# Named Entity Recognition for Medicines

Cinyoung Hur
2018.04.28


Such data!
So Big
Wow


B-26
B-27

# Why do I want to do NER for medication

-

| NAME |
| --- |
| 상황균사체엑스 1.1g |
| phellinus linteus mycelliun ext. 1.1g |
| 상황균사체엑스 550mg |
| phellinus linteus mycelliun ext. 0.55g |
| phellinus linteus mycelliun ext. 1.1g(36.667mg/mL) |
| 코노데옥시콜린산과 우르소데스콜린산의 3수 마그네슘2 250mg |
| magnesium trihydrate salt of chenodesoxycholic acid and ursodesoxycholic acid 0.25g |
| abciximab 10mg |
| abciximab 5mg |
| abciximab 5mg(2mg/mL) |
| abciximab 10mg(2mg/mL) |
| acamprosate 333mg |
| acarbose 100mg |
| acarbose 0.1g |
| acarbose 50mg |
| acebrophylline 100mg |

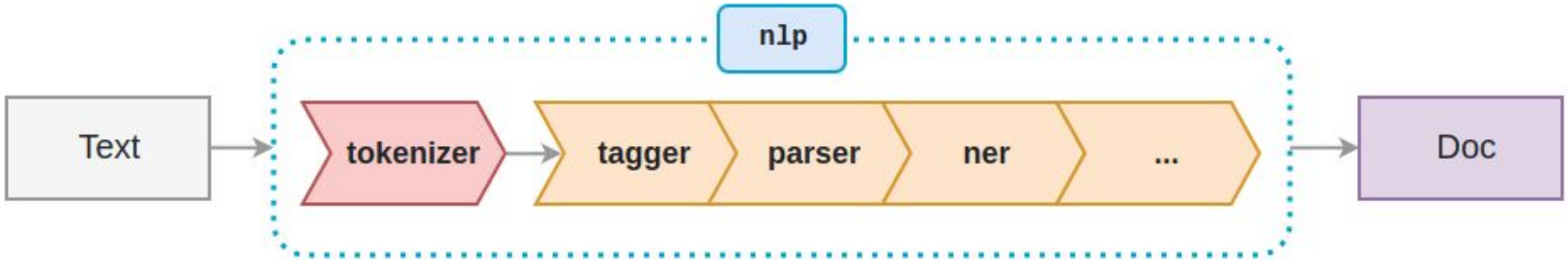
# Why do I want to do NER for medication

- Because I want to…

- Parse medication dosage and unit from semi-structured documents
- Enhance metadata of medicines

# spaCy

- Library for natural language processing
- NLP pipelines to generate models in corpora*
- open source and has several extra libraries and tools
  - displaCy, prodigy, etc.
- tools to build word and document vectors from text




*https://spacy.io/usage/processing-pipelines

# Named Entity Recognition*

Named entity recognition is the task of tagging proper nouns and numeric entities

Foundational tasks in NLP because most of work in NLP is annotations that are internal and contextual information

* https://spacy.io/usage/linguistic-features#101

# NER using spaCy

To start using spaCy for named entity recognition

- Install and download all the pre-trained word vectors

To train vectors yourself and load them

- Train model with entity position in train data

Named entities are available as the ents property of a Doc

# Example: NER using spaCy*

```
doc = nlp(u'Apple is looking at buying U.K. startup for $1 billion')

for ent in doc.ents:
    print(ent.text, ent.start_char, ent.end_char, ent.label_)
```


*https://spacy.io/usage/linguistic-features#section-named-entities

# Why do I want to do NER for medication

-

| NAME |
| --- |
| 상황균사체엑스 1.1g |
| phellinus linteus mycelliun ext. 1.1g |
| 상황균사체엑스 550mg |
| phellinus linteus mycelliun ext. 0.55g |
| phellinus linteus mycelliun ext. 1.1g(36.667mg/mL) |
| 코노데옥시콜린산과 우르소데스콜린산의 3수 마그네슘2 250mg |
| magnesium trihydrate salt of chenodesoxycholic acid and ursodesoxycholic acid 0.25g |
| abciximab 10mg |
| abciximab 5mg |
| abciximab 5mg(2mg/mL) |
| abciximab 10mg(2mg/mL) |
| acamprosate 333mg |
| acarbose 100mg |
| acarbose 0.1g |
| acarbose 50mg |
| acebrophylline 100mg |

# Toy example of training an additional entity types

# Example of training an additional entity type

List of new entity types

DRUG
DOSAGE
NANOGRAM
MILLIGRAM
GRAM
MILLILITER
PERCENT
PER

# TRAIN_DATA

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| p | h | e | l | l | i | u | s | | l | i | n | t | e | u | s | | m | y | c | e | l | l | i | u | n | | e | x | t | . | | | | 1 | . | 1 | g |

```
('phellinus linteus mycelliun ext.   1.1g',  {

'entities': [(0, 31, DRUG),  (33, 36, DOSAGE), (36, 37, GRAM)]

})
```

# Load spaCy model and add NER pipeline

```
nlp = spacy.load('en')  # load existing spaCy model
print("Loaded model '%s'" % model)

ner = nlp.create_pipe('ner')
nlp.add_pipe(ner)
```

# Add new entity label to entity recognizer

```
LABELS = [
    DRUG,
    DOSAGE,
    NANOGRAM,
    MILLIGRAM,
    GRAM,
    MILLILITER, ...
]
for LABEL in LABELS:
      ner.add_label(LABEL)
```

# Train NER

```
optimizer = nlp.entity.create_optimizer()

with nlp.disable_pipes(*other_pipes):  # only train NER
    for itn in range(n_iter):
        random.shuffle(TRAIN_DATA)
        losses = {}
        for text, annotations in TRAIN_DATA:
            nlp.update([text], [annotations],
                       sgd=optimizer, drop=0.25, losses=losses)
    print(losses)
```

# Test

Tested on 1000 medicines

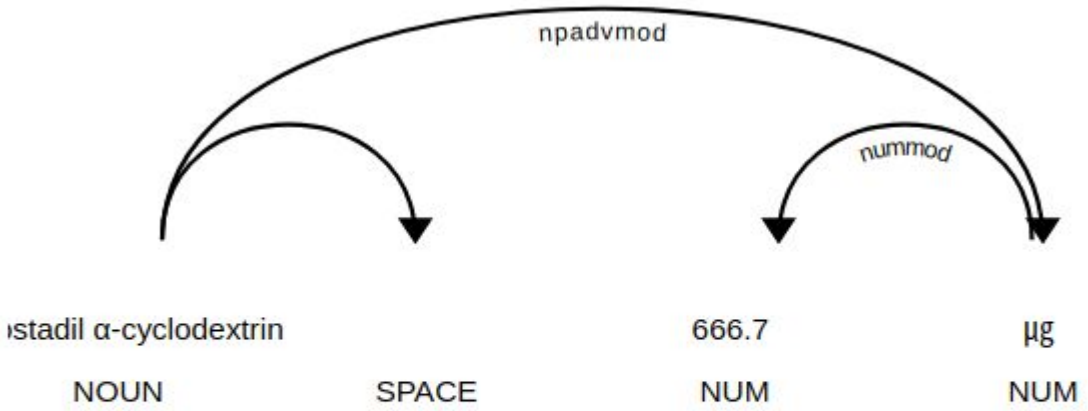
Performance

- good in relatively simple medicine names

Limitation of current state

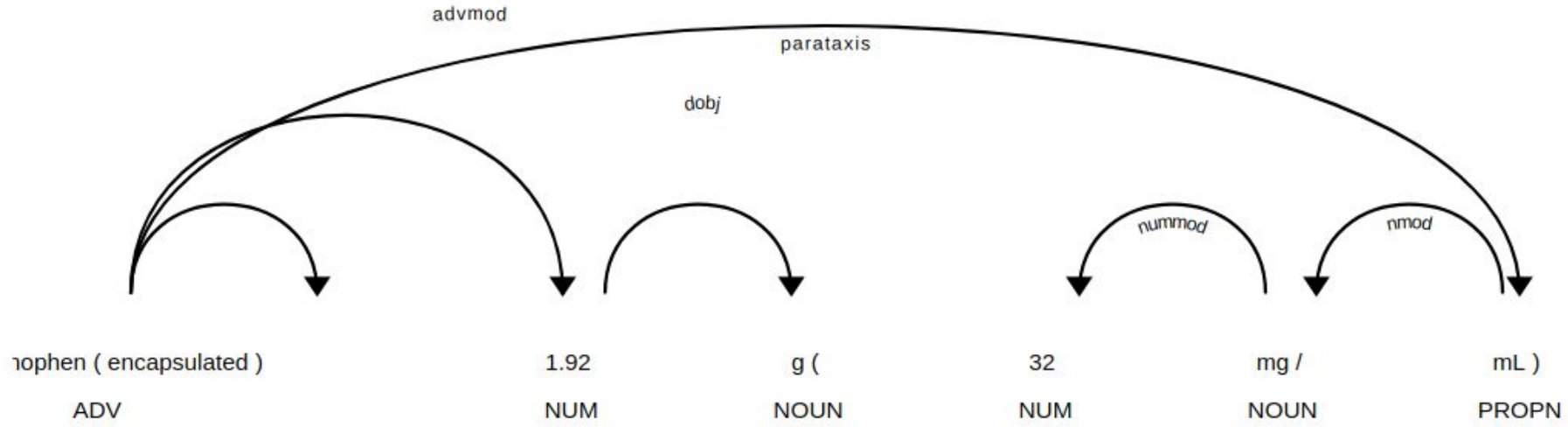
- inconsistent NER results

# Test result



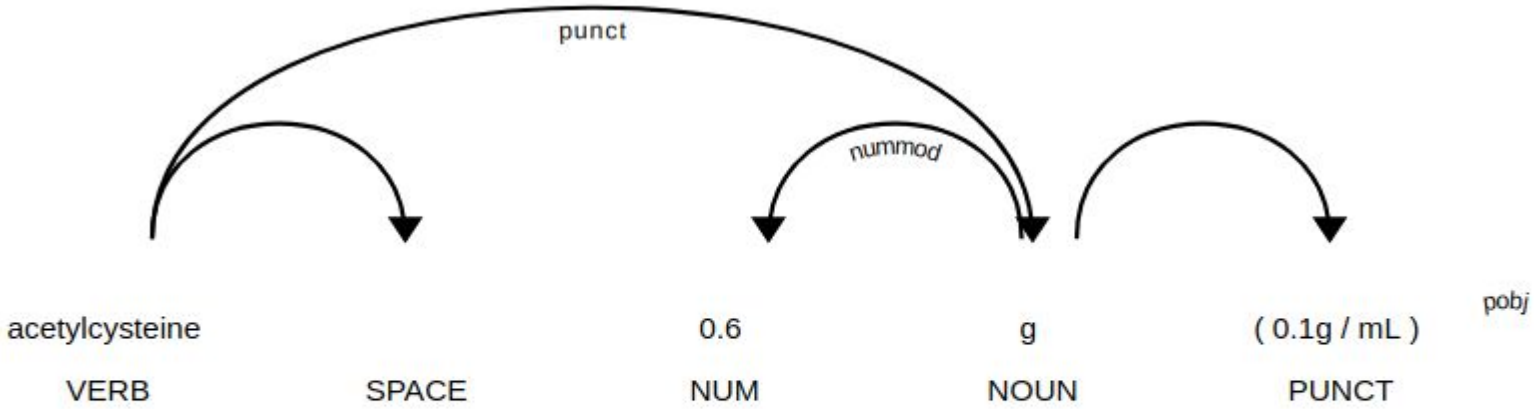

alprostadil α-cyclodextrin **DRUG** 666.7 **DOSAGE** µg **NANOGRAM**

# Test result




acetaminophen ( encapsulated ) **DRUG** | 1.92 **DOSAGE** | g **GRAM** | ( | 32 **DOSAGE** | mg **MILLIGRAM** | / **PER** | mL **MILLILITER** | )

# Test result




acetylcysteine **DRUG** | 0.6 **DOSAGE** | g **GRAM** | ( 0.1g / mL )

# What’s so hard about Named Entity Recognition? *

- This makes progress slow
-
- Structured prediction
- Knowledge intensive
- Mix of easy and hard cases

*https://github.com/explosion/talks/blob/master/2017-11-02___Practical-and-Effective-Neural-NER.pdf

# Next

Behind the NER of spaCy

- Deep learning for NER