



AWS Academy Natural Language Processing
Module 05 Student Guide
Version 0.1.0
200-ACMNLP-01-EN-SG

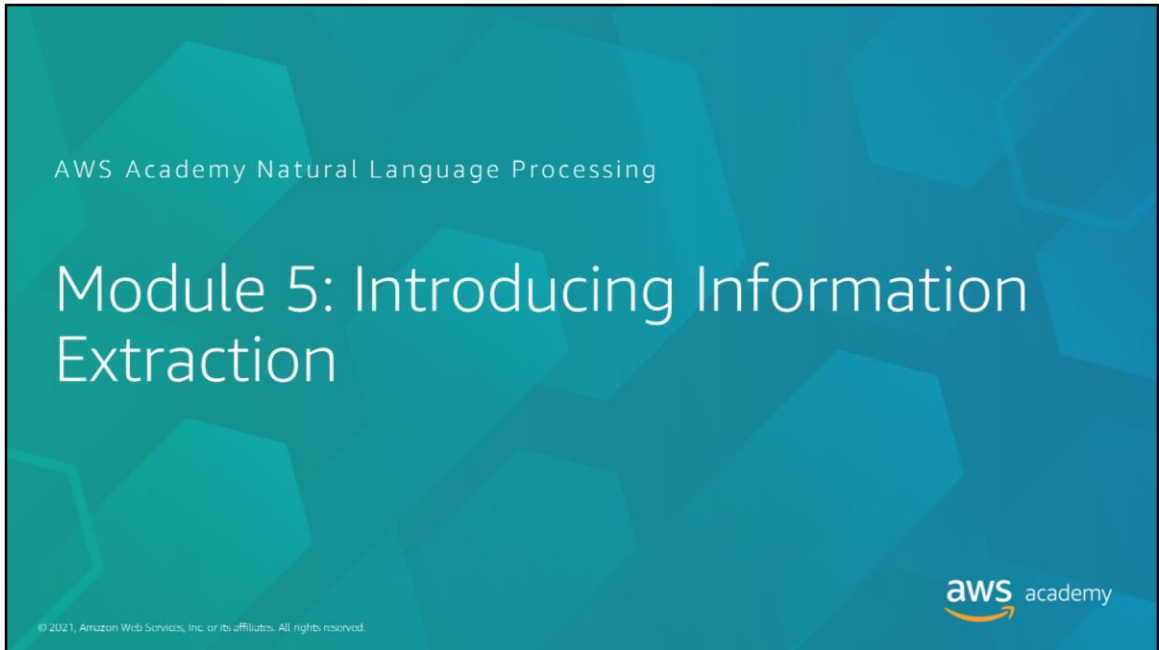
© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved.

This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited.

All trademarks are the property of their owners.


Contents

Module 5: Introducing Information Extraction	4
--	---



Welcome to Module 5: Introducing Information Extraction.

Module overview



Sections

- Section 1: Information extraction overview
- Section 2: Types of information extraction
- Section 3: Implementing information extraction

Labs

- Guided Lab: Implementing Information Extraction
- Guided Lab: Working with Entities

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.2

In this module, you learn about the following topics:

1. Information extraction overview – You learn what information extraction is and the typical use cases for it.
2. Types of information extraction – You examine the different types of information extraction and the problems that they can help solve.
3. Implementing information extraction – You look at some of the AWS services and open-source tools for information extraction.

The module includes two guided labs, where you will implement information extraction and work with entities.

Module objectives



At the end of this module, you should be able to:

- Understand the terms used in information extraction
- Describe the steps for information extraction
- Implement key phrase extraction (KPE)
- Extract entities from text

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

5

At the end of this module, you should be able to:

- Understand the terms used in information extraction
- Describe the steps for information extraction
- Implement key phrase extraction (KPE)
- Extract entities from text

Module 5: Introducing Information Extraction

Section 1: Information extraction overview

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Introducing Section 1: Information extraction overview.

What is information extraction?



- Information extraction (IE) refers to the natural language processing (NLP) task of extracting relevant information from text.
- IE can be a challenge with unstructured data.

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

5

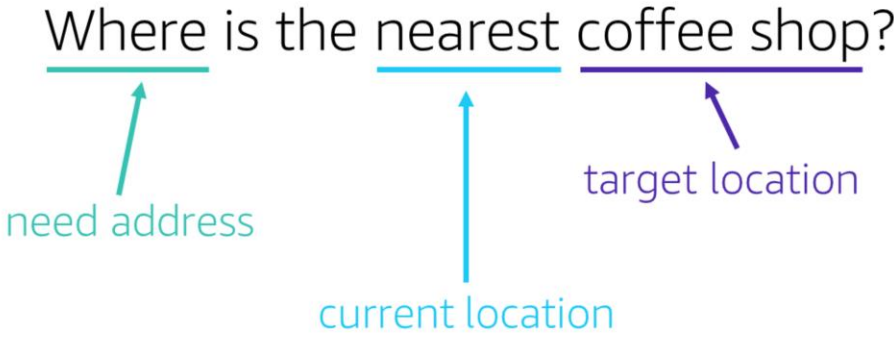
Information extraction (IE) is the natural language processing (NLP) task of extracting information from text. With structured data that is stored in a database, IE can be simple because column names, such as *name*, *address*, and *telephone number*, often label the data for you. With text, the data is unstructured, so you need different ways to extract the data.

A collection of movie reviews written by both professional movie reviewers and the public is a good example of unstructured data. The data is unstructured because there is no standard set of fields with a set length or data type, and there are no pre-determined relationships between the individual data objects. This lack of structure makes this sort of dataset an ideal candidate for information extraction with machine learning.

Tags that are created on Amazon.com product reviews are an example of IE. IE is used to extract key phrases from reviews to help customers find reviews that are relevant to their needs.

In the next few slides you will learn more about information extraction.

IE applications: Digital assistant queries




Where is the nearest coffee shop?

need address

current location

target location



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

b

IE has a number of key applications.

For example, a digital assistant must understand a question to form a response. If you ask, “Where is the nearest coffee shop?”, the digital assistant must understand that **coffee shop** is a location. The assistant must understand that you need the one **nearest** to your current location, and most likely, the assistant requires the address (from the **Where** statement). IE helps to extract the useful information from the sentence so that the digital assistant can act on the information.

IE applications: Common phrases



Read reviews that mention

easy to install video card power supply great value
triple A titles ultra settings liquid cooling hdmi
ports gddr6 overclock cable management fps
make sure installation every game

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

/

Another key application of IE is identifying common themes in reviews, as the example shows. When customers are researching products with many reviews, important phrases can be useful to help them filter the reviews that they want to read.

IE applications: Tagging news



Trending news topics

sanctions

US Supreme Court

vaccine

Texas

snow storms

climate change

nuclear power

virus

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

8

Tagging news is another application of IE. Extracting names, events, and locations from news articles can help identify trends or highlight important entities within them.

IE applications: Extracting financial data



Amazon (NASDAQ:AMZN) and Whole Foods Market, Inc. (NASDAQ:WFM) today announced that they have entered into a definitive merger agreement under which Amazon will acquire Whole Foods Market for \$42 per share in an all-cash transaction valued at approximately \$13.7 billion, including Whole Foods Market's net debt. "Millions of people love Whole Foods Market because they offer the best natural and organic foods, and they make it fun to eat healthy," said Jeff Bezos, Amazon founder and CEO. "Whole Foods Market has been satisfying, delighting and nourishing customers for nearly four decades – they're doing an amazing job and we want that to continue."

Known entity:
business

Known entity:
person

Relationship

Numeric
data

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

9

Identifying financial data in reports is another IE application. Key performance metrics, people, events, and relationships are all types of useful information to extract from financial results data.

IE applications: Extracting events



Whoop! Whoop! I just got a promotion to manager at AnyCompany!!!

Hi!
Don't forget that next Tuesday, March 9, 2021, is the new date for the Charity Ball at the Charming Community Center! See you at 7PM sharp!

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.10

You can also use IE to extract events from data. Events can be anything that happens at a certain point in time, such as meetings, elections, stock price changes, births, and promotions. In the first example, *promotion* is the event that is detected. The detected event could also have been *promotion to manager*.

In the second example, events are detected in an email. Many email systems use temporal information (for example, time and location) to identify events from emails and generate calendar reminders or actions based on the extracted information.

Section 1: Summary



- What is information extraction?
- Information extraction applications
 - Digital assistant extracting information from a question
 - Identifying common phrases or themes in reviews
 - Tagging news
 - Identifying financial data in reports
 - Extracting event information

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

11

In this section, you looked at what information extraction is and key applications of information extraction. IE applications include digital assistants extracting information to understand a question, identifying common phrases or themes in product reviews, tagging news topics, identifying financial data in reports, and extracting event information from emails.

In the next section, you look at types of information extraction.

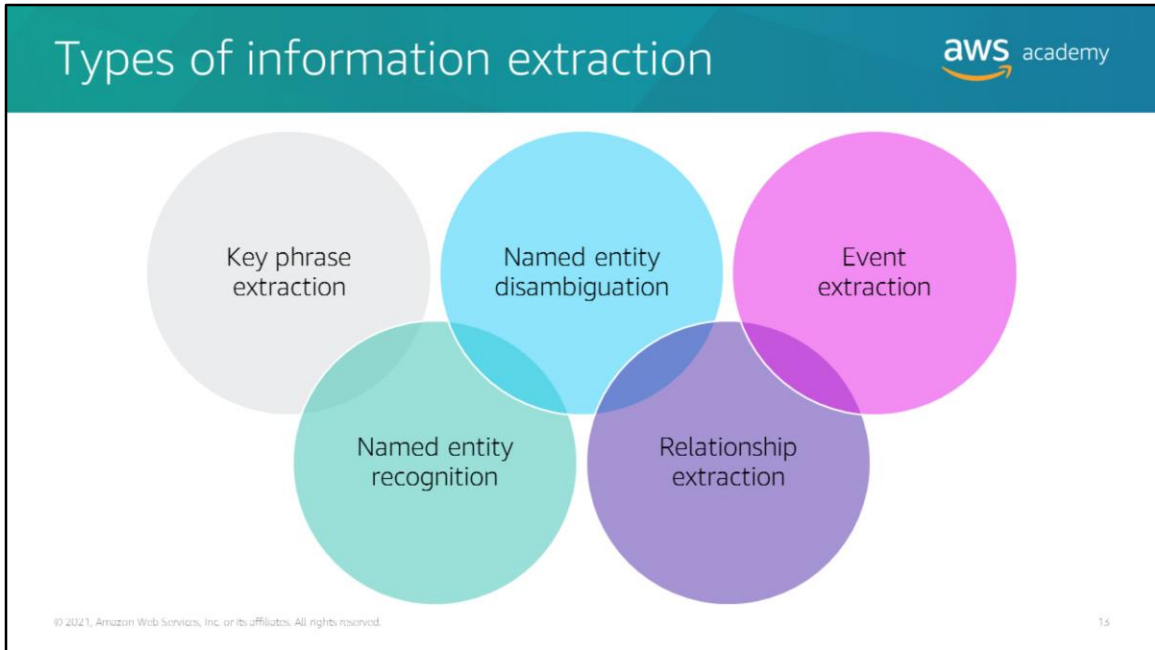
Module 5: Introducing Information Extraction

Section 2: Types of information extraction

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.



In this section, you look at the different types of information extraction.



This module provides information about five different types of information extraction:

- **Key phrase extraction (KPE):** Extracting useful words or phrases from text
- **Named entity recognition (NER):** Identifying entities such as people names, locations, organizations, dates, and product names
- **Named entity disambiguation (NED):** Uniquely identifying entities in text
- **Relationship extraction:** Identifying relationships between entities
- **Event extraction:** Extracting knowledge about events in text, such as stock splits or initial public offerings (IPOs)

Key phrase extraction (KPE)



- KPE extracts useful words and phrases from text to capture the general intent.
- KPE is generally unsupervised learning.
- Words are nodes in a weighted graph:
 - Weight indicates the importance of a key phrase.
 - How well are the nodes connected with the rest of the graph?
 - The top X nodes are returned.



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

14

We will start with **key phrase extraction**, or KPE. *Key phrase extraction (KPE)* extracts useful words and phrases from text. These phrases capture the general intent of the text.

KPE is generally an unsupervised machine learning problem. A few algorithms can do KPE. The general approach is to organize the words or phrases into a weighted graph, where the weight indicates the importance of the key phrase. Nodes that are more connected with the rest of the graph are more important. Typically, the top X nodes are returned. The Word2Vec visual from Module 3 is an excellent way to visualize this process. That visual, which is shown here, used T-distributed stochastic neighbor embedding, also known as t-SNE, Term frequency-inverse document frequency (TF-IDF) is a weight that is used to evaluate how important a word is to a document in a corpus. Common words in a document that are not common in the corpus are likely to be important words for that document.

KeyBERT is a minimal KPE technique. KeyBERT uses Bidirectional Encoder Representations from Transformers (BERT) embeddings to create keywords and key phrases that are most similar to a document.

Amazon Comprehend provides an application programming interface (API) for performing KPE.

Determining word significance



- Term frequency-inverse document frequency (TF-IDF) and KeyBERT are used to determine significance of a word.
- Amazon Comprehend provides application programming interface (API) actions for KPE.

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

15

Term frequency-inverse document frequency or TF-IDF, is a weight that is used to evaluate how important a word is the a document in a corpus. Common words in a document that are not common in the corpus are likely to be important words for that document.

KeyBERT is a minimal keyword extraction technique that uses Bidirectional Encoder Representations from Transformers, or BERT, embeddings. KeyBERT uses these embeddings to create keywords and key phrases that are most similar to a document.

Amazon Comprehend provides an application programming interface, or API, for performing KPE.

Named entity recognition (NER)



- NER is the NLP task of identifying entities such as names, locations, organizations, dates, and product names.
- NER is often needed for downstream tasks such as event or relation extraction and translation.
- spaCy and BERT provide tools for extracting and visualizing named entities
- Amazon Comprehend provides API actions for entity recognition to find insights and relationships in text.

Alexa, where was Alan Turing born?
Which companies were in the news?
Which products are mentioned in the review?

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

1b

Named entity recognition (NER) is the NLP task of identifying entities such as people names, locations, organizations, dates, and product names.

NER is often necessary to process downstream tasks such as the following:

- **Event extraction** – Extracting event names, locations, and dates
- **Relation extraction** – For example, “London is in England”
- **Translation** – Names do not need to be translated. By identifying named entities you can increase the efficiency of translation.

spaCy has an excellent set of tools for extracting and visualizing named entities. You can also use BERT.

Amazon Comprehend provides an API for entity recognition API. You can use this API without machine learning knowledge.

Named entity disambiguation (NED)



- NED is the NLP task of uniquely identifying entities in the text.

Ford forded the ford in the Ford.

- NER and NED enable named entity linking (NEL).
 - NEL maps named entities to specific instances in a knowledge base

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

1/

Named entity disambiguation (NED) is the task of uniquely identifying entities in text.

The example “Ford forded the ford in the Ford” contains a person, an action, a shallow body of water, and a car, which are all called *ford*. You first need to identify the entities of interest. When the entities are identified, you can then link them to more data on the topic. Often, this link is to a web article or entry in a knowledge base. The ford example is complicated because which ford to link to depends on the ford in question. The context in which *ford* is used can be helpful to decide whether to link to the car company, the body of water, the person, or the action.

For another fun example, search the internet for the following grammatically correct sentence: “Buffalo buffalo Buffalo buffalo buffalo buffalo Buffalo buffalo.”

Together, NER and NED enable *named entity linking (NEL)*. NEL is a key task for digital assistants and search engines to return the correct results. For example, if you search for news about Paris, an assistant needs to figure out whether “Paris” means the person or the place.

spaCy has tools to identify entities based on a knowledge base that you must create.

Relationship extraction



- Relationship extraction is the NLP task that identifies relationships between entities.

"Millions of people love Whole Foods Market because they offer the best natural and organic foods, and they make it fun to eat healthy," said Jeff Bezos, Amazon founder and CEO.

- How do you extract that Jeff Bezos is the Chief Executive Officer of Amazon?

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

18

Relationship extraction is the NLP task that deals with identifying relationships between entities. This task helps with applications such as Q&A systems, digital assistants, search, and text summarization.

In the example that is shown, it is easy to conclude that Jeff Bezos is the CEO of Amazon. To establish this information in NLP, you first must identify entities by using a named entity recognizer. Then, you must determine whether a semantic relationship exists between them. A number of techniques are available to determine this relationship:

1. **Rule based** defines a set of rules for the syntax and grammar to extract the relationship. These patterns look similar to "**X** such as **Y**," which could match "**..musical instruments** such as **guitars...**".
2. **Supervised** requires a large labeled dataset of relationships to learn from.
3. **Semi-supervised** uses a few labeled examples to seed the data, which is then used to extract more relationships.

Amazon Comprehend Medical provides relationship extraction that can tie information together. For example, consider the phrase "left chest pain". *Pain* is extracted as the medical condition, with *chest* as the anatomical location and *left* as the direction.

spaCy provides a set of rule-based matching, which can be used to build relationships.

Event extraction



- Event extraction is the process of identifying and extracting knowledge about events in text data.
- You often use pattern matching and NER in event extraction.

AnyCompany announced today a 2-to-1 stock split, which will be effective on May 24, 2021, at market open.

- NER uses to extract AnyCompany
- Pattern matching used to extract the stock split event.
- May 24, 2021 recognized as a date.

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

19

Event extraction is the process of extracting knowledge about events in text. Events can be anything from meetings, to stock price changes, to elections.

Event extraction is challenging. An event must be first detected in the text, and then the key event information, such as participants, location, and date, must be extracted.

Consider the example “AnyCompany announced today a 2-to-1 stock split, which will be effective on May 24, 2021, at market open.” One approach would be to detect the event, in this case “stock split”, by using pattern matching. From that result, you could find the entities that are detected in the sentence to find the organization. Then, you could look for the date and numerical values for the split by using pattern matching.

Amazon Comprehend has functionality to detect events. These events include bankruptcy, employment, corporate acquisition, investment, corporate mergers, IPOs, rights issues, secondary offerings, shelf offerings, tender offerings, and stock splits.

Section 2: Summary



- Key phrase extraction
- Named entity recognition
- Named entity disambiguation
- Relationship extraction
- Event extraction

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

23

In this section, you learned five different types of information extraction:

- Key phrase extraction extracts useful words or phrases from text.
- Named entity recognition identifies entities such as people's names, locations, organizations, dates, and product names.
- Named entity disambiguation uniquely identifies entities in text.
- Relationship extraction identifies relationships between entities.
- Event extraction extracts knowledge about events in text, such as stock splits or IPOs.

In the next section, you look at tools to perform information extraction.

Module 5: Introducing Information Extraction

Section 3: Implementing information extraction

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.



In this section, you look at some of the AWS services and open-source tools for information extraction.

IE with Amazon Comprehend



Amazon Comprehend uses pretrained models to do the following:

- Detect entities
- Detect events
- Detect key phrases
- Detect personally identifiable information (PII)
- Analyze syntax

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

22

Amazon Comprehend provides a number of asynchronous jobs through the API for tasks that include sentiment analysis, language detection, and topic modeling. For general information extraction, a number of capabilities are relevant:

- **Detect entities** – Detect textual references to the names of people, places, and items, and references to dates and quantities
- **Detect events** – Analyze documents to detect specific types of events, such as mergers, bankruptcy, and employment
- **Detect key phrases** – Find key phrases such as “good morning” in a document or set of documents
- **Detect personally identifiable information (PII)** – Analyze documents to detect personal data that could be used to identify an individual, such as address, bank account number, or phone number
- **Analyze syntax** – Parse the words in your text to show the speech syntax for each word, which helps you to understand the content of the document

You can run these jobs against your documents with little data preparation or knowledge of machine learning. To understand and use the results that are returned does require some knowledge of how to handle the returned data.

Entity detection with Amazon Comprehend



- `StartEntitiesDetectionJob`: Starts an asynchronous job
- Input and output to Amazon Simple Storage Service (Amazon S3)
- UTF-8 with a max size of 102,400 bytes

```
{
  "File": "small_doc",
  "Entities": [{
    "BeginOffset": 0,
    "EndOffset": 4,
    "Score": 0.645766019821167,
    "Text": "Maat",
    "Type": "PERSON"}]
```

location of entity in document

confidence level

text

type

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

25

Before you can start a job, you must prepare your data. You can store data in a single file with one document per line or have each document in its own file. Regardless, the maximum size of a document is 102,400 bytes, and it must be in UTF-8 format. Input files must be stored in an Amazon Simple Storage Service (Amazon S3) bucket.

To start an asynchronous job, use the **StartEntitiesDetectionJob** function. To manage your jobs, you can use functions for listing (**ListEntitiesDetectionJobs**) and describing (**DescribeEntitiesDetectionJob**). As soon as the job is complete, the results are output to the S3 bucket that you specified. The example on the slide shows the output, which includes the following:

- **BeginOffset** and **EndOffset** specify the start and end characters of the entity.
- **Score** is the confidence level that Amazon Comprehend has attached to the entity.
- **Text** is the actual entity that is found.
- **Type** indicates the type of entity that was found, which can be PERSON, LOCATION, ORGANIZATION, COMMERCIAL_ITEM, EVENT, DATE, QUANTITY, TITLE, or OTHER.

Training custom entity recognizers



- Amazon Comprehend can implement custom entity recognition.
- You can provide data in two ways:

Annotations

Entity lists

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

24

Amazon Comprehend also has the capability for you to train a custom entity recognizer rather than limiting you to the entity types already available. If you have sufficient data to train effectively, you can identify almost any entity type. This is a form of supervised learning.

You can provide data to Amazon Comprehend in two ways:

- **Annotations** – Provides the location of your entities in a large number of documents so that Amazon Comprehend can train on both the entity and its context
- **Entity lists** – Provides only a limited context and uses only the specific entities list so that Amazon Comprehend can train to identify the custom entity

Annotations



- Train 25 entity types per model
- Use training data with an annotation file

TRAINING DATA

documents.txt

Ana Carolina Silva is an engineer in the high-tech industry.
John Doe has been an engineer for 14 years.
Jorge Souza is a judge on the Washington Supreme Court.
Our latest new employee, Smith, has been a manager in the industry for 4 years.

ANNOTATIONS

File	Line	Begin Offset	End Offset	Type
documents.txt	0	0	18	ENGINEER
documents.txt	1	0	5	ENGINEER
documents.txt	3	25	30	MANAGER

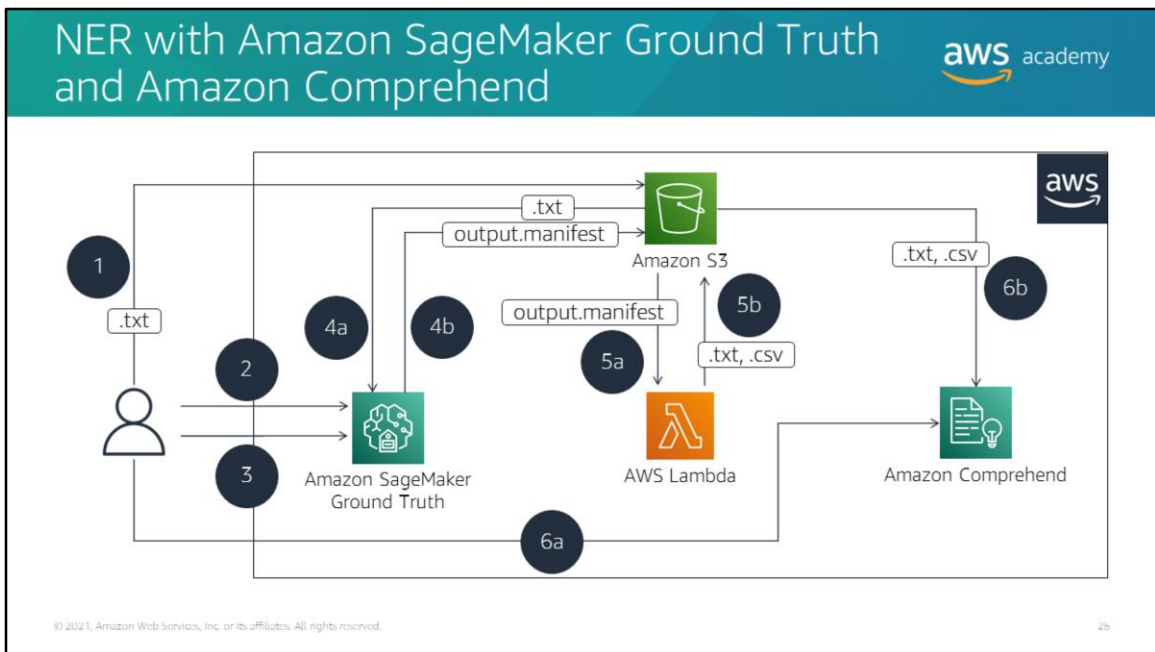
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

25

When you use annotations to train a custom entity recognizer, you must provide annotated training data. Annotations label entities in context by associating your custom entity types with the locations where they occur in your training documents. You can provide this training data as a comma-separated values (.csv) file.

The lines in the first box are an example of a training file:

The next few lines are the annotations for the previous text. Notice that the third line has no annotation because that line has no entity of interest. The line “Jorge is a judge” is not identified because the named entity recognizer is not looking for entities with the type “Judge”.



Building a large annotated training document can be a complex process. You can use Amazon SageMaker Ground Truth to label the dataset that is required to train an Amazon Comprehend custom entity recognizer. Ground Truth provides built-in labeling workflows that take human labelers step by step through tasks. The workflows provide tools to efficiently and accurately build the annotated NER datasets that Amazon Comprehend requires.

The diagram on the slide shows an end-to-end process:

1. Upload a set of text files to Amazon S3.
2. Create a private work team and an NER labeling job in Ground Truth.
3. The private work team labels all of the text documents.
4. On completion, Ground Truth creates an augmented manifest, named "manifest," in Amazon S3.
5. An AWS Lambda function then parses the output manifest to create the annotations and documents in .csv format, which Amazon Comprehend requires.
6. On the Amazon Comprehend console, launch a custom NER training job by using the dataset that was generated in step 4.

For more information about how to use Ground Truth with Amazon Comprehend, see [Developing NER Models with Amazon SageMaker Ground Truth and Amazon Comprehend](#).

Entity lists



- Train 25 entity types per model
- Use training data with an entity list

TRAINING DATA

documents.txt

Josef Brown is an engineer in the high-tech industry.
J Doe has been an engineer for 14 years.
Emilio Johnson is a judge on the Washington Supreme Court.
Our latest new employee, Smith, has been a manager in the industry for 4 years.

ENTITY LIST

Text	Type
Jo Brown	ENGINEER
John Doe	ENGINEER
Jane Smith	MANAGER

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

2 /

When you use an entity list for custom entity recognition, you must supply a .csv file with only two columns:

- **Text** – The text of an entry example as it is written in the accompanying document corpus.
- **Type** – The customer-defined type. An entity type must be an uppercase, underscore-separated string such as MANAGER or SENIOR_MANAGER. You can train up to 25 entity types per model.

The following lines are an example of a training file:

Josef Brown is an engineer in the high-tech industry.
J Doe has been an engineer for 14 years.
Emilio Johnson is a judge on the Washington Supreme Court.
Our latest new employee, Smith, has been a manager in the industry for 4 years.

The next few lines provide the entity list for the previous text. Notice that Emilio Johnson is

not present because that data does not contain the ENGINEER or MANAGER entity.

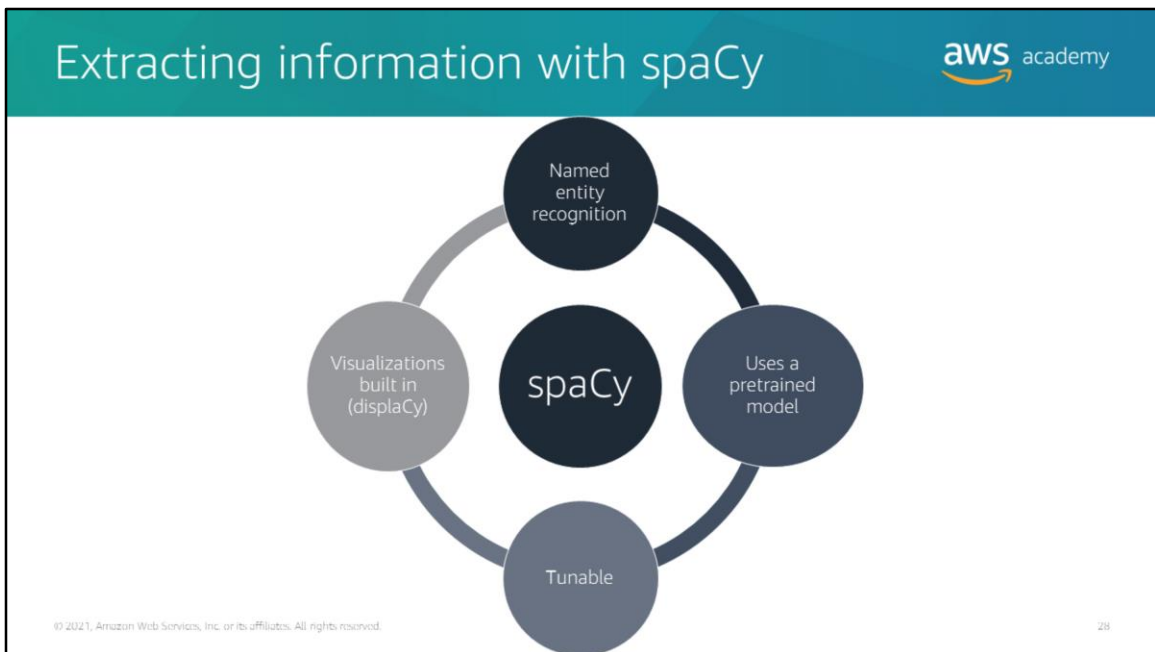
Text, Type

Jo Brown, ENGINEER

John Doe, ENGINEER

Jane Smith, MANAGER


A minimum of 200 entity matches are needed per entity in the entity list to train a model for custom entity recognition.



spaCy is a Python library that provides many tools for NLP. It includes a statistical entity recognition system, which assigns labels to contiguous spans of tokens. The default trained pipelines can identify various named and numeric entities, including companies, locations, organizations, and products.

Identifying entities is only as good as the data that is used to train the model. You might need to use your own labeled data to train the model, especially for domain-specific terms. To update an existing model, you need a few hundred to a few thousand examples. For a new category, you need a few thousand to hundreds of thousands of examples. It can be time-consuming to create these training sets, and the process requires human annotators.

spaCy example




```
import spacy

nlp = spacy.load("en_core_web_sm")
doc = nlp("AnyCompany 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_)
```

Text	Start	End	Label	Description
AnyCompany	0	10	ORG	Companies, agencies, institutions
U.K.	32	36	GPE	Geopolitical entities, such as countries, cities, and states
\$1 billion	49	59	MONEY	Monetary values, including unit

displaCy output:



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

29

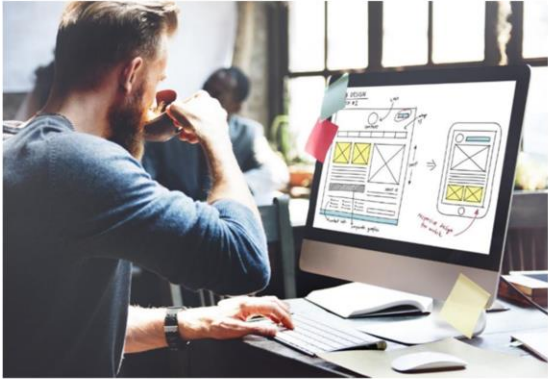
In the example on the slide, the code first loads a pretrained model, named **en_core_web_sm**, and initializes the **nlp** object with the text. This action returns a **doc** object. You can iterate through the **ents** property of the **doc** object to display the entities.

spaCy also contains a visualizer that is named **displaCy**, which is useful for learning and troubleshooting.

Module 5

Guided Lab 1: Implementing Information Extraction

50



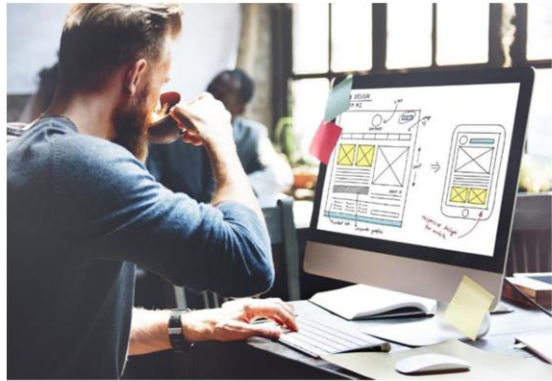
aws academy

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

You will now complete Module 5 Guided Lab 1: Implementing Information Extraction.

Module 5 Guided Lab 2: Working with Entities

51



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

You will now complete Module 5 Guided Lab 2: Working with Entities.

Section 3: Summary



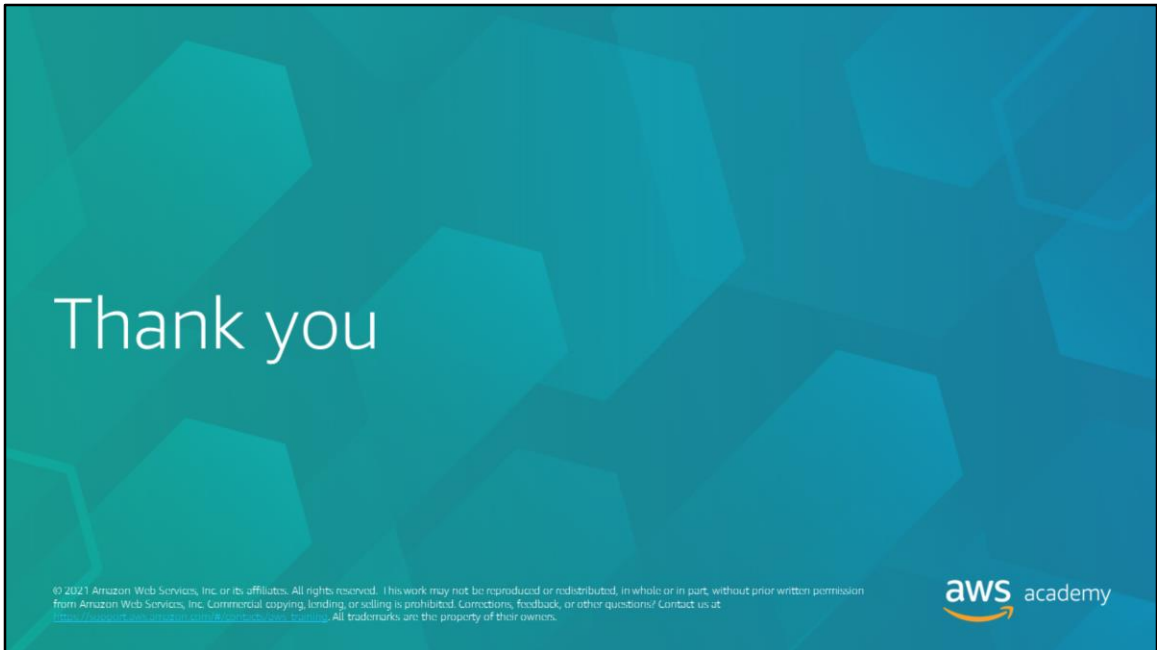
- Amazon Comprehend
 - Pretrained models
 - Custom models
- spaCy
 - Entity recognition system
 - Assigns labels to contiguous spans of tokens

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

52

In this section, you learned about the IE capabilities that Amazon Comprehend provides, which you can use with little machine learning knowledge. You also looked at how you can extend the pretrained models by using your own annotations or entity lists.

spaCy is one of many third-party packages that provide functions for IE tasks. spaCy includes a statistical entity recognition system, which assigns labels to contiguous spans of tokens.



Thank you for completing this module.