

Natural Language Processing

Named Entity Recognition

W3 Agenda

- **End-of-course assignments**
- **Named Entity Recognition summary**
- **NER data annotation**
- **Introduction to Spacy**
- **Classification introduction**
- **Error analysis in classification**

End-of-course assignments

To consolidate your learnings you will train your own NLP models using Spacy

Classification

- Classify e-commerce items based on their description
- Maximize f1 score for test set predictions
- Evaluate model performance based on metrics and confusion matrix
- Experiment to see how to improve performance
- Make predictions for completely new items
- Data and code snippets available in Assignment_1_Classification

Named Entity Recognition

- Classify named entities such as people, countries, and organizations
- Maximize recall for each entity
- Evaluate if model performs well across all entities
- Experiment to see how to improve performance
- Extract named entities from new texts
- Data and code snippets available in Assignment_2_NER

Assignment goals

- In groups of 3 prepare coding assignments based on theory and code practiced during our course
- Each group will present their findings during our last lesson 2023-01-26, please send you notebooks/presentations by 2023-01-25 EOD on Teams
- Key goal of this assignment is to learn how to adapt code and knowledge from the course to solve new problems
- Getting best possible model is important, but presenting your discovery journey and evaluating model performance is even more important
- You can present everything with a notebook or prepare a few ppt slides with summary

How to make your work on the assignment easier

- Should you divide work into chunks or focus on coding together?
- Agree who will hold the master code – the task is quite linear so working in parallel will be tough
- Do you understand W3 and W4 material? If not now is the good time to ask as 95% of code needed in the assignment is already there
- Quickly load your data and see if you can successfully start model training – majority of technical issues happen up to this point
- How do you plan improving your model? What parameters can you change in training?
- How do you plan to present your results, what will you analyze?

Discuss these topics in your group and start coding – I would like to discuss these questions with each group towards the end of the lesson

Named entity summary

Named Entity Recognition summary

- We use NER when we want to extract specific information from text
- NER extract structured data from text
- Single NER entity e.g. person can have thousands of instances
- Position of the entity itself is important, which makes data annotation as well as processing especially challenging

geo **gpe** **per**
Lebanon 's top Shi'ite cleric is opposing British Prime Minister
Tony Blair 's expected visit to Beirut Monday .
per **geo** **tim**

```
[(0, 7, 'geo'),  
(42, 49, 'gpe'),  
(50, 55, 'per'),  
(65, 75, 'per'),  
(97, 103, 'geo'),  
(104, 110, 'tim')]
```

org **per**
Democratic U.S. Senator Edward Kennedy has urged the government
to spend more money on education as millions of students return
to school for a new academic year .
tim

```
[(11, 15, 'org'), (16, 38, 'per'), (144, 156, 'tim')]
```


Under-the-hood NER is classifying tokens by their relations to specific entities

Spacy uses BIOES-style labeling by default, where each token in analyzed text can get assigned the following values:

- **"B"**: Beginning of an entity
- **"I"**: Inside an entity
- **"O"**: Outside of any entity

Together with the label it also gets assigned an entity class.

From model perspective we are conducting multilabel classification for each token.

When **Sebastian Thrun** started working on self-driving cars at **Google** in **2007**, few people outside of the company took him seriously. I can tell you very senior CEOs of major **American** car companies would shake my hand and turn away because I wasn't worth talking to, said **Thrun**, in an interview with **Recode** earlier **this week**.

```
Token: Sebastian      ent_iob: B      ent_type: PERSON
Token: Thrun          ent_iob: I      ent_type: PERSON
Token: 2007           ent_iob: B      ent_type: DATE
Token: American       ent_iob: B      ent_type: NORP
Token: Thrun          ent_iob: B      ent_type: PERSON
Token: Recode         ent_iob: B      ent_type: ORG
Token: this           ent_iob: B      ent_type: DATE
Token: week           ent_iob: I      ent_type: DATE
```

Difference between NER and Classification

Classification

- Categorize the overall meaning or sentiment of a sentence or whole text
- Can have multiple labels per text
- It is hard to precisely locate part of text, which influenced the decision

Named Entity Recognition

- Extract short entities (usually 1–3 words), which are important
- One entity per any set of words
- We have clear location within text for each predicted entity

If our Entities become too long or have unclear boundaries we might be facing an actual classification problem.
Example: „This rental offer **includes total monthly costs** – the **additional administrative rent is paid by the owner**”
Can we easily mark the „administrative_rent_included” entity or is it a classification problem?

NER data annotation

Data annotation is usually the toughest part of NER

- Before you start annotation you need to define how many labels you need
- Labeling above 10 labels becomes much slower as they might be harder to distinguish
- If you are not able to make clear guidelines for each label, their definitions are probably too vague
- If you want to mark more than 3-4 words, or even whole sentences your label might be more suitable for classification
- Avoid labels, which have a lot of overlap as they will be challenging both in annotation and training
- Remove html and split words, which mix chars, nums and interpunction – this will help you make sure you can always annotate whole words as an entity

The screenshot displays a web-based NER annotation tool. The main area contains a list of 8 sentences in Polish, each with words highlighted by colored boxes indicating assigned entity labels. The sidebar on the right, titled 'Entity Annotations', lists the following labels: 'fee' (green), 'balcony' (blue), 'standard' (light blue), 'equipment' (yellow), 'monthly_costs_total' (orange), and 'rent_value' (red). Each label has a corresponding colored box next to it and a 'Clear all' button at the top right of the sidebar. The sentences and their annotations are as follows:

Line	Sentence	Annotations
1	do wynajęcia bezpśrednio mieszkanie w warszawie - stary rembertów. 62 m2.	fee: bezpśrednio
2	3 pokoje (salon ok 20 m2, sypialnia1 ok 12 m2, sypialnia2 ok 10 m2, łazienka, oddzielne wc, oddzielna kuchnia).	
3	parter, domofon, podwójne drzwi, bal , balkon .	balcony: bal, balkon
4	blok z lat 80 po wymianie okien i ocieplenia / elewacji.	
5	mieszkanie sta po odświeżeniu , equ umeblowane i wyposażone do wprowadzenia się : meble, 3 kanapy / łóżka, lodówka, pralka, piekarnik, płyta gazowa, naczynia. w pobliżu mieszkania przedszkole, szkoła, przystanek / pętla autobusowa, sklep ogólnospżywczy, basen, place zabaw dla dzieci.	equipment: umeblowane
6	bezpośrednie sąsiedztwo akademii sztuki wojennej.	
7	możliwość podłączenia telefonu / internetu / tv - sat z różnych źródeł.	
8	miesięczny koszt najmu to mit 3000 zł .	monthly_costs_total: 3000

NER data can be saved in many formats

- NER data consists of text, entities and their locations
- Different annotation tools have different format
- Before we can train our model we often need to transform data format

Data format from AWS Sage Maker

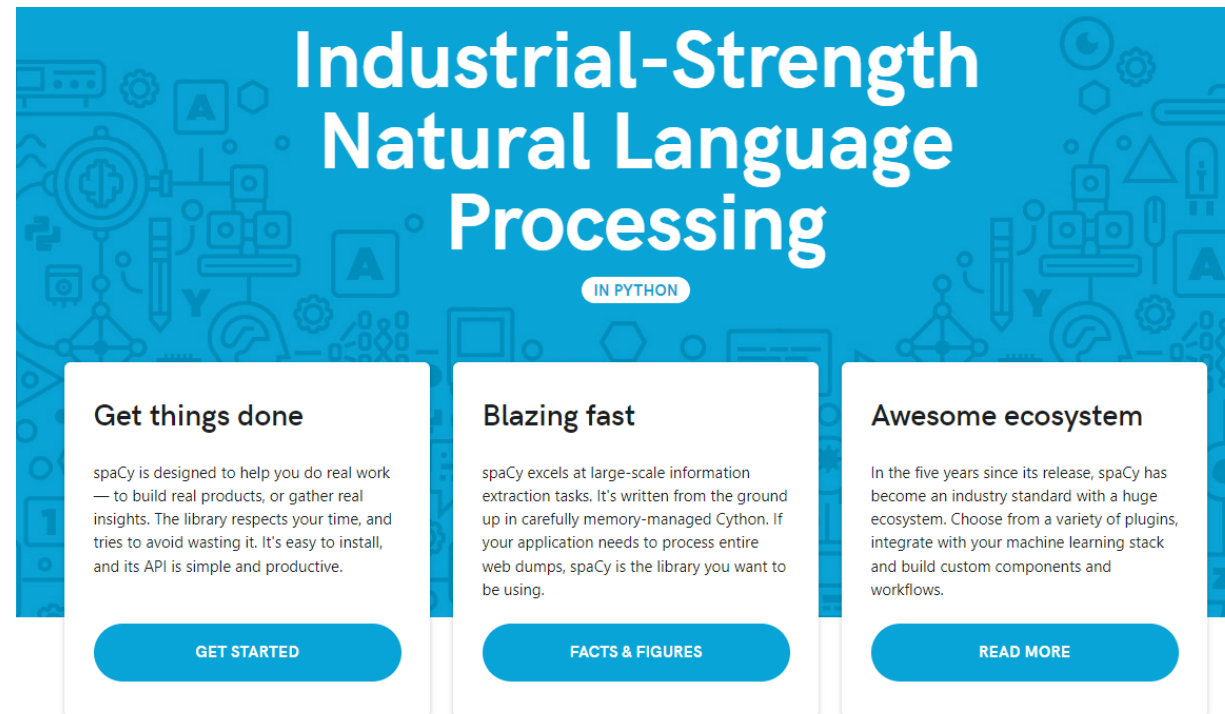
```
{'workerId': 'some-random-worker-no-123',  
  'dataObject': {'content': 'Mr. Egeland said the latest figures show 1.8 million people  
are in need of food assistance - with the need greatest in Indonesia , Sri Lanka , the  
Maldives and India .'},  
  'annotationData': {'content': {'entities': [{ 'endOffset': 11,  
    'label': 'per',  
    'startOffset': 0},  
    { 'endOffset': 128, 'label': 'tim', 'startOffset': 119},  
    { 'endOffset': 134, 'label': 'per', 'startOffset': 131},  
    { 'endOffset': 140, 'label': 'gpe', 'startOffset': 135},  
    { 'endOffset': 155, 'label': 'geo', 'startOffset': 147},  
    { 'endOffset': 165, 'label': 'geo', 'startOffset': 160}]}}}}
```

Final format we need for Spacy NER

```
('Mr. Egeland said the latest figures show 1.8 million people are in need of food  
assistance - with the need greatest in Indonesia , Sri Lanka , the Maldives and India .',  
{ 'entities': [(0, 11, 'per'),  
  (119, 128, 'tim'),  
  (131, 134, 'per'),  
  (135, 140, 'gpe'),  
  (147, 155, 'geo'),  
  (160, 165, 'geo')]}])
```

What is Spacy

- Spacy is one of the most popular libraries for NLP
- It has a low barrier of entry
- You can build models within a few dozens lines of code in a notebook
- CLI interface great for building production models and replicability

The graphic is a landing page for SpaCy, featuring a blue background with white icons representing various NLP concepts like neural networks, data flow, and language processing. The main title 'Industrial-Strength Natural Language Processing' is in large white font, with 'IN PYTHON' in a smaller white box below it. Three white boxes with blue borders contain key features: 'Get things done', 'Blazing fast', and 'Awesome ecosystem'. Each box has a blue button at the bottom with white text: 'GET STARTED', 'FACTS & FIGURES', and 'READ MORE' respectively.

Industrial-Strength Natural Language Processing

IN PYTHON

Get things done

spaCy is designed to help you do real work — to build real products, or gather real insights. The library respects your time, and tries to avoid wasting it. It's easy to install, and its API is simple and productive.

GET STARTED

Blazing fast

spaCy excels at large-scale information extraction tasks. It's written from the ground up in carefully memory-managed Cython. If your application needs to process entire web dumps, spaCy is the library you want to be using.

FACTS & FIGURES

Awesome ecosystem

In the five years since its release, spaCy has become an industry standard with a huge ecosystem. Choose from a variety of plugins, integrate with your machine learning stack and build custom components and workflows.

READ MORE

Jupyter exercise NER:

Proceed to notebook W3_NER_BLANK

1 Spacy intro

In [18]:

```
import spacy

# Load English tokenizer, tagger, parser and NER
nlp = spacy.load("en_core_web_sm")

# Process whole documents
text = ("When Sebastian Thrun started working on self-driving cars at "
        "Google in 2007, few people outside of the company took him "
        "seriously. "I can tell you very senior CEOs of major American "
        "car companies would shake my hand and turn away because I wasn't "
        "worth talking to," said Thrun, in an interview with Recode earlier "
        "this week.")
```

In [19]:

```
doc = nlp(text)
```

In [20]:

```
# Analyze syntax
print("Noun phrases:", [chunk.text for chunk in doc.noun_chunks])
print("Verbs:", [token.lemma_ for token in doc if token.pos_ == "VERB"])
```

```
Noun phrases: ['Sebastian Thrun', 'self-driving cars', 'Google', 'few people', 'the company', 'him', 'I', 'you', 'very senior C
EOs', 'major American car companies', 'my hand', 'I', 'Thrun', 'an interview', 'Recode']
Verbs: ['start', 'work', 'drive', 'take', 'tell', 'shake', 'turn', 'talk', 'say']
```

In [21]:

```
# Find named entities, phrases and concepts
for entity in doc.ents:
    print(entity.text, entity.label_)
```

```
Sebastian Thrun PERSON
Google ORG
2007 DATE
American NORP
Thrun PERSON
Recode ORG
earlier this week DATE
```

1.1 Token labels

In []:

```
for token in doc:
    if token.ent_iob_ != 'O':
        print(f"Token: {token.text}\tent_iob: {token.ent_iob_}\tent_type: {token.ent_type_}")
```

2 Preparing Spacy format

In [24]:

```
with open('../data/GMB_data_sagemaker.pickle', 'rb') as f:
    sagemaker_data = pickle.load(f)
```

In [85]:

```
## Current format
sagemaker_data[0]
```

```
Out[85]: {'workerId': 'some-random-worker-no-123',
'dataObject': {'content': 'Mr. Egeland said the latest figures show 1.8 million people are in need of food assistance - with t
he need greatest in Indonesia , Sri Lanka , the Maldives and India .'},
'annotationData': {'content': {'entities': [{'endOffset': 11,
'label': 'per',
'startOffset': 0},
{'endOffset': 128, 'label': 'tim', 'startOffset': 119},
{'endOffset': 134, 'label': 'per', 'startOffset': 131},
{'endOffset': 140, 'label': 'gpe', 'startOffset': 135},
{'endOffset': 155, 'label': 'geo', 'startOffset': 147},
{'endOffset': 165, 'label': 'geo', 'startOffset': 160}]}}}}
```

Classification introduction

Text Classification

Text classification basics

- Text Classification aims to divide texts into clear classes
- We can classify binary classes like spam vs normal email
- Or multiple classes like e-commerce category
- It is usually easier to work with exclusive classes but we can also have multiple classes per text

Typical segmentation criteria

- Topic e.g. classifying messages from customers
- Positive vs Negative e.g. review, sentiment
- Business area
- Item categorization
- Hate speech detection
- True vs fake news
- Fake accounts

When classification works best... and when it doesn't

Common use cases

- Spam classification
- Hate speech detection
- Categorization e.g. article topic, item category

Other approach might work better

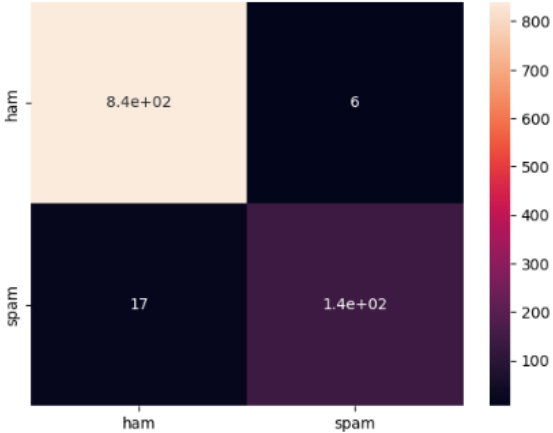
- Problems, where labels can be sorted e.g. sentiment analysis, where we have more than positive/negative – ordinal regression might work better
- When we are looking for a high number of categories or very specific information, where we should choose NER

Error analysis in classification

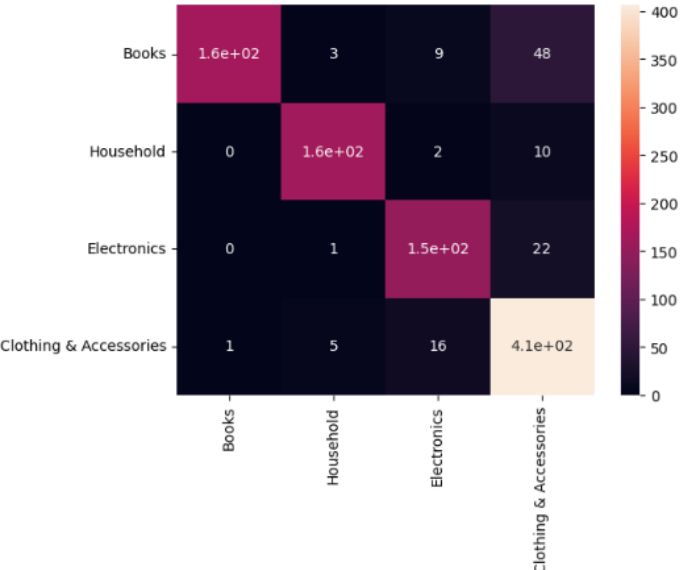
Confusion matrix

		Predicted	
		Positive	Negative
True	Positive	True positive (TP)	False negative (FN)
	Negative	False positive (FP)	True negative (TN)

Binary



Multi class



Precision and Recall

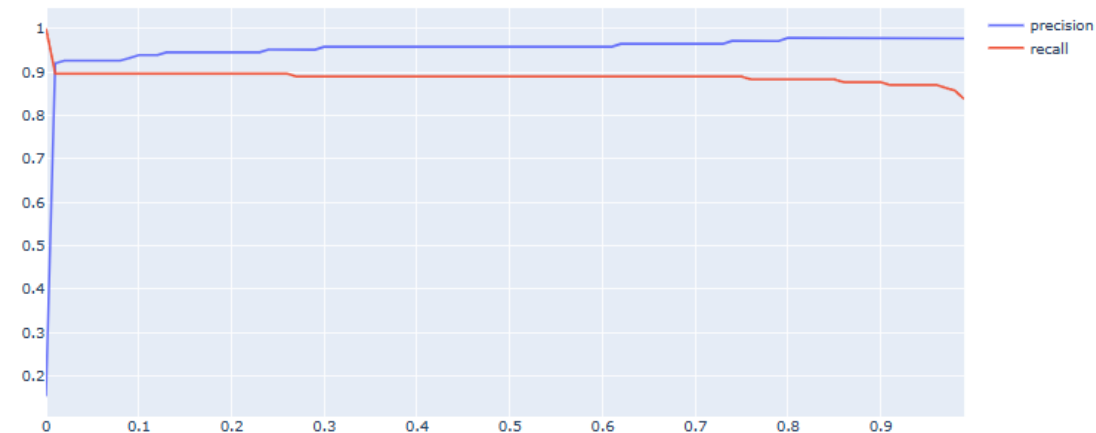
How many of our predictions are actually right?

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

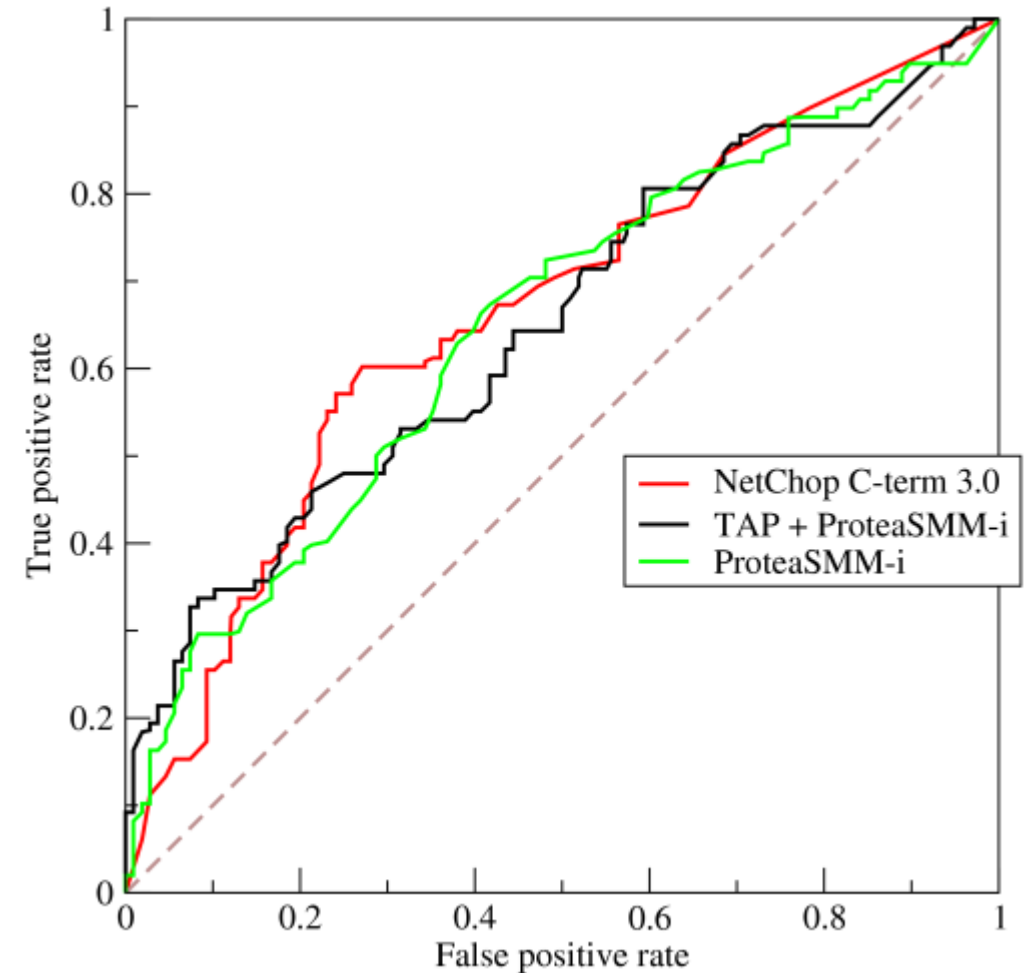
What share of all the instances we were looking for did we actually find?

Precision and Recall by threshold



Receiver operating characteristic (ROC)

- Choosing the right model and the right decision threshold can be tough – ROC curve is one of the most common evaluation criterias
- We can compare multiple models, with different performances depending on threshold by analyzing the area under the curve (AUC) – larger area = better model



Jupyter exercise NER:

Proceed to notebook
W3_Classification_BLANK

1 Spacy intro

In [18]:

```
import spacy

# Load English tokenizer, tagger, parser and NER
nlp = spacy.load("en_core_web_sm")

# Process whole documents
text = ("When Sebastian Thrun started working on self-driving cars at "
        "Google in 2007, few people outside of the company took him "
        "seriously. "I can tell you very senior CEOs of major American "
        "car companies would shake my hand and turn away because I wasn't "
        "worth talking to," said Thrun, in an interview with Recode earlier "
        "this week.")
```

In [19]:

```
doc = nlp(text)
```

In [20]:

```
# Analyze syntax
print("Noun phrases:", [chunk.text for chunk in doc.noun_chunks])
print("Verbs:", [token.lemma_ for token in doc if token.pos_ == "VERB"])
```

```
Noun phrases: ['Sebastian Thrun', 'self-driving cars', 'Google', 'few people', 'the company', 'him', 'I', 'you', 'very senior C
EOs', 'major American car companies', 'my hand', 'I', 'Thrun', 'an interview', 'Recode']
Verbs: ['start', 'work', 'drive', 'take', 'tell', 'shake', 'turn', 'talk', 'say']
```

In [21]:

```
# Find named entities, phrases and concepts
for entity in doc.ents:
    print(entity.text, entity.label_)
```

```
Sebastian Thrun PERSON
Google ORG
2007 DATE
American NORP
Thrun PERSON
Recode ORG
earlier this week DATE
```

1.1 Token labels

In []:

```
for token in doc:
    if token.ent_iob_ != 'O':
        print(f"Token: {token.text}\tent_iob: {token.ent_iob_}\tent_type: {token.ent_type_}")
```

2 Preparing Spacy format

In [24]:

```
with open('../data/GMB_data_sagemaker.pickle', 'rb') as f:
    sagemaker_data = pickle.load(f)
```

In [85]:

```
## Current format
sagemaker_data[0]
```

Out[85]:

```
{'workerId': 'some-random-worker-no-123',
 'dataObject': {'content': 'Mr. Egeland said the latest figures show 1.8 million people are in need of food assistance - with t
he need greatest in Indonesia , Sri Lanka , the Maldives and India .'},
 'annotationData': {'content': {'entities': [{'endOffset': 11,
      'label': 'per',
      'startOffset': 0},
      {'endOffset': 128, 'label': 'tim', 'startOffset': 119},
      {'endOffset': 134, 'label': 'per', 'startOffset': 131},
      {'endOffset': 140, 'label': 'gpe', 'startOffset': 135},
      {'endOffset': 155, 'label': 'geo', 'startOffset': 147},
      {'endOffset': 165, 'label': 'geo', 'startOffset': 160}]}}}}
```