job titles from open source database and added to the spacy pipeline.
- Tokenization, Lemmatization, Part of Speech Tagging:
 - On each sentence Tokenization, Lemmatization and Part of Speech Tagging is performed to store as features.
 - Tokenization: Sentence is tokenized into tokens using spacy.
 Eg: "Amazon bought Audible."
 Tokens: ["Amazon", "bought", "Audible", "."]
 - Lemmatization: Each token's lemma is stored as feature.
 Eg: "bought" – "buy"
 - POS Tagging: Each token's associated POS Tag is extracted.
 Eg: Tokens: ["Amazon", "bought", "Audible", "."]
 POS Tags: ["PROPN", "VERB", "PROPN", "PUNCT"]

- Named Entity Recognition (NER), Merge Entities:
 - Named Entity Recognition is used to find features like Person, Organization, Location, Geographical Location Entities, Date, Monetary Entity, etc from sentence.
Eg: “Amazon bought Whole Food Market for \$13 Million Dollars.” The NER is shown below.
- 
- Merge Entities: Each entities are merged into single tokens for easier referencing to the entities.
 - Check Template Triggering:
 - Each sentences is checked on triggering of BUY, WORK and PART Template based on various heuristics to achieve efficiency. Once triggered, it performs extraction job.
 - Semantic Role Labeling:
 - In natural language processing, semantic role labeling is the process that assigns labels to words or phrases in a sentence that indicate their semantic role in the sentence, such as that of an agent, goal, or result.
 - The major thematic roles that can be extracted are Agent, Experiencer, Theme, Patient, Instrument, Beneficiary, etc.
 - Dependency Parsing:
 - Dependency Parsing is very useful technique to understand the grammatical structure of the sentence and defines relations between head words and words which modifies other words.
 - Phrase Matching:
 - Sometimes, template follows certain structure which can be extracted by specifying custom rules for phrases. Spacy provides

Matcher tool where we can specify custom rules for matching the phrases.

- WordNet Feature Extraction:
 - We are extracting various wordnet features like Hypernyms, Hyponyms, Holonyms, Meronyms, Entailments, etc. to use as features in our template extraction.

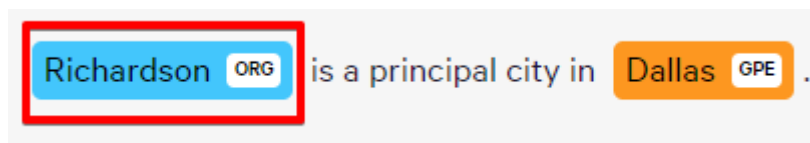
c. Result and Error Analysis:

There were various errors that we encountered while analyzing the result generated by our code. They are as follows:

- Sometimes Spacy's NER doesn't correctly label entities which results in error as we identify entities and use them for template filling in our strategies.

Example Sentence 1: "Richardson is a principal city in Dallas."

Spacy's NER annotation:



Instead of detecting Richardson as "LOC", it detected it as "ORG" and so we could not extract PART template from it.

Example Sentence 2: "Amazon sold Whole Food Market to Google."

Spacy's NER annotation:



It did not detect Whole Food Market as an "ORG" and so it would not extract the BUY template.

- Sometimes the Coreference resolution also fails to resolve the references correctly. And this cause incorrect extraction of information.

They received funding from ¹ a then - semi - retired Intel product marketing manager and engineer Mike Markkula .
⁰ Scott McNealy , one of the cofounders of Sun Microsystems , said that ¹ Jobs broke a " glass age ceiling " in Silicon Valley because ⁰ he 'd created a very successful company at a young age . ¹ Markkula brought ² Apple to the attention of Arthur Rock , which after looking at the crowded ² Apple booth at the Home Brew Computer Show , started with a \$ 60,000 investment and went on the ² Apple board .

Here, instead of resolving pronoun “he” with “Jobs”, it resolved the coreference with Scott McNealy.

- Lack of words and its relation in Wordnet Vocabulary. Due to that we cannot find Wordnet features which hinders our extraction of PART template.

Example Sentence: “Richardson is a principal city in Dallas and Collin counties in the U.S. state of Texas.”

Spacy’s NER annotation:

Richardson ORG is a principal city in Dallas GPE and Collin GPE counties in the U.S. GPE state of Texas GPE .”

Here Collin is detected as a “GPE”(GeoPolitical Entity), but when we try to find Wordnet relations for Collin, we get nothing.

Wordnet Features:

```
Richardson holonyms []
Richardson synonyms ['Richardson', 'Henry_Hobson_Richardson', 'Richardson', 'Ralph_Richardson', 'Sir_Ralph_David_Richardson']
is holonyms []
is synonyms ['be', 'be', 'be', 'exist', 'be', 'be', 'equal', 'be', 'constitute', 'represent', 'make_up', 'comprise', 'be', 'be', 'follow', 'embody', 'b
a holonyms ['nanometer', 'abampere']
a synonyms ['angstrom', 'angstrom_unit', 'A', 'vitamin_A', 'antiphthalmic_factor', 'axerophthol', 'A', 'deoxyadenosine_monophosphate', 'A', 'adenine',
principal holonyms ['loan']
principal synonyms ['principal', 'principal', 'school_principal', 'head_teacher', 'head', 'star', 'principal', 'lead', 'principal', 'corpus', 'principa
city holonyms []
city synonyms ['city', 'metropolis', 'urban_center', 'city', 'city', 'metropolis']
in holonyms ['foot', 'Corn_Belt', 'Midwest', 'United_States']
in synonyms ['inch', 'in', 'indium', 'In', 'atomic_number_49', 'Indiana', 'Hoosier_State', 'IN', 'in', 'in', 'in', 'inwards', 'inward']
Dallas holonyms ['Texas']
Dallas synonyms ['Dallas']
and holonyms []
and synonyms []
Collin holonyms []
Collin synonyms []
counties holonyms []
counties synonyms ['county', 'county']
the holonyms []
the synonyms []
U.S. holonyms ['North_America']
U.S. synonyms ['United_States_government', 'United_States', 'U.S._government', 'US_Government', 'U.S.', 'United_States', 'United_States_of_America', 'A
state holonyms []
state synonyms ['state', 'province', 'state', 'state', 'state', 'nation', 'country', 'land', 'commonwealth', 'res_publica', 'body_politic', 'state_of_m
of holonyms []
of synonyms []
Texas holonyms ['Gulf_States', 'Southwest', 'United_States']
Texas synonyms ['Texas', 'Lone_Star_State', 'TX']
```

And so we cannot apply our proposed solution to extract PART template.

d. Summary of Problems encountered and their solution

- We faced few problems related to NER for example, spacy's v2.2.0 model detect entity "Souq.com" as "ORG" but spacy's v2.2.5 model doesn't. Same ways entity "Ring" was detected as "ORG" in v2.2.5 but not in v2.2.0. Thus, we decide to ensemble both models to get maximum coverage of detecting entities.
- Also, spacy's NER doesn't provide labelling to Person's Job Title entities. So, we used open-source data for Job Title and then provided it as an input to spacy's EntityRuler which after applying default NER matches patterns provided to it and labels matched entities as "TITLE".
- There were many 3rd party NLP tools that are available for python. And the problem was some provide better results for certain tasks than the other. So, after detailed analysis we concluded to use AllenNlp for Semantic Role Labelling and Coreference Resolution; and Spacy for rest of the tasks.
- For the WORK Template extraction, spacy's NER tagged certain Organization as "PERSON" and so we used both "ORG" as well as "PERSON" label to trigger the WORK Template extraction. The given solution popped some False Positive which we resolve using dependency parsing.

e. Potential Improvement

- Instead of using default model for NER, SRL and Coreference provided by Spacy and AllenNlp, on top of it, we could train a model to increase our performance.
- We could further find instances where our proposed solution fails and provide a workaround for that.