Semantic relationship extraction in patents

Alex-Răzvan Ispas
Benjamin Maudet

G06K
RECOGNITION OF DATA;
PRESENTATION OF DATA; RECORD
CARRIERS; HANDLING RECORD
CARRIERS

Patents are tedious

These and other implementations may each optionally include one or more of the following features: that generating the first compact representation of the first object in the first image includes generating a first initial representation of the first object in the first image, mapping, using the first machine learning logic, the first initial representation of the first object to the first compact representation of the first object, and generating the second compact representation of the second object in the second image includes generating a second initial representation of the second object in the second image, and mapping, using the second machine learning logic, the second initial representation of the second object to the second compact representation of the second object; that the first initial representation of the first object is a first initial feature vector (IFV), the first compact representation of the first object is a first compact feature vector (CFV), the second initial representation of the second object is a second IFV, the second compact representation of the second object is a second CFV; that the first IFV and the second IFV each includes one or more texture features, one or more color features, one or more context features, and one or more viewpoint features; that identifying the subset of features of the first object and the second object as being more determinative than the other features of the first object and the second object includes computing a feedback difference between the similarity score and the predetermined target output, and identifying the subset of features from the first initial representation of the first object and the second initial representation of the second object as being more determinative than the other features from the first initial representation of the first object and the second initial representation of the second object based on the feedback difference; that generating the first compact representation of the first object includes reducing a first number of features comprising a first initial representation of the first object to obtain the first compact representation of the first object, and generating the second compact representation of the second object includes reducing a second number of features comprising a second initial representation of the second object to obtain the second compact representation of the second object; that the predetermined target output indicates whether the first object in the first image and the second object in the second image represent a same object; that adjusting one or more first parameters of the first machine learning logic and one or more second parameters of the second machine learning logic based on the identified subset of features; that the one or more first parameters of the first machine learning logic are identical to the one or more second parameters of the second machine learning logic; that determining that the one or more first parameters of the first machine learning logic and the one or more second parameters of the second machine learning logic are sufficiently adjusted, and responsive to determining that the one or more first parameters of the first machine learning logic and the one or more second parameters of the second machine learning logic are sufficiently adjusted, implementing the first machine learning logic in a first vehicle and implementing the second machine learning logic in a second vehicle; that receiving, from the first vehicle, a third compact representation of a third object in a third image, the third compact representation of the third object generated by the first machine learning logic implemented in the first vehicle, receiving, from the second vehicle, a fourth compact representation of a fourth object in a fourth image, the fourth compact representation of the fourth object generated by the second machine learning logic implemented in the second vehicle, computing a first similarity score between the third object in the third image and the fourth object in the fourth image using the third compact representation of the third object and the fourth compact representation of the fourth object, and determining whether the third object in the third image is a same object as the fourth object in the fourth image based on the first similarity score; that determining that the one or more first parameters of the first machine learning logic and the one or more second parameters of the second machine learning logic are sufficiently adjusted by computing a feedback difference between the similarity score and the predetermined target output, and determining that the feedback difference between the similarity score and the predetermined target output satisfies a predetermined difference threshold; that determining that the one or more first parameters of the first machine learning logic and the one or more second parameters of the second machine learning logic are sufficiently adjusted by determining a number of times the one or more first parameters of the first machine learning logic and the one or more second parameters of the second machine learning logic are adjusted, and determining that the number of times the one or more first parameters of the first machine learning logic and the one or more second parameters of the second machine learning logic are adjusted satisfies a predetermined number threshold; that computing the similarity score is performed by third machine learning logic, computing a feedback difference between the similarity score and the predetermined target output, and adjusting one or more third parameters of the third machine learning logic based on the feedback difference; that determining that the one or more third parameters of the third machine learning logic are sufficiently adjusted, and responsive to determining that the one or more third parameters of the third machine learning logic are sufficiently adjusted, implementing the third machine learning logic in a computing server; that the first machine learning logic is a first subnetwork of a neural network and the second machine learning logic is a second subnetwork of the neural network, the first subnetwork is identical to the second subnetwork

This is one sentence.

Seriously.

Extract useful knowledge

Extract meaningful terms

Find relationships

performs(*pre-processing circuit, image processing)*

Term detection

- Domain is kind of vague
- “Recognition of data” : applicable to many domains
- Have terms from computer science, biology, automotive, physics, textile, 3G geometry…
- Can’t rely on a domain specific list of terms
- Have to train a model to learn and recognize terms that “look technical” ?

Term definition

An entity in the patent can be considered a term if it meets one of the following conditions:

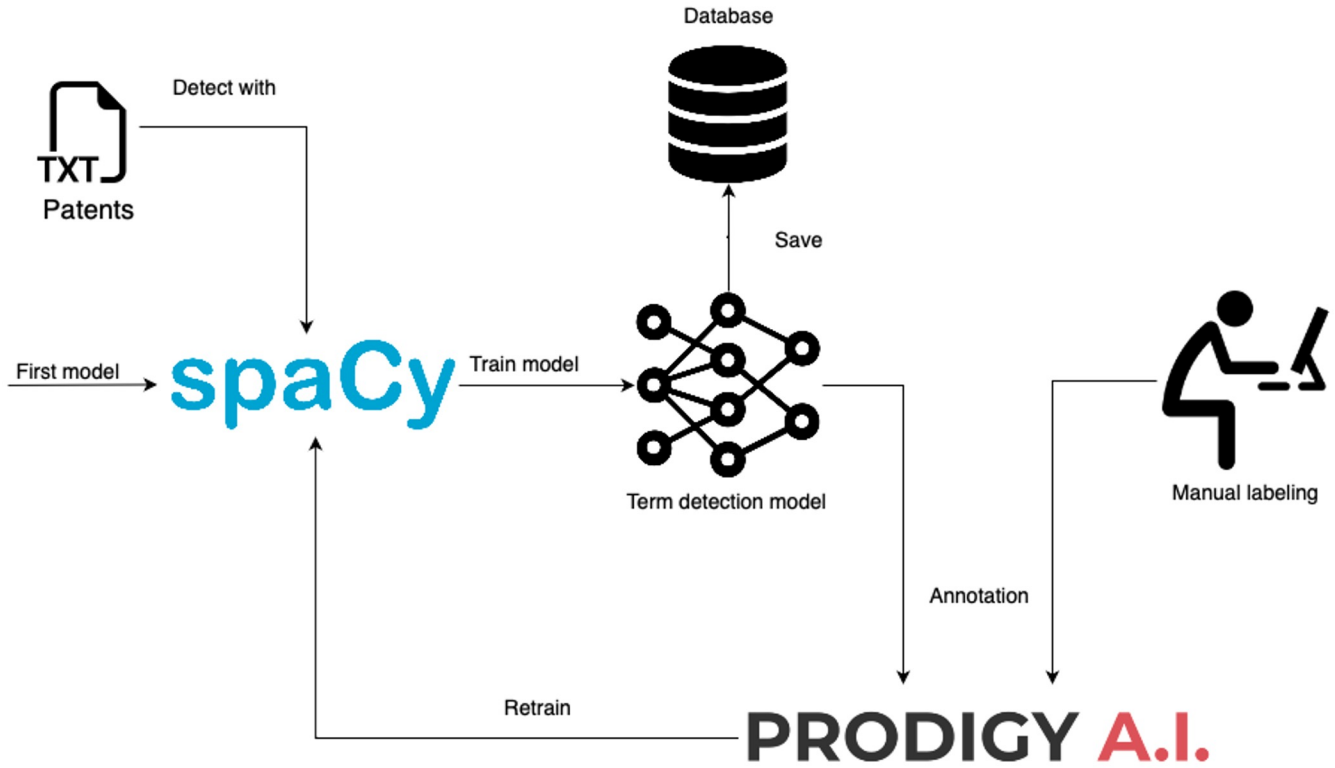
1. It contains at least one noun and it is succeeded by a reference numeral.
2. It contains at least one noun and it has a high frequency in a paragraph.
3. It is required in a relationship for a term that was identified in the other rules from above
   - Term 1 is term 2
   - Term 1 contains term 2

Term detection : annotations

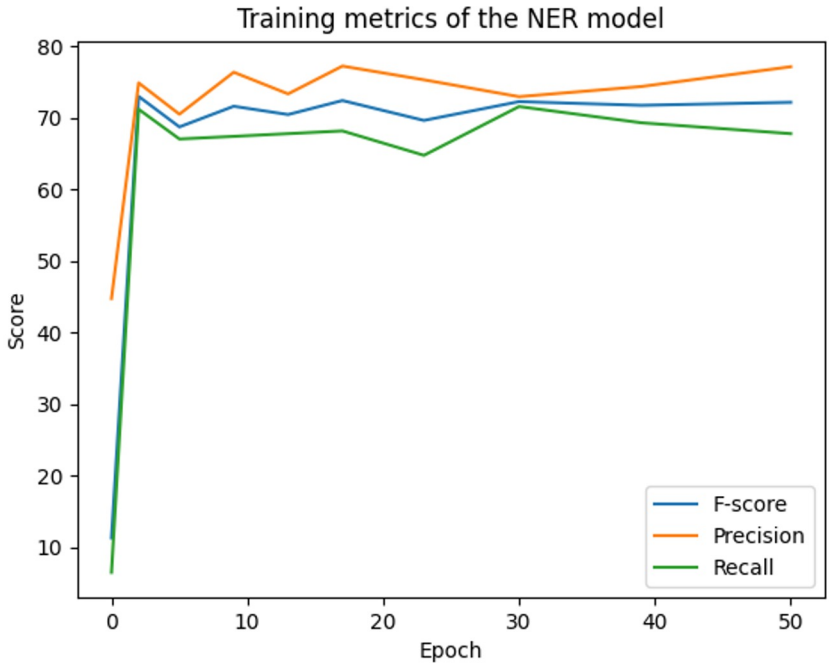
prodigy

- Active learning
- **ner.teach** recipe
  - Model suggests a term, press Yes, No or Skip
- **ner.correct** recipe
  - Model suggests terms, can say Yes, No, but can also correct the prediction manually
  - This results in gold text
- 431 annotations total

Term detection



Term detections : training the NER model



80% / 20% split of the annotated data

431 annotations

Term detection : model

Model-best prodigy and spacy

To determine whether an object is present in a laser point cloud image TT , the object TT detector software and/or module can associate arrangements of laser-indicated points TT with patterns matching objects TT , environmental features TT , and/or categories of objects or features TT . The object TT detector can be pre-loaded TT (or dynamically instructed) to associate arrangements TT according to one or more parameters corresponding to physical objects/features TT in the environment TT surrounding the vehicle TT 100. For example, the object TT detector can be pre-loaded TT with information indicating TT a typical height TT of a pedestrian, a length TT of a typical automobile TT , confidence TT thresholds for classifying suspected objects, etc.

Initial spacy model

To determine whether an object is present in a laser point cloud TT image, the object detector software and/or module can associate arrangements of laser-indicated points with patterns matching objects, environmental features, and/or categories of objects or features. The object detector can be pre-loaded (or dynamically instructed) to associate arrangements according to one or more parameters corresponding to physical objects/features in the environment surrounding the vehicle 100 TT . For example, the object detector can be pre-loaded with information indicating TT a typical height of a pedestrian, a length of a typical automobile, confidence thresholds for classifying suspected objects, etc.

Relationship extraction

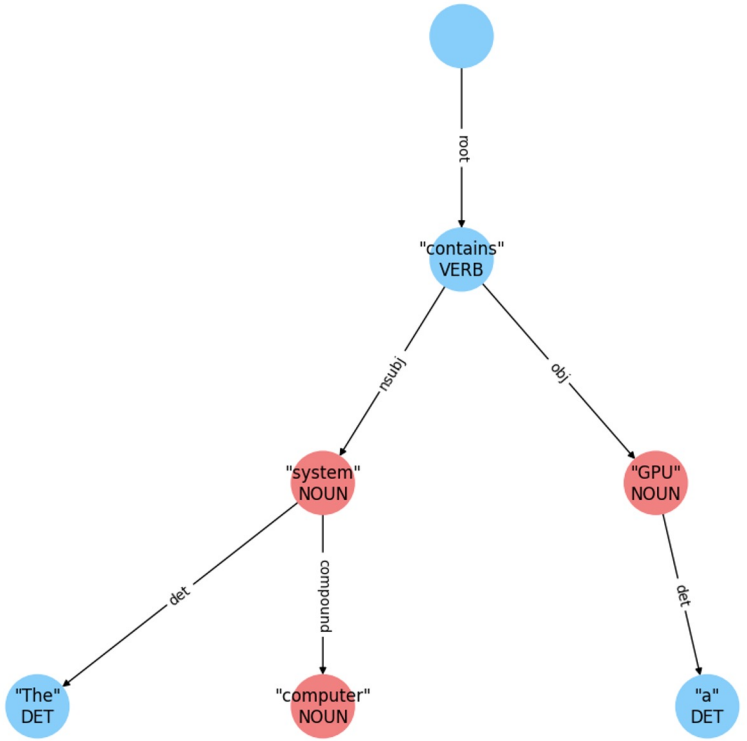
- We have terms. How do we connect them?
- A relationship should be a meaningful link between two terms
  - includes(computer system, chiller system) : meaningful
  - and(word1, word2) : *not* meaningful

Relationship extraction
First step : dependency parsing

nlp-uoregon/
trankit

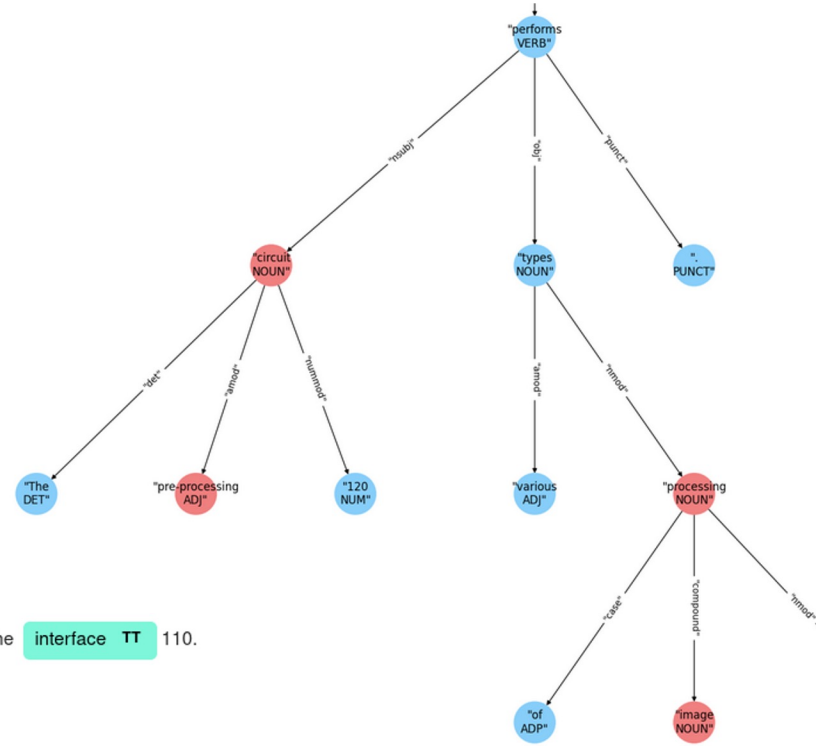
Trankit is a Light-Weight Transformer-based Python Toolkit for Multilingual Natural Language Processing

NetworkX
Network Analysis in Python



Relationship extraction : A basic algorithm

- For each term in graph
  - Find shortest path to each other term in the graph that don't include other terms
  - Sort each node in the path by their span start
  - Assemble relation name



The pre-processing circuit TT 120 performs various types of image processing TT on image data input TT from the interface TT 110.

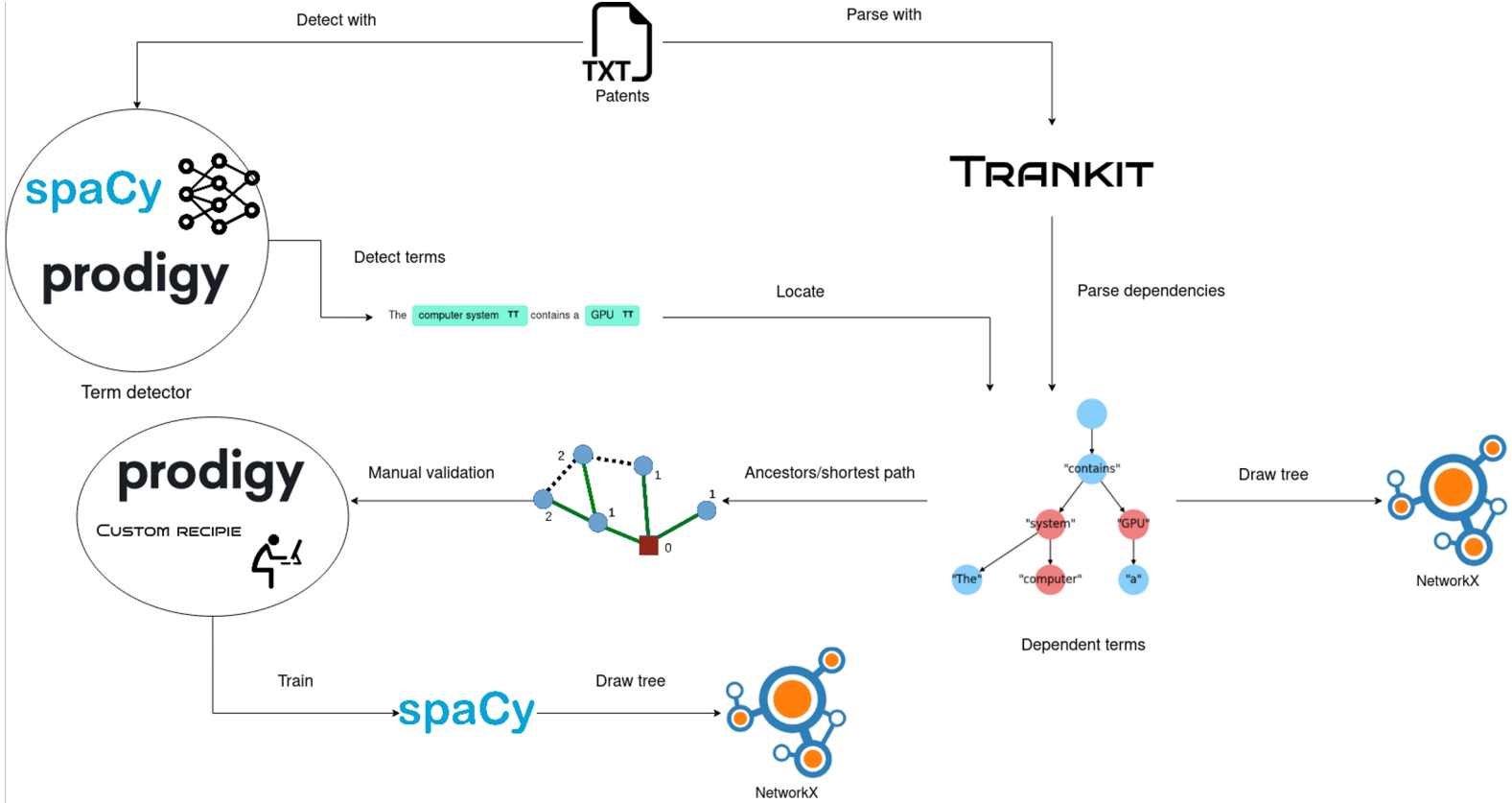
performs_types(pre-processing circuit, image processing)

Relationship extraction : A basic algorithm

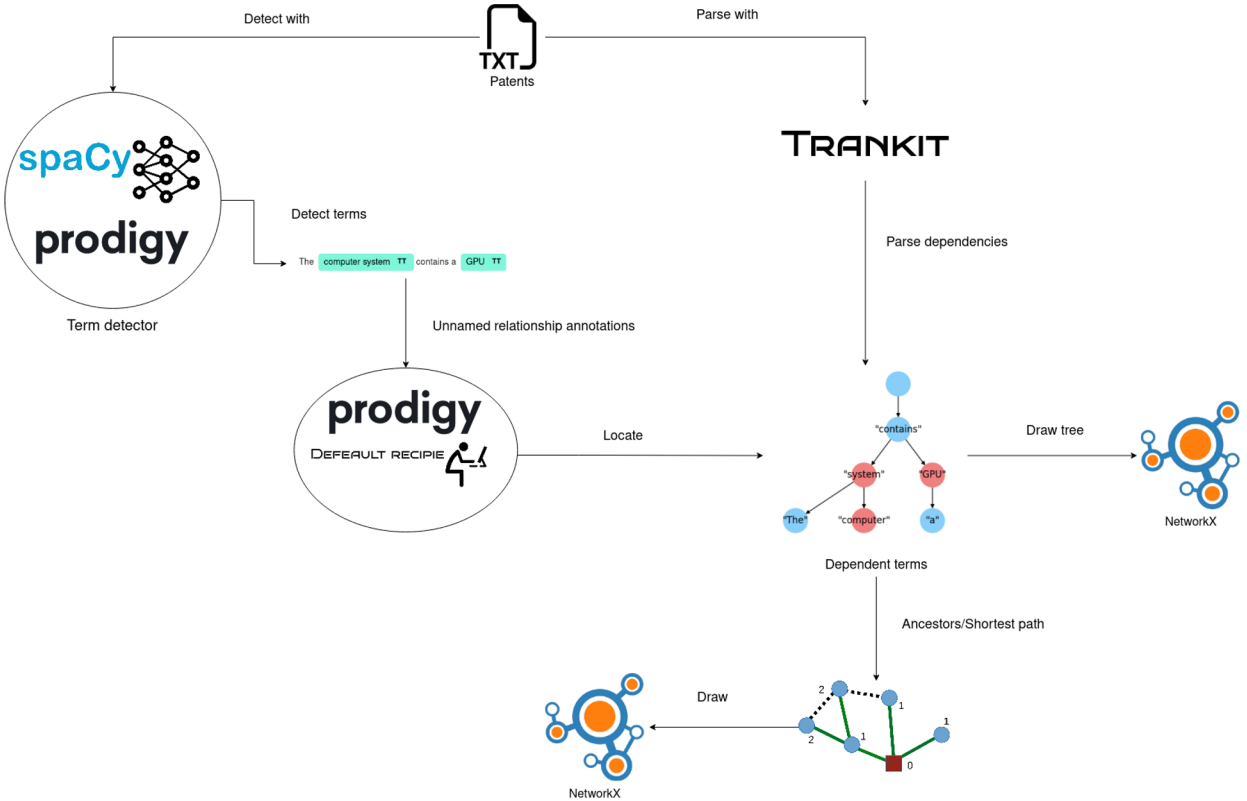
- Pros : non meaningful words are lower in the dependency graph, we don't use them for the relationship
- Problems
  - Doesn't work with coordinations (path is "blocked" in the graph by the first term)
  - Rules too simple : if there's a path between two terms in the dependency graph, they'll be matched together (poorly)
  - We need to
    - Identify if two terms **should** be linked
    - Make relationship names better

```
comprises(RFID apparatus, integrated circuit)
comprises(RFID apparatus, first RF module)
comprises(RFID apparatus, second RF module)
comprises_located(RFID apparatus, first position)
comprises_adapted_couple(RFID apparatus, coupling region)
comprises_adapted_shielded(RFID apparatus, second RF module)
comprises_adapted_shielded_located(RFID apparatus, first position)
comprises_adapted_shielded(RFID apparatus, second RF module)
comprises_adapted_couple(RFID apparatus, first booster antenna)
comprises_adapted_couple_shielded(RFID apparatus, first RF module)
comprises_adapted_couple_shielded_located(RFID apparatus, first position)
comprises_traces_comprising(RFID apparatus, second RF module)
comprises(second RF module, integrated circuit)
comprises(second RF module, first RF module)
comprises_located(second RF module, first position)
comprises_adapted_couple(second RF module, coupling region)
comprises_adapted_shielded_located(second RF module, first position)
comprises_adapted_couple(second RF module, first booster antenna)
comprises_adapted_couple_shielded(second RF module, first RF module)
comprises_adapted_couple_shielded_located(second RF module, first position)
comprises(integrated circuit, first RF module)
comprises(integrated circuit, second RF module)
comprises_located(integrated circuit, first position)
comprises_adapted_couple(integrated circuit, coupling region)
comprises_adapted(integrated circuit, second RF module)
comprises_adapted_located(integrated circuit, first position)
comprises_adapted(integrated circuit, second RF module)
comprises_adapted_couple(integrated circuit, first booster antenna)
comprises_adapted_couple(integrated circuit, first RF module)
comprises_adapted_couple_located(integrated circuit, first position)
comprises_traces_comprising(integrated circuit, second RF module)
comprising(second RF module, first RF module)
comprising_couple(second RF module, coupling region)
comprising_located(second RF module, first position)
```

Relationship extraction
Improvement Idea 1 : custom prodigy recipe



Relationship extraction
Improvement Idea 2: rel.manual annotations

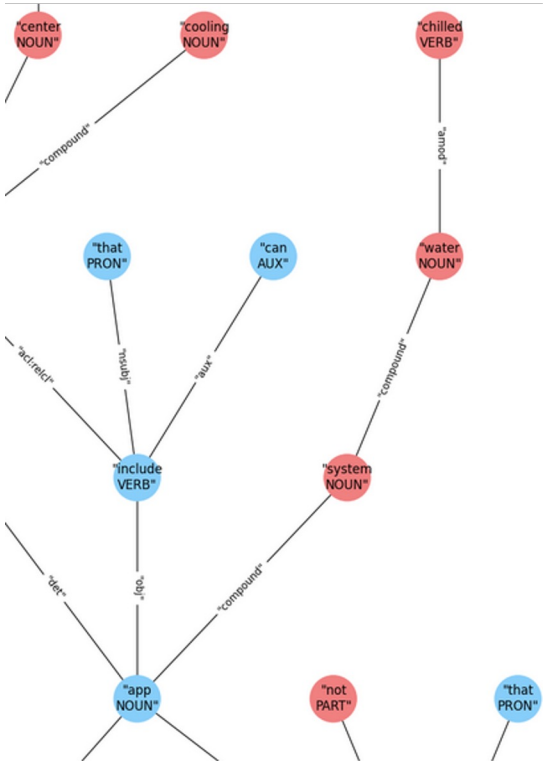


Relationship extraction
Improvement Idea 3 : term extension

No more manual work!!
We can be smarter about this

Extend matched terms with their compounds
-> correct some of the term matcher's mistakes

at can include a **chilled water system** TT app wit



Relationship extraction
Improvement Idea 4 : co-occurrence analysis

- We shouldn't match all terms together
- Compute statistics about terms that frequently occur next to each other
- Only link terms if they have a high co-occurrence statistic

```
[98]: for e in tqdm(doc.ents):
          for f in doc.ents:
              if e.text == f.text:
                  continue
              co_occ_mat[e.text][f.text] += abs(e.start_char-f.start_cha

100% 988/988 [02:05<00:00, 8.20it/s
```

```
]: co_occ_mat["perception controller"].sort_values()

]: perception controller            0
   fidelity                        60
   virtual model                   76
   detected objects               135
   predetermined importance       163
                                  ...
   system                      183862
   image processing system     191323
   preliminary action          198663
   moving body                 198967
   embodiment                  205510
```

In an additional aspect of the present disclosure, the at least one perception controller TT is further configured to increase the fidelity TT of the virtual model TT by assigning a priority level each of the detected objects TT based on a predetermined importance TT of each object.

Relationship extraction
Improvement Idea 5 : coreference extraction

- “This”, “it”, “they”... They might all reference terms and give precious information
- Pretrained coreference resolution model on english text

spaCy

CoreferenceResolver

CLASS,EXPERIMENTAL

STRING NAME: coref TRAINABLE:

The marsouin system TT is linked to a detection apparatus TT . It is also connected to a serial port TT . This system also depends on strawberries, which is not true for the cheetah system TT .

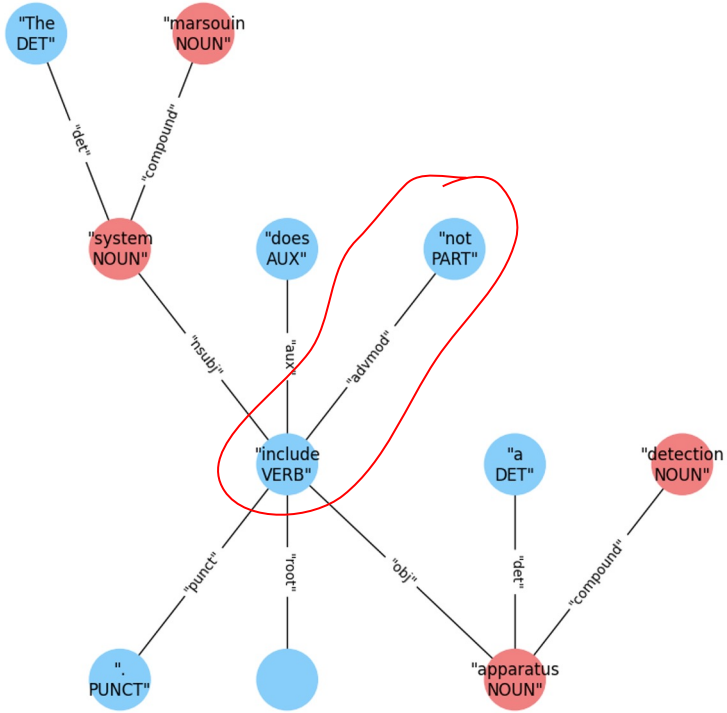
```
import coreferee
nlp = spacy.load("en_core_web_lg")
nlp.add_pipe('coreferee')
doc = nlp(txt)

doc._.coref_chains.print()
```

```
0: system(2), It(10), system(20)
```

Relationship extraction
Improvement Idea 6 : handling negations

- Trankit doesn't seem to have a "NEG" tag in its dependency relation parser
- Can hardcode a list of negative words and follow adverbial modifiers
- Can use spacy's dependency parser



Negation modifier

A negation modifier (`neg`) is an adverb that gives negative meaning to its head.

Other work

- Fixed relation datasets
  - TACRED (10.1 8653/v1/D17-1004)
  - Wiki80 (10.48550 /arXiv.1909.13078)
- Existing models
  - OpenNRE : Open-Source Neural Relation Extraction
  - BERT, CNN …
- Not really suitable for our patents


```
"place served by transport hub": "P931",
"mountain range": "P4552",
"religion": "P140",
"participating team": "P1923",
"contains administrative territorial entity": "P150",
"head of government": "P6",
"country of citizenship": "P27",
"original network": "P449",
```


Examples relationships from Wiki80

The TAC Relation Extraction Dataset

A large-scale relation extraction dataset with 106k+ examples over 42 TAC KBP relation types.

For now, we have the following available models:

- `wiki80_cnn_softmax`: trained on `wiki80` dataset with a CNN encoder.
- `wiki80_bert_softmax`: trained on `wiki80` dataset with a BERT encoder.
- `wiki80_bertentity_softmax`: trained on `wiki80` dataset with a BERT encoder (using entity representation concatenation).
- `tacred_bert_softmax`: trained on `TACRED` dataset with a BERT encoder.
- `tacred_bertentity_softmax`: trained on `TACRED` dataset with a BERT encoder (using entity representation concatenation).

Conclusion

- Baseline system, works on simple relations
- We have the tools and ideas to make it better with a bit more time
- Can use current rule based system to generate silver, make gold with prodigy, train ML model

References

Han, Xu, et al. "OpenNRE: An open and extensible toolkit for neural relation extraction." *arXiv preprint arXiv:1909.13078* (2019).

Bach, Nguyen, and Sameer Badaskar. "A review of relation extraction." *Literature review for Language and Statistics II* 2 (2007): 1-15.

Prodigy: https://prodi.gy/

Spacy: https://spacy.io/