UiPath Document Understanding with Named Entity Recognition (NER)

Introduction

In today's world, organizations process many different documents daily. Document processing has gained a lot of attention in the automation world as it is a very monotonous task that requires a lot of time and effort from people. The documents come in different layouts: structured, semi-structured, and unstructured. We can easily apply templates to extract data from structured documents as they follow the same structure in all the papers. Today's technology enables us to use various machine learning models to extract information from semi-structured documents such as invoices, purchase orders, etc. However, extracting information from unstructured documents is a little tricky as the document does not follow a format. In this article, let's look at how the UiPath Named Entity Recognition model enables the users to extract entity information from unstructured documents.

Requirements

Before getting started, you'll need the following

- Basic understanding of the UiPath Document Understanding Framework
- Overview of UiPath AI Center and its usage
- Already available (newly created cloud platform trial, or enterprise)

Building the Scenario

Let’s consider a simple scenario where we need to extract certain information from legal contracts. As we know, legal contracts are very unstructured and do not follow a particular format. Hence, the information available in one legal document may not be available in another legal document, or it maybe available in a different format. In our scenario here, we are mainly looking at information such as company name, employee name, vendee name, and document effective date. Following are a few sample legal documents that we plan to use for this use case.

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header

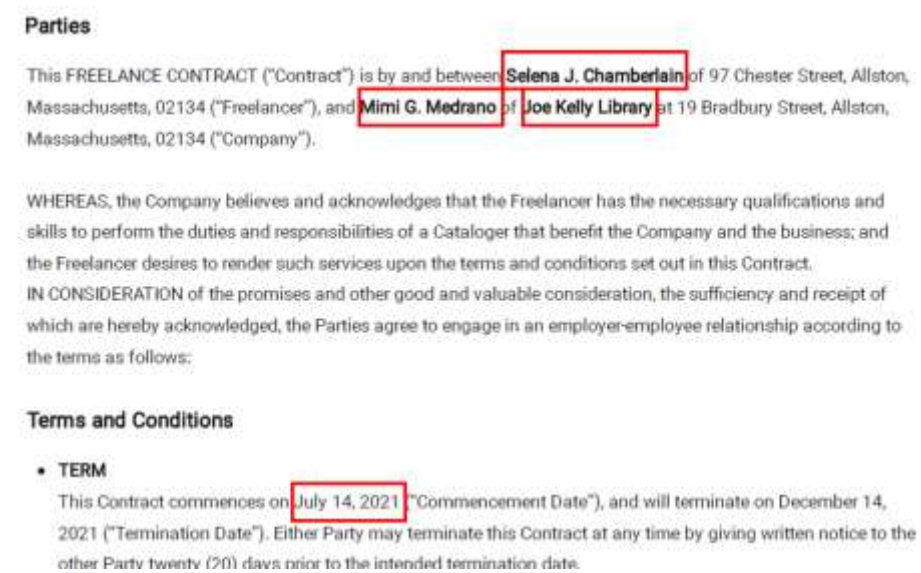
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

Copy-of-Vendor-Contract-Template.pdf

Copy-of-Free-Simple-Freelance-Contract-Template.pdf

Copy-of-Construction-Contract-Templa

Following are two screenshots of the two documents highlighting the values we need to extract.



Parties

This FREELANCE CONTRACT ("Contract") is by and between **Selena J. Chamberlain** of 97 Chester Street, Allston, Massachusetts, 02134 ("Freelancer"), and **Mimi G. Medrano** of **Joe Kelly Library** at 19 Bradbury Street, Allston, Massachusetts, 02134 ("Company").

WHEREAS, the Company believes and acknowledges that the Freelancer has the necessary qualifications and skills to perform the duties and responsibilities of a Cataloger that benefit the Company and the business; and the Freelancer desires to render such services upon the terms and conditions set out in this Contract.
IN CONSIDERATION of the promises and other good and valuable consideration, the sufficiency and receipt of which are hereby acknowledged, the Parties agree to engage in an employer-employee relationship according to the terms as follows:

Terms and Conditions

- **TERM**
  This Contract commences on July 14, 2021 ("Commencement Date"), and will terminate on December 14, 2021 ("Termination Date"). Either Party may terminate this Contract at any time by giving written notice to the other Party twenty (20) days prior to the intended termination date.

Sample-contract-1

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Header

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Footer

Empire Appliance Center

VENDOR CONTRACT

Parties

This VENDOR CONTRACT ("Contract") by and between Jean O. Spruce of Empire Appliance Center ("Vendor") located at 3206 Clinton Street, Portland, Pennsylvania 97205, and Richie W. Anderson ("Vendee") shall commence on August 05, 2027 ("Effective Date").

In consideration of the mutual promises and covenants in this Contract, of which the receipt and sufficiency are hereby acknowledged, the Parties agree to the terms below:

Scope of Engagement

The Vendor hereby agrees to supply the following home appliances ("Goods") at the Vendee's residence, located at 2425 Bird Street, Portland, Pennsylvania 97205, according to the provisions provided in this Vendor Contract:

Sample-contract-2

Let's now have a look at what Named Entity Recognition is, and how it helps us to extract the information we need from these documents.

Named Entity Recognition

Named Entity Recognition is a process of identifying and extracting information units such as names, of people, organizations, numeric values, date and time information, geolocations etc. The models used for NER is capable of identifying such information from a given string and categorizing it according to its type. The UiPath AI Center offers a pretrained NER model and a Custom NER model under the out-of-the-box packages that we can plug and play to extract the information we are looking in this scenario. We can use the pretrained NER model for our use case.

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header

Commented [GC1]: There are two NER models - pretrained NER model, and Custom NER model.

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

Ner-package

However, if you wish you identify your own customized information, you can easily create your own NER model through UiPath. However, this is a topic for another article. The model provided by UiPath is capable of extracting the following information from a given string.

- PERSON – names of people
- GEO – geo locations
- DATE – time and date
- MONEY – currency information in a string (figures and amount in text)
- ORG – Organization names
- GPO – Geopolitical information and many more..

As we now have an idea about NER, and the capabilities available in UiPath, let's now get back to our scenario, and start building a simple flow.

Step 1: Creating NER Skill in UiPath AI Center

First, let, try to get the ML skill up and running so that we can build the workflow around it. Let's follow the below steps to get the model created in AI Center once you login to your cloud environment.

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Header

Commented [GC2]: This screenshot might not be the latest. There is one model missing, i.e., Light Text Classification.

Commented [LF3R2]: I don't see that option still in my Trial environment. Maybe it is not yet in public version?

Commented [OC4]: Can you please add more visual things in the first party (creating the Skill)? Readers may be a bit lost if they are beginners with AI Center.

Commented [LF5R4]: Done. Hope this is helpful

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

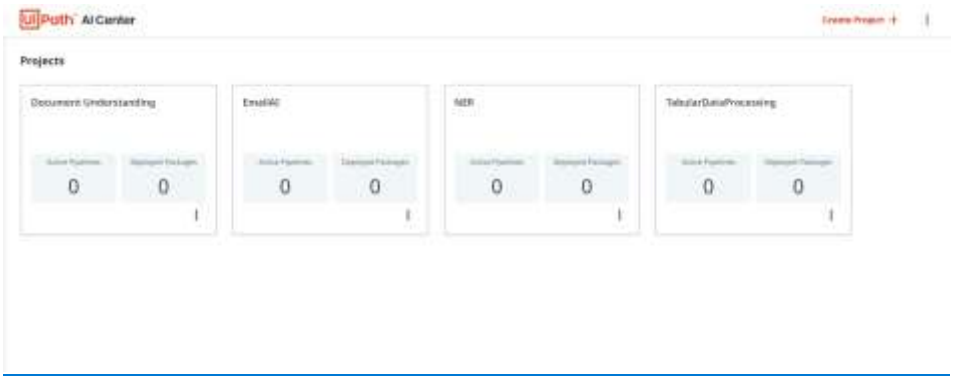
Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Footer

Step 1.1: Creating an AI Center Project

Once you sign in and navigate into the AI Center, you will land oni~~n~~ the AI Center home page which displays all the projects you have created. Let's create a new project by clicking on ~~**New Project**~~**Create Project buton on the top right** and giving it a meaningful name.

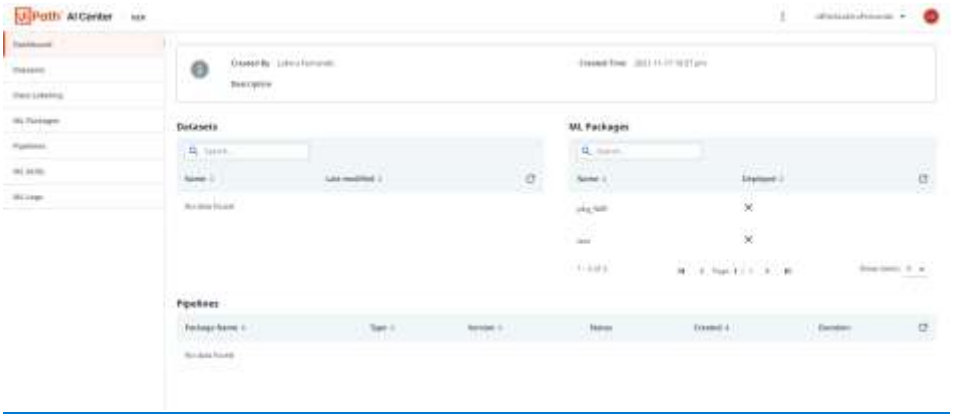


Create-ai-project

Once the project is created, let's navigate into the project so we can start configuring it.

Step 1.2: Creating NER package

The left side panel of the AI Center contains all the stages a ML model goes through.



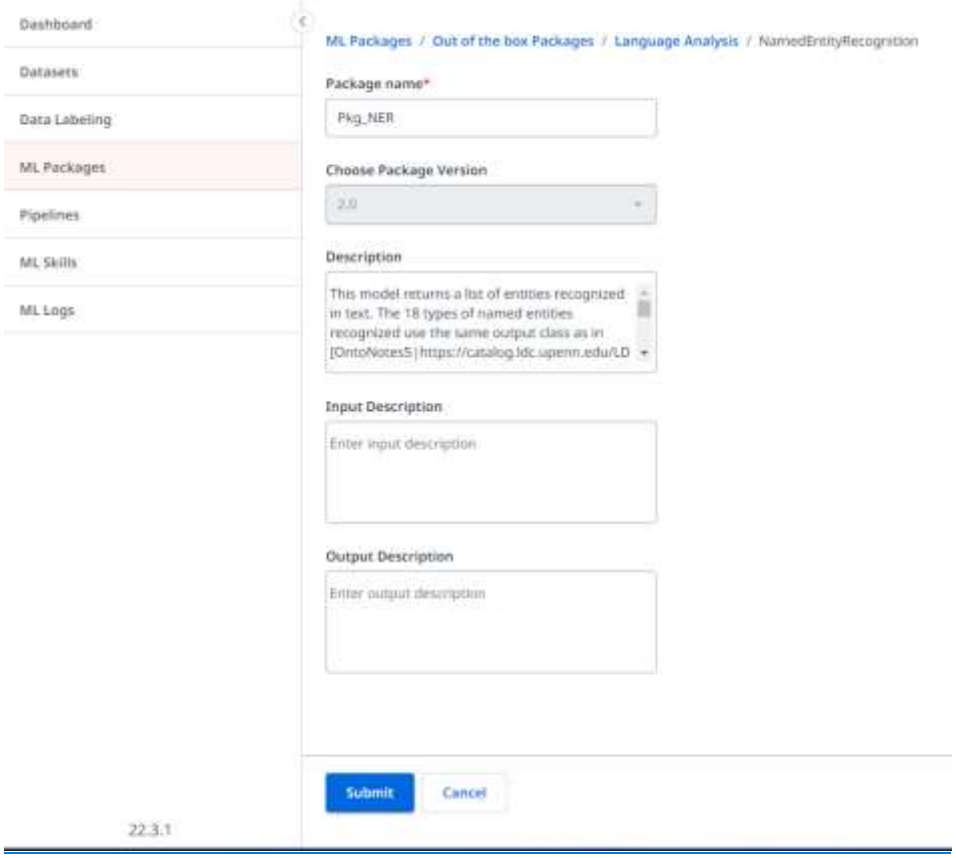
Ai-project-dashboard

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

Since we are using a pretrained model that do not require additional training, we can skip the datasets, data labeling parts. Let's get into the **ML packages** section and navigate into the **Out-of-the-box packages** section.

Navigate into the **Language Analysis** category to locate the NER model. Follow the below steps to create the package.

- Select the Named Entity Recognition model and view the details page.
- Click on the blue color ~~Create~~ Submit button to start creating the package
- Provide a meaningful name such as "Pkg_NER" and click on the Create button to create the package.



Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Font: Bold
Formatted: Font: Bold
Formatted: Font: Bold
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

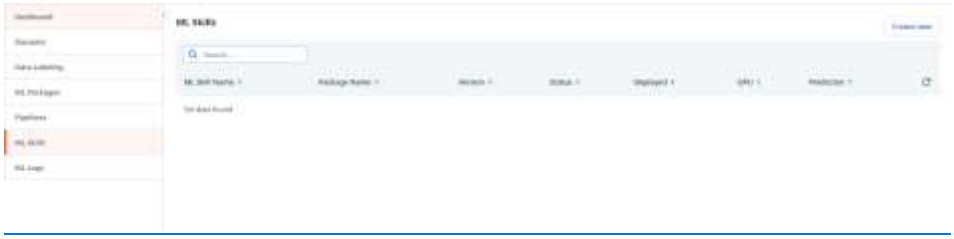
- ner-package-creation

Once you complete the above steps, you should see the created package under the ML Packages section in AI Center.

Step 1.3: Creating ML Skill

Navigate into the ML Skills section in the AI Center and follow the below steps to create the NER skill.

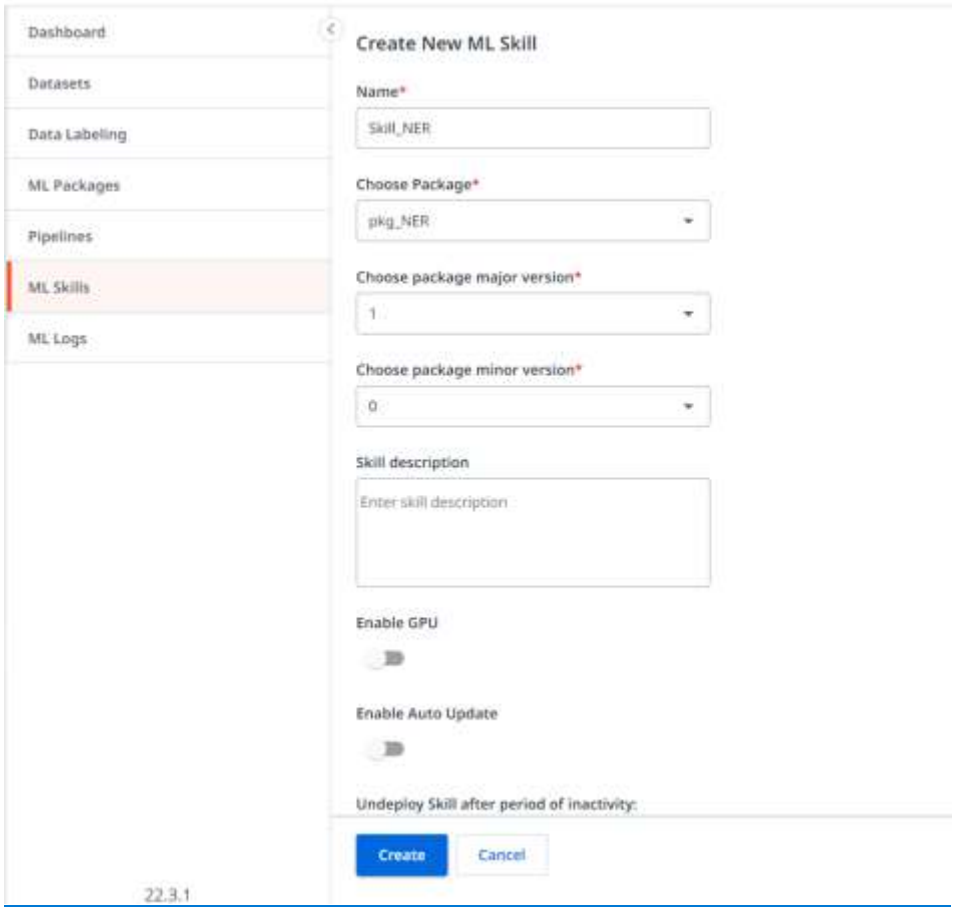
- Click on the **Create New** option to navigate into Skill creation page.



- ml-skill-screen

- Provide a meaningful skill name such as “Skill_NER” for our scenario
- Select the NER package we created from the dropdown menu under ML package
- Select the major version option with the latest version available in the package.
- Select the minor version with the oldest version (ideally zero)
- Click on the Create button to start deploying the skill

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Normal, No bullets or numbering
Formatted: Font: Bold
Formatted: Normal, No bullets or numbering
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer



- create-ml-skill

Once you complete the above steps, you will see the model being deployed in the ML Skills section. It would take a while to complete and wait until the status changes to "Available." You will need to manually refresh the page to see the status change.

Step 2: Preparing the Document Understanding Solution

Let's now have a look at how to create the workflow in Studio. Create a new simple process in UiPath Studio and add he following dependencies.

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Normal, No bullets or numbering
Commented [JT6]: You will need to manually refresh the grid or the page to see the change.
Commented [LF7R6]: Done
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

- UiPath.DocumentUnderstanding.ML.Activities
- UiPath.IntelligentOCR.Activities
- UiPath.OCR.Activities

In addition to the above dependencies, we need two additional dependencies to use the NER model. Add the following dependencies..

- UiPath.MLServices.Activities
- UiPath.WebAPI.Activities

Du-dependencies

As we now have the solution prepared, let's start building our workflow.

Step 3: Building the Workflow

We have one important thing to remember here. The Document Understanding flow can generate the validated data for us through multiple extractors such as ML Extractor, Form Extractor etc. All such extracted data are usually saved in an excel file with the column names defined in the Taxonomy. We can use the same excel file to store the NER data and perform the data manipulations later. NER is used for unstructured data. Hence, it require some level of logic building after data is extracted.

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Header

Commented [GC8]: In step 3, you used Document Understanding. Note, NER model can be used outside of Document Understanding.

Commented [LF9R8]: Yes.. The goal here is to showcase how NER can be used along with DU and combine the data into DU extracted data

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

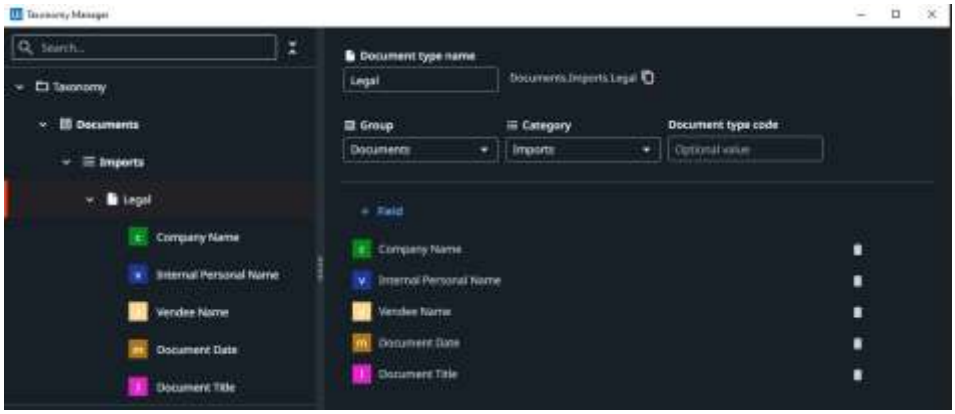
Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Footer

Step 3.1: Create Taxonomy

Irrespective of whether the document is structured or unstructured, create the taxonomy with the fileds we need. In our case, we can use the following fields.

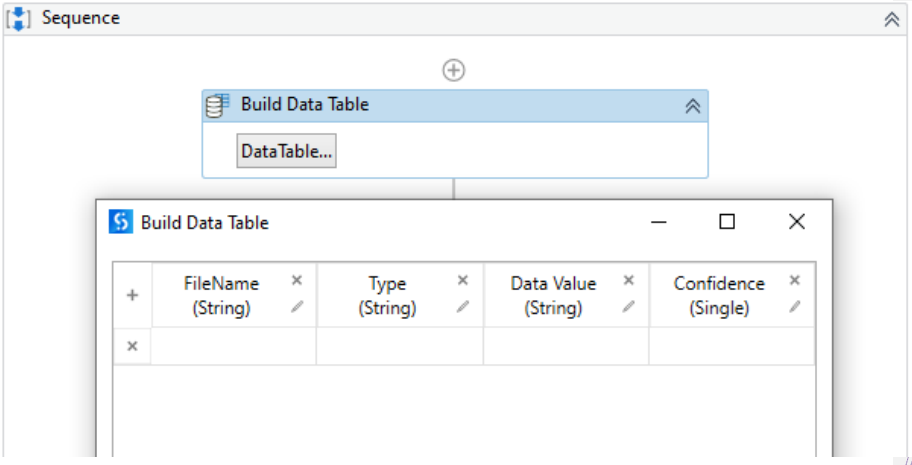


Taxonomy-building

Step 3.2: Building the workflow

Let's follow the below steps to create our workflow.

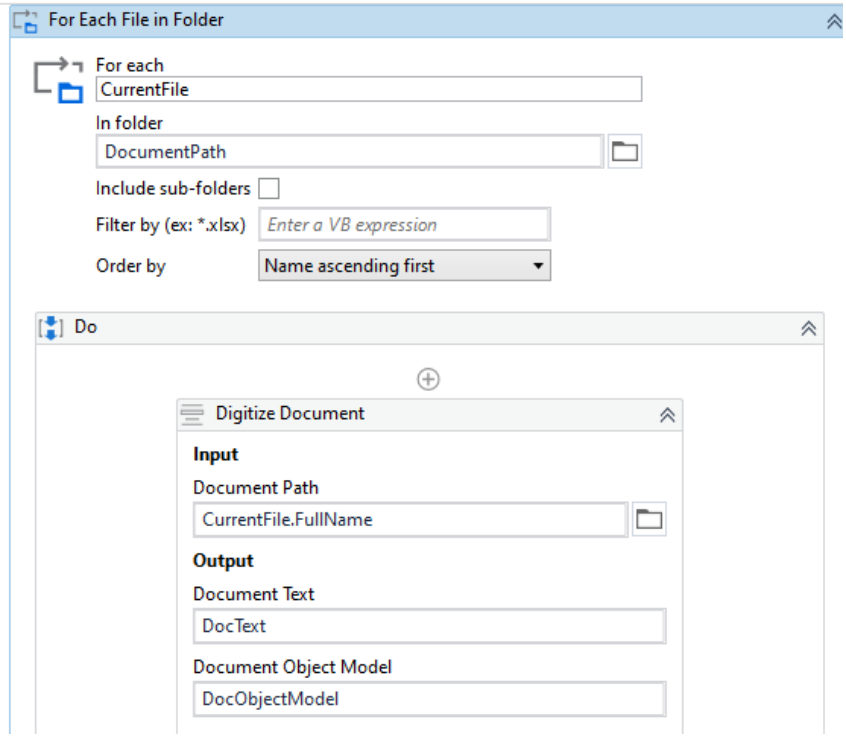
1. Use a Build Data Table activity to store the NER data as shown in the following screenshot



Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

Build-data-table

2. Drag and drop the Load Taxonomy acitivity and configure it to get the Taxnomy
3. Let's include all the documents we want to process in a folder named "Input Files"
4. Now, let's use For Each File activity to loop through the files in the folder as shown in the following screenshot

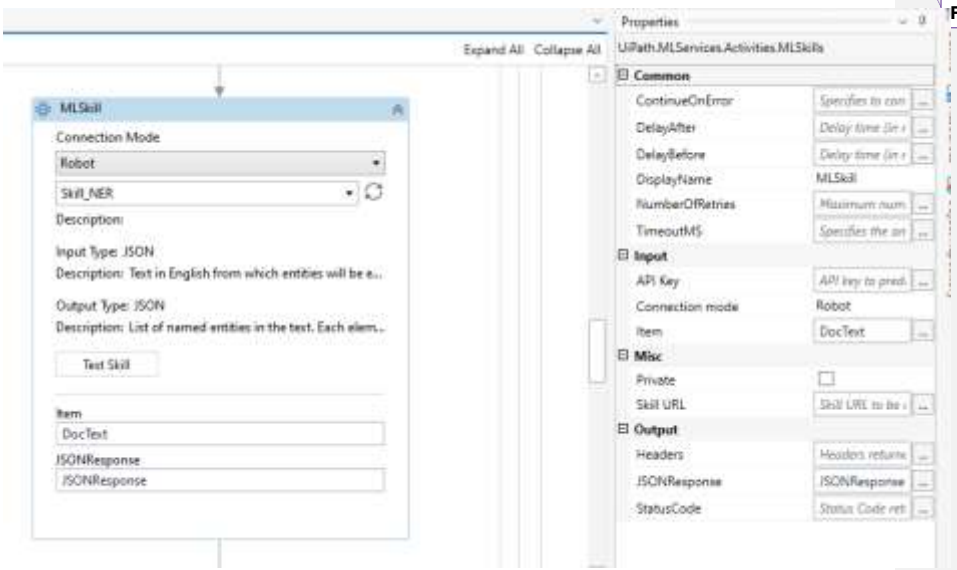


For-each-file-loop

5. Use a Digitize Document activity to convert the document into a digital format as shown in the above screenshot
6. Drag and drop the ML Skill activity and configure it to connect with the ML Skill in the AI Center. You can either connect by selecting Connection Mode as Robot or Endpoint (You need to make the model public to use the endpoint). Let's use Robot connection method and select the ML Skill by clicking on the refresh button and selecting the model from the dropdown option.

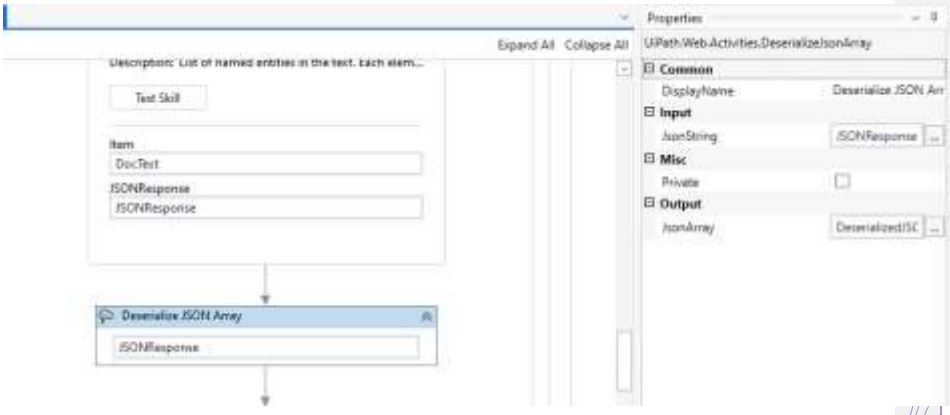
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

7. The model expects string as input and provides a JSON array string as the output. Configure the activity as follows.



Configure-ml-skill

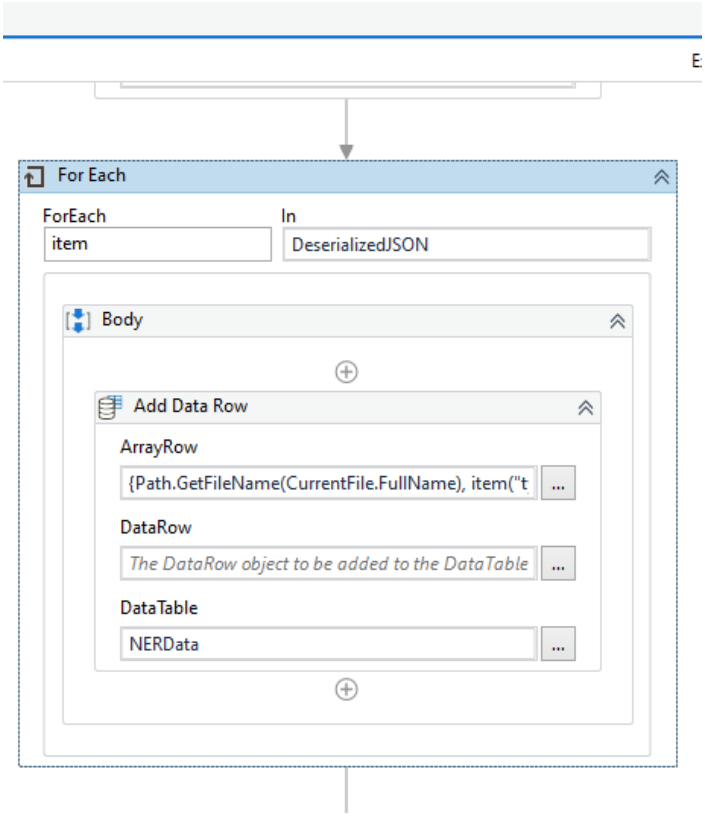
8. Once we get the output, we need to deserialize the response to gain access to the array structure and its elements. In order to deserialize, let's use the Deserialize JSON Array activity and configure it as shown in the following screenshot.



Deserialize-json-array

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

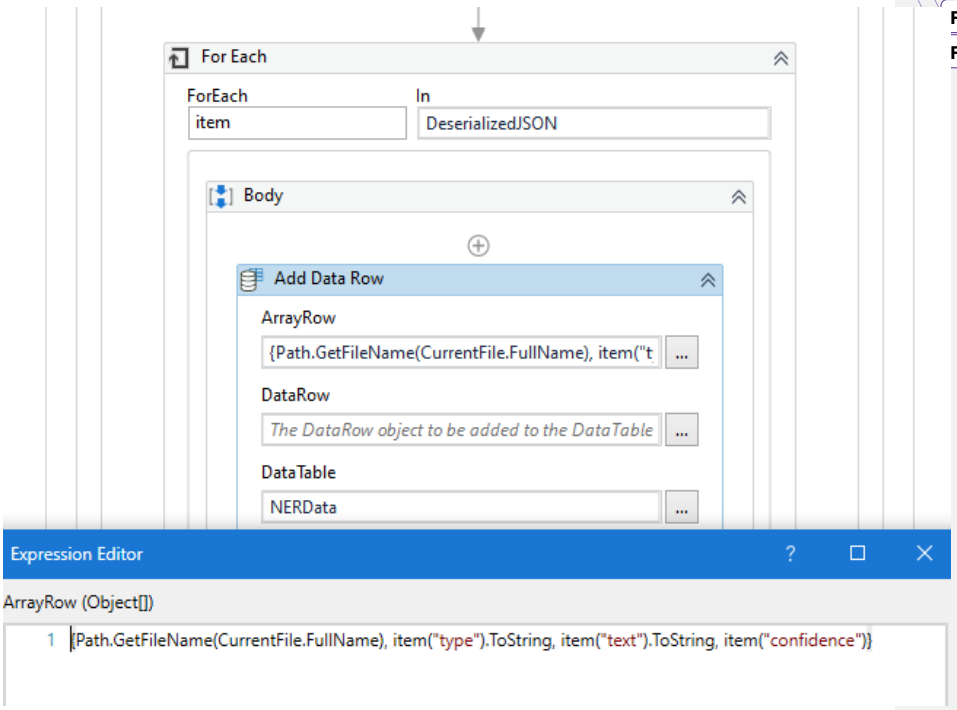
9. Once we deserialize, we can easily loop through the items in the array. Now, let's use a For Each activity to loop through the deserialized Json Array. The type argument of the For Each activity is Newtonsoft.Json.Linq.JObject
10. Place an Add Data Row activity within the For Each activity as shown in the following screenshot



11. Co<u>nfigure</u>~~nfigire~~ the Add Data Row activity to get the information we need as follows. The "Type" provides to which entity a specific value belong to, and the "Text" column provides the identified

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

valuae along with confidence specified in “Confidence” column.



12. Now you can use a Write Range activity to write the NER data available in the data table to the same excel sheet used to store data extracted by DU extraction methods. As a best practice, always use a separate sheet in the same excel file for the intial NER data export. The data written

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header
Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

to the excel file would look similar to the following.

FileName	Type	Data Value	Confidence
Copy-of-Vendor-Contract-Template.pdf	PERSON	Jean O. Spruce	0.905654788
Copy-of-Vendor-Contract-Template.pdf	ORG	Empire Appliance Center ("Vendor")	0.782834947
Copy-of-Vendor-Contract-Template.pdf	PERSON	Clinton	0.808507383
Copy-of-Vendor-Contract-Template.pdf	GPE	Pennsylvania	0.9538427
Copy-of-Vendor-Contract-Template.pdf	PERSON	Richie W. Anderson	0.691781223
Copy-of-Vendor-Contract-Template.pdf	DATE	August 05, 2027	0.758688867
Copy-of-Vendor-Contract-Template.pdf	ORG	Parties	0.877339065
Copy-of-Vendor-Contract-Template.pdf	LAW	below:Scope of EngagementThe Vendor	0.810971439
Copy-of-Vendor-Contract-Template.pdf	LOC	Vendee's	0.549844086
Copy-of-Vendor-Contract-Template.pdf	GPE	Pennsylvania	0.97850579
Copy-of-Vendor-Contract-Template.pdf	ORG	Vendor	0.979345679
Copy-of-Vendor-Contract-Template.pdf	PERSON	Vendee	0.587375402
Copy-of-Vendor-Contract-Template.pdf	CARDINAL	three	0.521842122
Copy-of-Vendor-Contract-Template.pdf	ORG	Vendee	0.979953289
Copy-of-Vendor-Contract-Template.pdf	ORG	Vendor	0.995877028
Copy-of-Vendor-Contract-Template.pdf	PERSON	Vendee	0.571525693
Copy-of-Vendor-Contract-Template.pdf	PERSON	Vendor's	0.542256713
Copy-of-Vendor-Contract-Template.pdf	MONEY	Vendee twenty dollars ($20.00)	0.681489468
Copy-of-Vendor-Contract-Template.pdf	NORP	Vendee's	0.568870008
Copy-of-Vendor-Contract-Template.pdf	DATE	August 9, 2027	0.952560484

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Header

Exported-ner-data

NER data provides all the information the model can find through the Document Text. Once the data is extracted, we need to write a common business logic that applies to all the documents to extract only the information we need. This logic is purely written based on the patterns we identify in the extracted data. We can easily filter the information using the type (ex: PERSON, ORG, DATE), and extract the values using the order of values.

Point to remember here is: All values are extracted based on the order you see them in the document from top to bottom. Further, it would be always easy if you use the excel sheet as a working sheet until you prepare the data and update another sheet that contains the taxonomy as shown below.

Formatted: Header, Indent: Left: -0.08"
Formatted: Header, Centered
Formatted: Header, Right, Right: -0.08"
Formatted Table
Formatted: Footer

Updated-taxonomy-fields

Once you have the data in this format, we can easily use this structure to pass the data into downstream applications or other automation processes.

The two videos below also describe the steps in more detail for you to easily follow.

https://youtu.be/sZE83Xd4G8g

https://youtu.be/fbLLV_FBJiE

In addition to the above videos, we also have another two videos that describes about NER use cases and how this model is used.

https://youtu.be/gYEEnotKLqE

https://youtu.be/ShftOuT7C0g

Conclusion

Document processing is not always straight forward due to the nature of the documents. However, with the capabilities that UiPath offers in Document Understanding and AI Center, a lot of complicated processing has become much simpler. The Named Entity Recognition is one of the AI models that enables users to extract data from unstructured documents as described in this article.

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Header

Commented [GC10]: We have two videos on NER use cases. Would be great to link to them so readers can learn how this model in used.

Commented [GC11R10]: https://youtu.be/gYEEnotKLqE

Commented [GC12R10]: https://youtu.be/ShftOuT7C0g

Commented [LF13R10]: Done.. Looks good?

Formatted: Header, Indent: Left: -0.08"

Formatted: Header, Centered

Formatted: Header, Right, Right: -0.08"

Formatted Table

Formatted: Footer