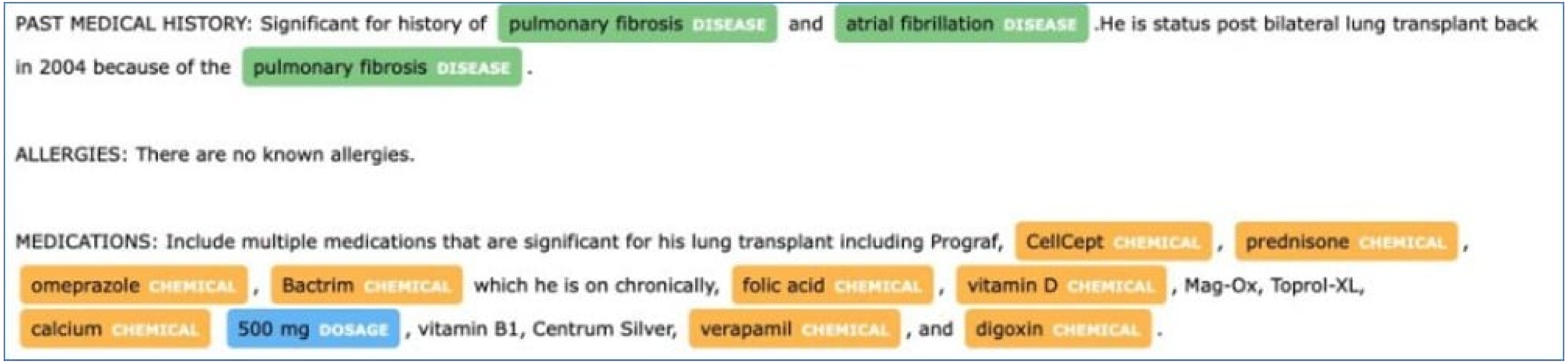
Customizing spaCy models

NATURAL LANGUAGE PROCESSING WITH SPACY

Azadeh Mobasher
Principal data scientist

Why train spaCy models?

- Go a long way for general NLP use cases
- But may **not** have seen **specific domains** data during their training, e.g.
  - **Twitter** data
  - **Medical** data



PAST MEDICAL HISTORY: Significant for history of pulmonary fibrosis DISEASE and atrial fibrillation DISEASE .He is status post bilateral lung transplant back in 2004 because of the pulmonary fibrosis DISEASE .

ALLERGIES: There are no known allergies.

MEDICATIONS: Include multiple medications that are significant for his lung transplant including Prograf, CellCept CHEMICAL , prednisone CHEMICAL , omeprazole CHEMICAL , Bactrim CHEMICAL which he is on chronically, folic acid CHEMICAL , vitamin D CHEMICAL , Mag-Ox, Toprol-XL, calcium CHEMICAL 500 mg DOSAGE , vitamin B1, Centrum Silver, verapamil CHEMICAL , and digoxin CHEMICAL .

Why train spaCy models?

- Better results on your **specific domain**
- Essential for **domain specific text classification**

Before start training, ask the following questions:

- Do `spaCy` models perform well enough on our data?
- Does our domain include many labels that are absent in `spaCy` models?

Models performance on our data

- Do `spaCy` models perform well enough on our data?
- `Oxford Street` is not correctly classified with a `GPE` label:

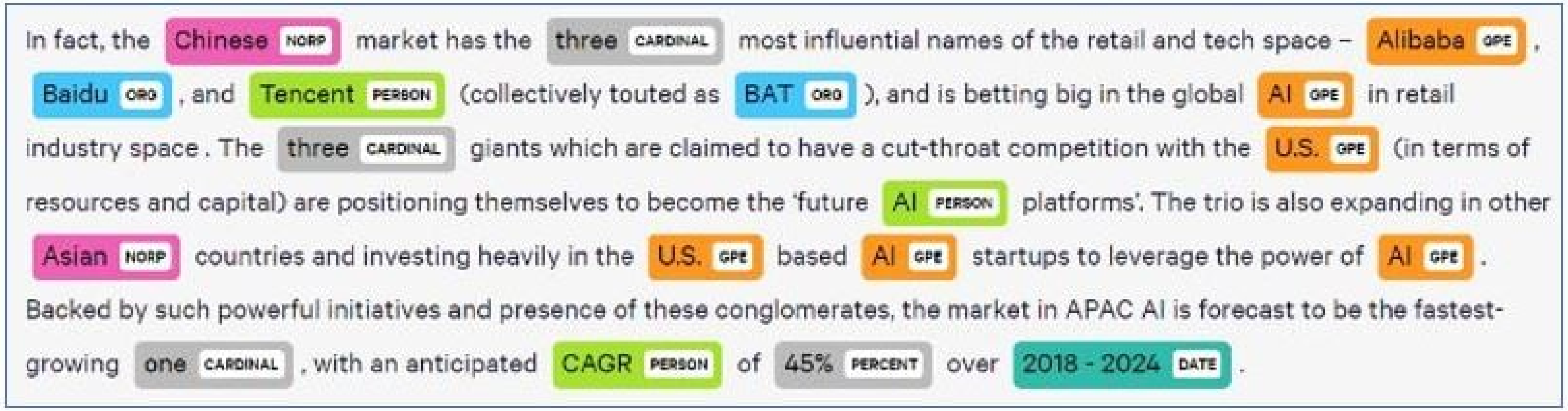
```python
import spacy
nlp = spacy.load("en_core_web_sm")

text = "The car was navigating to the Oxford Street."
doc = nlp(text)
print([(ent.text, ent.label_) for ent in doc.ents])
```

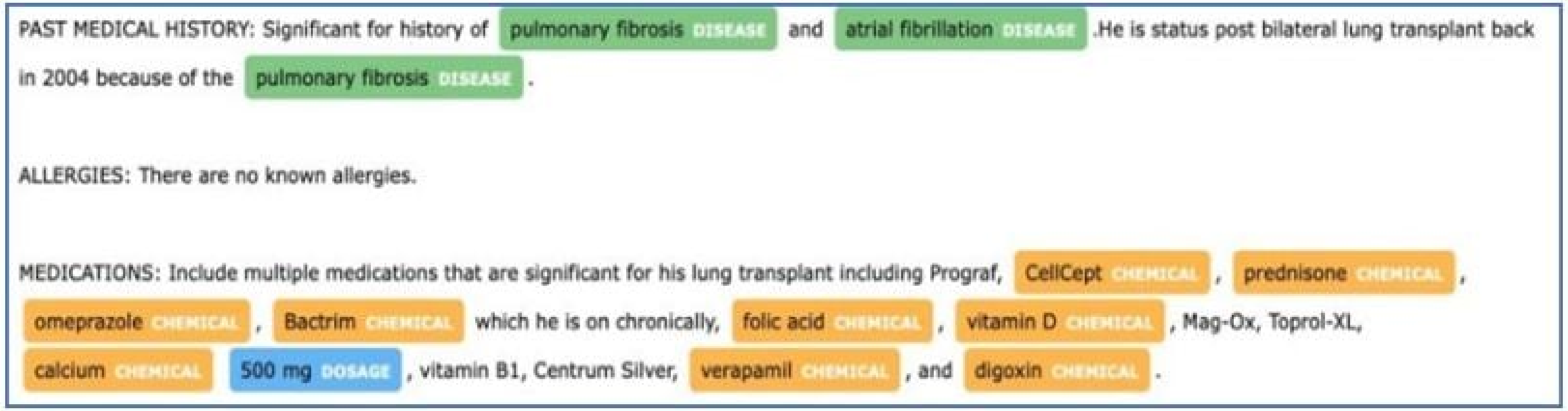
```
[('the Oxford Street', 'ORG')]
```

Output labels in spaCy models

- Does our domain include many labels that are absent in `spaCy` models?



In fact, the Chinese NORP market has the three CARDINAL most influential names of the retail and tech space – Alibaba GPE , Baidu ORG , and Tencent PERSON (collectively touted as BAT ORG), and is betting big in the global AI GPE in retail industry space . The three CARDINAL giants which are claimed to have a cut-throat competition with the U.S. GPE (in terms of resources and capital) are positioning themselves to become the 'future AI PERSON platforms'. The trio is also expanding in other Asian NORP countries and investing heavily in the U.S. GPE based AI GPE startups to leverage the power of AI GPE . Backed by such powerful initiatives and presence of these conglomerates, the market in APAC AI is forecast to be the fastest-growing one CARDINAL , with an anticipated CAGR PERSON of 45% PERCENT over 2018 - 2024 DATE .



PAST MEDICAL HISTORY: Significant for history of pulmonary fibrosis DISEASE and atrial fibrillation DISEASE .He is status post bilateral lung transplant back in 2004 because of the pulmonary fibrosis DISEASE .

ALLERGIES: There are no known allergies.

MEDICATIONS: Include multiple medications that are significant for his lung transplant including Prograf, CellCept CHEMICAL , prednisone CHEMICAL , omeprazole CHEMICAL , Bactrim CHEMICAL which he is on chronically, folic acid CHEMICAL , vitamin D CHEMICAL , Mag-Ox, Toprol-XL, calcium CHEMICAL 500 mg DOSAGE , vitamin B1, Centrum Silver, verapamil CHEMICAL , and digoxin CHEMICAL .

Output labels in spaCy models

If we need custom model training, we follow these steps:

- Collect our domain specific data
- Annotate our data
- Determine to update an existing model or train a model from scratch

Let's practice!

NATURAL LANGUAGE PROCESSING WITH SPACY

Training data preparation

NATURAL LANGUAGE PROCESSING WITH SPACY

Azadeh Mobasher
Principal data scientist

Training steps

1. Annotate and prepare input data
2. Initialize the model weight
3. Predict a few examples with the current weights
4. Compare prediction with correct answers
5. Use optimizer to calculate weights that improve model performance
6. Update weights slightly
7. Go back to step 3.

Annotating and preparing data

- First step is to prepare training data in required format
- After collecting data, we **annotate** it
- **Annotation** means labeling the intent, entities, etc.
- This is an example of annotated data:

```
annotated_data = {
"sentence": "An antiviral drugs used against influenza is neuraminidase inhibitors.",
"entities": {
             "label": "Medicine",
             "value": "neuraminidase inhibitors",
    }
}
```

Annotating and preparing data

- Here's another example of annotated data:

```
annotated_data = {
"sentence": "Bill Gates visited the SFO Airport.",
"entities": [{"label": "PERSON", "value": "Bill Gates"},
             {"label": "LOC", "value": "SFO Airport"}]
}
```

spaCy training data format

- Data annotation prepares training data for what we want the model to learn
- Training dataset has to be stored as a dictionary:

```
training_data = [
("I will visit you in Austin.", {"entities": [(20, 26, "GPE")]}),
("I'm going to Sam's house.", {"entities": [(13,18, "PERSON"), (19, 24, "GPE")]}),
("I will go.", {"entities": []})
]
```

Three example pairs:

- Each example pair includes a sentence as the first element
- Pair's second element is list of annotated entities and start and end characters

Example object data for training

- We cannot feed the raw text directly to spaCy
- We need to create an `Example` object for each training example

```python
import spacy
from spacy.training import Example

nlp = spacy.load("en_core_web_sm")

doc = nlp("I will visit you in Austin.")
annotations = {"entities": [(20, 26, "GPE")]}

example_sentence = Example.from_dict(doc, annotations)
print(example_sentence.to_dict())
```

Let's practice!

NATURAL LANGUAGE PROCESSING WITH SPACY

Training with spaCy

NATURAL LANGUAGE PROCESSING WITH SPACY

Azadeh Mobasher
Principal Data Scientist

Training steps

1. Annotate and prepare input data
2. Disable other pipeline components
3. Train a model for a few epochs
4. Evaluate model performance

Disabling other pipeline components

- **Disable** all pipeline components except **NER:**

```
other_pipes = [pipe for pipe in nlp.pipe_names if pipe != 'ner']

nlp.disable_pipes(*other_pipes)
```

Model training procedure

- Go over the training set several times; one iteration is called an `epoch` .
- In each epoch, update the weights of the model with a small number.
- **Optimizers** update the model weights.

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

```
losses = {}
for i in range(epochs):
  random.shuffle(training_data)
  for text, annotation in training_data:
    doc = nlp.make_doc(text)
    example = Example.from_dict(doc, annotation)
    nlp.update([example], sgd = optimizer, losses=losses)
```

Save and load a trained model

- Save a trained NER model:

```
ner = nlp.get_pipe("ner")
ner.to_disk("<ner model name>")
```

- Load the saved model:

```
ner = nlp.create_pipe("ner")
ner.from_disk("<ner model name>")
nlp.add_pipe(ner, "<ner model name>")
```

Model for inference

- Use a saved model at inference.

- Apply NER model and store tuples of (entity **text**, entity **label**):

```
doc = nlp(text)
entities = [(ent.text, ent.label_) for ent in doc.ents]
```

Let's practice!

NATURAL LANGUAGE PROCESSING WITH SPACY

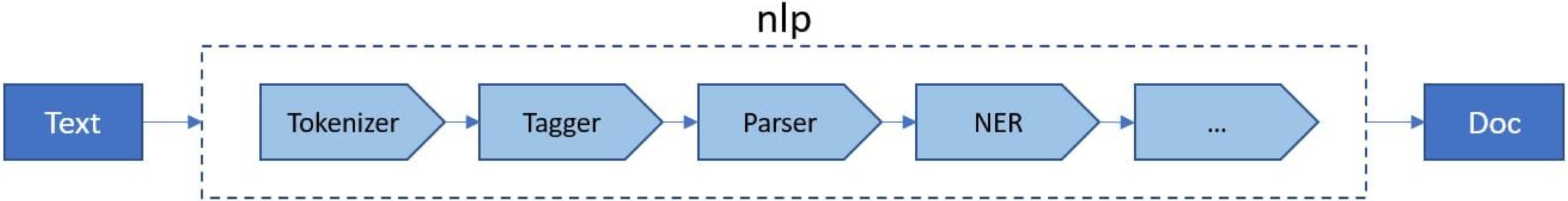
Wrap-up

NATURAL LANGUAGE PROCESSING WITH SPACY

Azadeh Mobasher
Principal data scientist

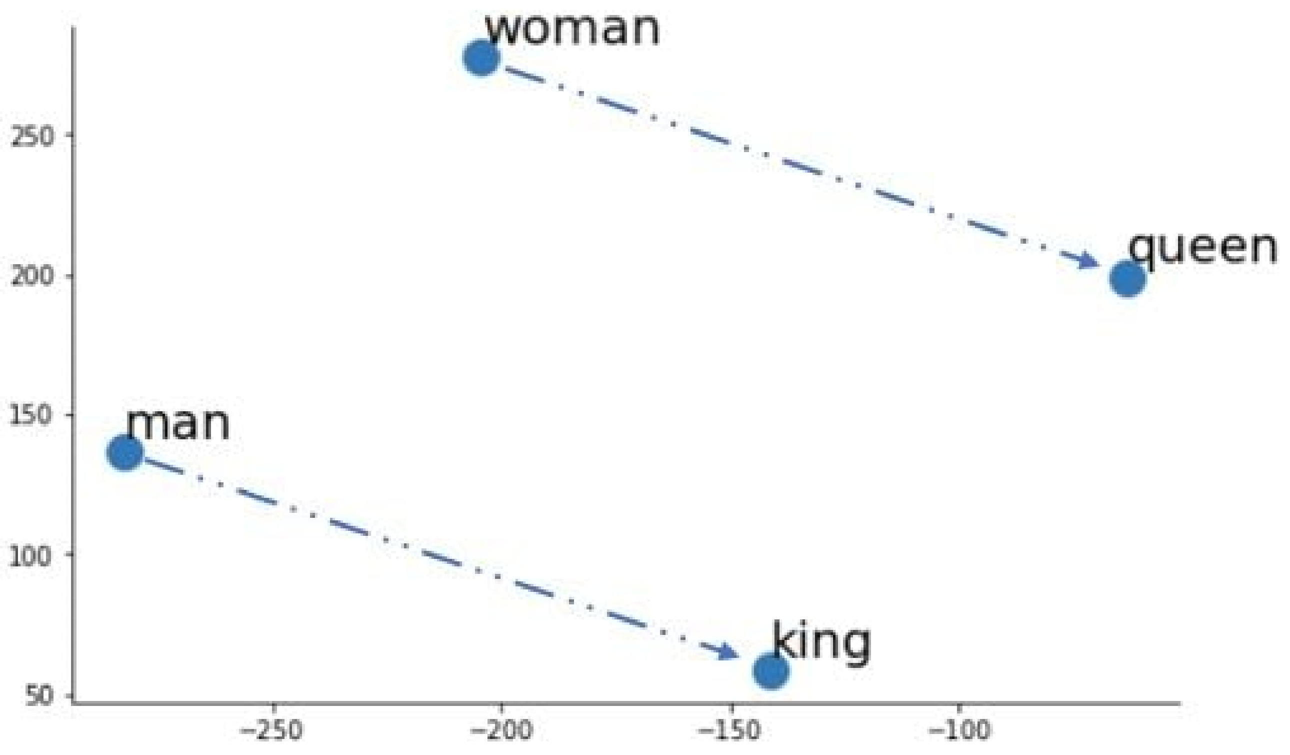
Chapter 1 - Introduction to NLP and spaCy

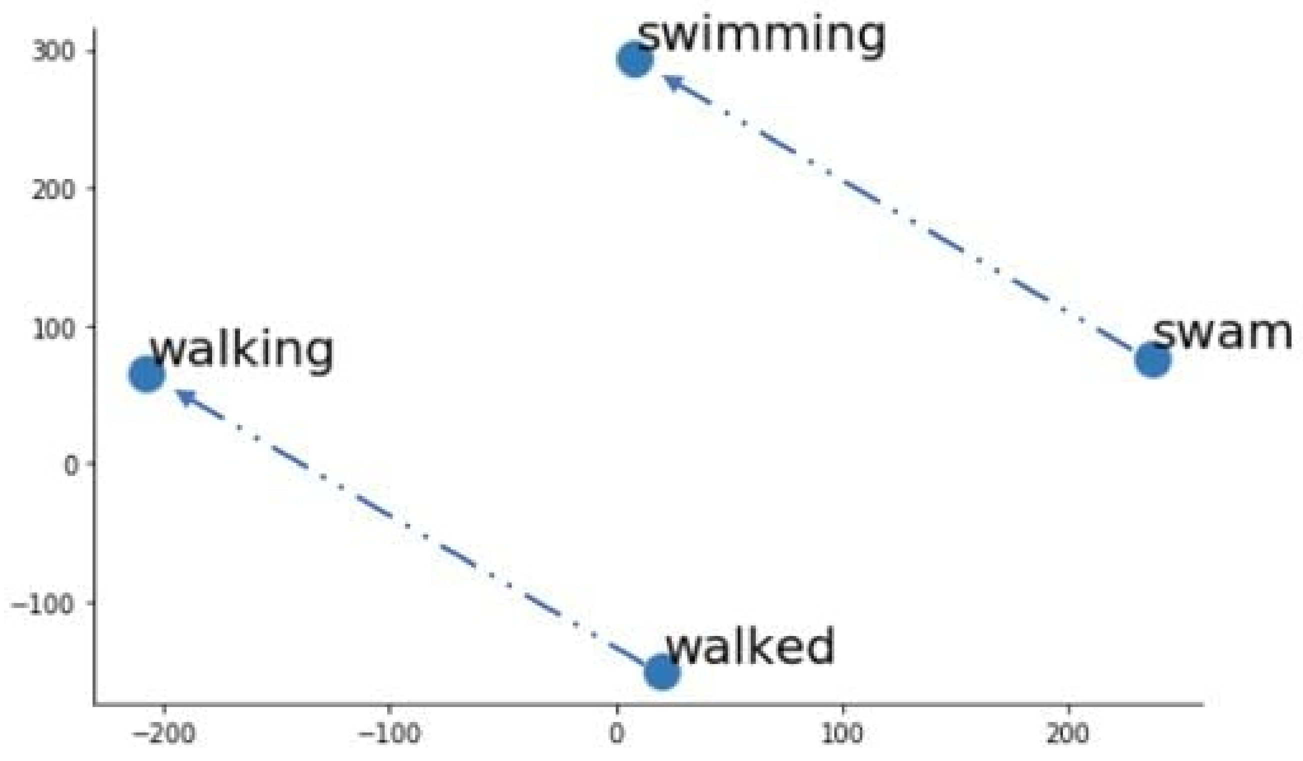
- Use `spaCy` 's text processing pipelines to extract linguistic features:



Chapter 2 - spaCy linguistic annotations and word vectors

- Work with `spaCy` 's classes such as `Doc` , `Token` and `Span` and predict semantic similarities using word vectors:





Chapter 3 - Data analysis with spaCy

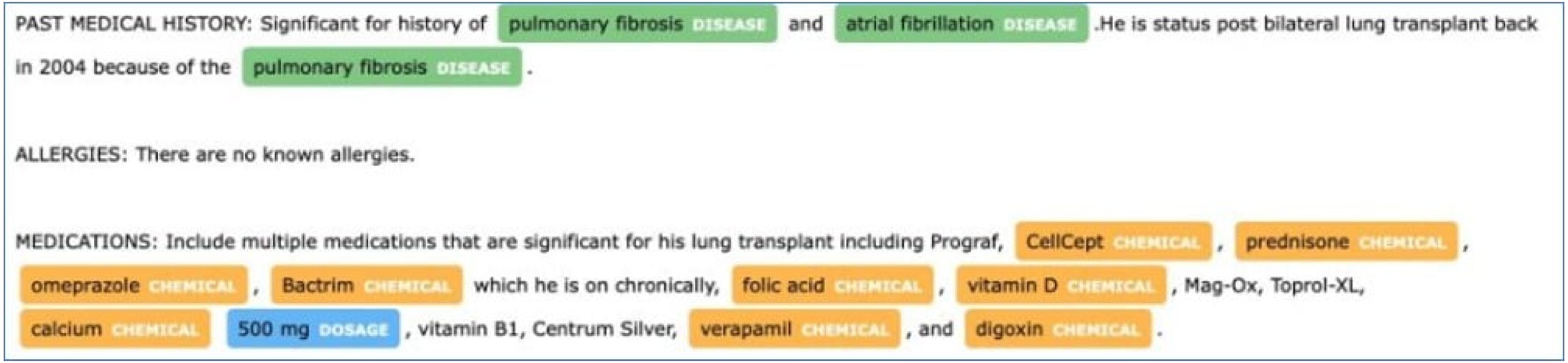
- Write matching patterns to extract terms and phrases using `spaCy` 's `Matcher` and `PhraseMatcher` :

```
matcher = Matcher(nlp.vocab)
pattern = [{"LOWER": "good"}, {"LOWER": {"IN": ["morning", "evening"]}}]
matcher.add("morning_greeting", [pattern])
```

```
matcher = PhraseMatcher(nlp.vocab, attr = "LOWER")
patterns = [nlp.make_doc(term) for term in terms]
matcher.add("InvestmentTerms", patterns)
```

Chapter 4 - Customizing spaCy models

- Annotate and prepare our data for training
- Train `spaCy` models and use them at inference time



PAST MEDICAL HISTORY: Significant for history of pulmonary fibrosis DISEASE and atrial fibrillation DISEASE .He is status post bilateral lung transplant back in 2004 because of the pulmonary fibrosis DISEASE .

ALLERGIES: There are no known allergies.

MEDICATIONS: Include multiple medications that are significant for his lung transplant including Prograf, CellCept CHEMICAL , prednisone CHEMICAL , omeprazole CHEMICAL , Bactrim CHEMICAL which he is on chronically, folic acid CHEMICAL , vitamin D CHEMICAL , Mag-Ox, Toprol-XL, calcium CHEMICAL 500 mg DOSAGE , vitamin B1, Centrum Silver, verapamil CHEMICAL , and digoxin CHEMICAL .

Recommended resources

- Introduction to Deep Learning in Python
- Introduction to Deep Learning with PyTorch
- Introduction to ChatGPT

Congratulations!

NATURAL LANGUAGE PROCESSING WITH SPACY