

AI Builder documentation

AI Builder is a Microsoft Power Platform capability you can use to bring the power of Microsoft AI to your organization, without the need for coding or data science skills.

About AI Builder

OVERVIEW

[What is AI Builder?](#)

[AI model types](#)

TRAINING

[Learn AI Builder](#)

VIDEO

[Overview of AI Builder capabilities](#) 

Build AI models

HOW-TO GUIDE

[Build an AI model](#)

[Train your AI model](#)

[Manage AI models](#)

[Work with sample data](#)

Use your AI models

HOW-TO GUIDE

[Use in Power Automate](#)

[Use in Power Apps](#)

[Work with prebuilt models](#)

[Share your AI models](#)

[Distribute your AI models](#)

What's new and planned?

WHAT'S NEW

[Release plans](#)

[Feature availability by region](#)

Manage AI Builder

REFERENCE

[Administer AI Builder](#)

[Licensing](#)

[Allocate capacity](#)

Get support

REFERENCE

[AI Builder community](#)

[Microsoft support](#)

Overview of AI Builder

Article • 01/05/2023

AI Builder is a Microsoft Power Platform capability that provides AI models that are designed to optimize your business processes. AI Builder enables your business to use intelligence to automate processes and glean insights from your data in [Power Apps](#) and [Power Automate](#). With AI Builder, you don't need coding or data science skills to access the power of AI. You can build *custom* models tailored to your needs, or choose a *prebuilt* model that is ready to use for many common business scenarios.

- Explore AI Builder in [Power Automate](#).
- Explore AI Builder in [Power Apps](#).

The screenshot shows the 'AI Models' interface in the Power Platform. On the left is a navigation sidebar with options like Home, Approvals, My flows, Create, Templates, Connectors, Data, Monitor, AI Builder, Explore, Models, Document automation, Process advisor, Solutions, Learn, and Ask a chatbot. The main area has a banner that says 'Unlock the power of generative AI with GPT'. Below it, the heading 'AI Models > Explore' is followed by a filter bar with tabs for All, Documents, Text, Structured data, and Images. There are eight cards displayed in a grid:

Model Type	Description
Invoice processing	Extract information from invoices
Text recognition	Extract all the text in photos and PDF documents (OCR)
Receipt processing	Extract information from receipts
Identity document reader	Extract information from identity documents
Business card reader	Extract information from business cards
Document processing	Extract custom information from documents
Azure OpenAI Service	Create text, answer questions, summarize documents and more with GPT
Sentiment analysis	Detect positive, negative, or neutral sentiment in text data

Add intelligence to your business

Integration with Power Apps and Power Automate makes using AI easy.

To add intelligence to your business:

1. **Choose an AI model type:** Use the model type that suits your business need. Choose from a growing set of AI solutions.
2. **Connect data:** Select your business-specific data from the available options.

3. **Tailor your AI model:** Depending on the type of model, you can tweak custom models to optimize how your AI performs.
4. **Train your AI model:** Training is an automatic process. It teaches your AI model how to resolve your business problem (for example, how to recognize your products on an image) based on your business data and tailoring. When trained, your AI model can generate insights such as the result of a prediction, or the list and number of objects detected in an image.
5. **Use insights from your AI model:** Use the results from your AI model across Power Platform to create solutions that meet your business needs, even if you have no coding skills. For example, you can create a flow that automates document processing in Power Automate or an app in Power Apps that predicts whether a supplier will be out of compliance.

Learn to use AI Builder

Get started with AI Builder using the following learning resources:

- [AI Builder learning paths and modules](#)
- [AI Builder community forums](#) ↗
- [AI Builder hands-on labs](#) ↗
- [Work with sample data](#)
- [AI Builder licensing](#)

Release status

ⓘ Important

- Some features in AI Builder haven't been released yet for general availability (GA) and remain in preview status. For more information, go to [Release status](#) later in this topic.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- Administrators can control preview feature availability for their environment by using the Power Platform admin center. More information: [Enable or disable AI Builder preview features](#)

Because of technical dependencies, some features are released differently in various locations. For a breakdown of the release status of AI Builder features and model types for your region, go to [Feature availability by region](#).

For information about license capacity, pricing, and restrictions:

- Find the right Power Apps plan for your business needs [↗](#).
- Power Automate pricing [↗](#).
- AI Builder licensing.

For administrators

- If you're an administrator and want information on model and environments, go to [Administer AI Builder](#).
- For information on security for administrators, go to [Roles and security in AI Builder](#).

Next step

[Learn about AI model types](#)

See also

[Training: Get started with AI Builder \(module\)](#)

AI Builder licensing

Article • 04/18/2023

You might receive this notification when you sign in: **You've consumed all of your AI Builder credits. Contact your administrator to get more capacity.** If you get this message, read this article to learn how to increase capacity and get details and FAQs on licensing.

The screenshot shows a modal window titled "Extract information from invoices". A yellow warning bar at the top contains the text: "You've consumed all of your AI Builder credits. Contact your administrator to get more capacity. Learn more". Below the bar, there is descriptive text about the invoice processing prebuilt AI model. At the bottom right of the modal, there are two buttons: "Prebuilt model" and "Premium".

Prerequisite to set up an environment ready for AI Builder

AI Builder is licensed as an add-on to your Power Apps, Power Automate, or Dynamics 365 license. This means you need a Power Apps, Power Automate, or Dynamics 365 license that allows you to create a Microsoft Power Platform environment.

Get entitlement to AI Builder capacity

First, you need to be entitled to some AI Builder capacity. This can be through one of the following paid capacities.

- Purchase an AI Builder capacity add-on, or
- Purchase other licenses offering some seeded capacity.

This can be through a trial. The capacity is then linked to a user. Trials are available only when there's no paid capacity within the tenant.

Then, a tenant administrator has to allocate part or all of the purchased capacity to the environment where you want to use AI Builder.

It's also possible to leave part or all of the purchased capacity unallocated, at the tenant level. In this case, all environments without allocated capacity can consume these tenant-level unallocated credits.

Check the consumption level of an environment

When browsing AI Builder pages in Power Apps or Power Automate portal, you might get this notification: **You've consumed all of your AI Builder credits. Contact your administrator to get more capacity.** This means the monthly consumption is higher than the allocated capacity.

To get details on your environment allocation, check environment details in [Administer AI Builder, Manage capacity](#).

To get details on your environment consumption, check the [AI Builder consumption report](#).

When you add all the consumptions of the current month of a specific environment, you'll get the monthly consumption of this environment.

If your environment has no more capacity, you need to provide capacity to your environment. To do this, reallocate existing capacity (from the tenant or environment level). Alternatively, you can purchase more capacity and make it available to your environment.

To help estimate the required add-on capacity based on your estimated consumption, use the [AI Builder calculator](#).

Purchase AI Builder capacity

Some Microsoft products like Power Apps per app plan, Power Apps per user plan, and Power Automate per user plan with attended RPA include some AI Builder capacity. Your environment admin can check entitlement in Power Platform admin center by following the instructions in [Capacity add-ons](#). When this amount isn't enough, you need to complete it with one or more AI Builder capacity add-ons.

To purchase the AI Builder credit add-on in the Microsoft 365 admin center:

1. In the admin center, select **Billing > Purchase services**. You need to be the billing administrator of your tenant to access this page.
2. On the **Purchase services** page, do the following steps:
 - a. Search for **AI Builder**.
 - b. Select **Details of the AI Builder Capacity add-on** tile
 - c. Follow the purchase process.

Allocate capacity

When a tenant has purchased AI Builder capacity, corresponding credits are by default unallocated and available as a pool on the tenant. In this state, credits can be used on any environment.

The administrator can restrict usage by allocating all credits to specific environments. The administrator can also reserve some capacity to an environment by allocating an amount of credits to this environment.

To learn how to allocate capacity in [Power Platform admin center](#), go to [Allocate or change capacity in an environment](#).

To learn details on how to manage AI Builder capacity, go to [Manage capacity](#).

Capacity Overuse

When an environment consumption exceeds its credit allocation, the following functions will begin to fail:

- Create a model or a new version of a model.
- Run a model in Power Apps or in a Power Automate flow.
- Scheduled model run or retrain.
 - Some scenarios allow scheduled run or retrain, which is configured on the model settings panel. These scheduled tasks will fail when you exceed capacity.

Only limited overage is allowed. To continue using AI Builder when you exceed capacity, purchase the AI Builder add-on capacity, and allocate it to your environment.

Trial licenses

To get started using AI Builder, use premium features for a limited time with a [trial license](#). Alternatively, use preview features without obtaining a license at all.

AI Builder is licensed as an add-on to your Power Apps, Power Automate, or Dynamics 365 license. This means you can start your AI Builder trial after you have a Power Apps, Power Automate, or Dynamics 365 license that allows you to create a Microsoft Power Platform environment.

Note

You can't start an AI Builder trial if you have AI Builder credits already on your tenant by purchasing a capacity add-on or through other products.

Learn more about AI Builder licensing in the following articles:

- [Microsoft Power Platform Licensing Guide](#)
- [AI Builder licensing FAQ](#)
- [Estimate the AI Builder capacity that's right for you](#)

Benefits of a trial license

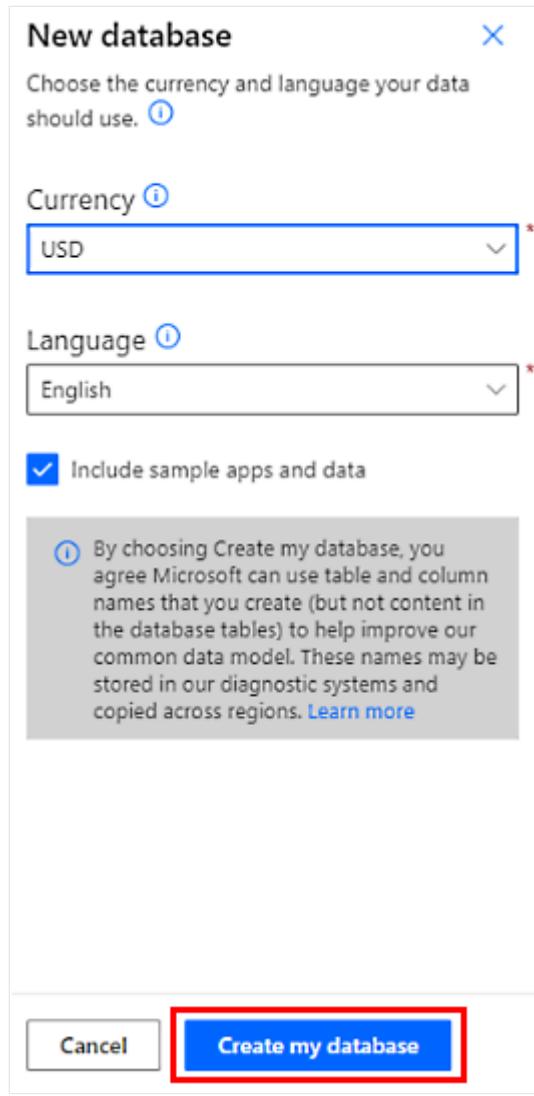
An [AI Builder trial license](#) enables you to use AI Builder features for free during the 30-day trial period.

What you get an AI Builder trial license:

- Create and use AI models in any environment (trial or production).
- Store your AI model results in Dataverse.
- Use AI model in your apps, flows, and more.

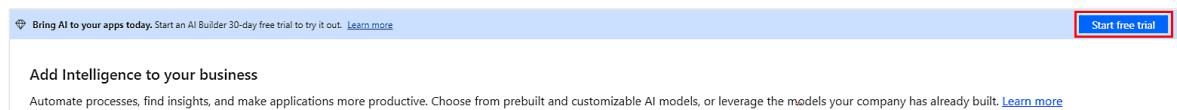
Activate an AI Builder trial license

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. Select **AI Builder > Explore**.
3. (If requested) Select **Create a database**, and then choose a currency and language.
4. Select **Create my database**.



5. Refresh the **Explore** page.

6. At the top of the screen, select **Start free trial**.



(!) Note

Your AI Builder trial license is applicable at the user level, not the environment level. You can use your trial license on multiple environments. Another user would have to start their own trial or paid license to use your models in any environment.

After the trial expires

Your AI Builder trial license expires after 30 days.

To continue using AI Builder, purchase AI Builder add-on capacity and allocate AI Builder capacity to your environments. You may also be able to extend the trial period. Trials can be extended a limited number of times. Extension can only occur after the trial expires.

To renew your AI Builder trial license, sign in to Power Apps or Power Automate, select **AI Builder > Explore**, and then select **Extend trial** in the banner on the top.

Trial capacity

An AI Builder trial license includes a limited amount of AI Builder capacity. You use this capacity when running or training models.

If you exceed your AI Builder capacity, you'll receive an over-capacity notification. These notifications appear as banners in AI Builder pages or when using the model.

The following functions aren't available when you exceed capacity:

- Create a model or a new version of a model.
- Run a model within Power Apps or in a Power Automate flow.
 - You can still share your model with another user. They can run it using their own AI Builder capacity if they have a valid trial.
- Scheduled model run or retrain.
 - Some scenarios allow scheduled run or retrain, which is configured on the model settings panel. These scheduled tasks will fail when you exceed capacity. Therefore, data won't be refreshed, and the model won't be retrained.

To continue using AI Builder when you exceed capacity, [purchase AI Builder add-on](#), and [allocate capacity](#) to your environments.

If you extend a trial after expiration, capacity is reset. You can again run and train your models, and create new ones. Scheduled run and retrain instances will resume according to the existing settings.

Preview features

AI Builder features that are in preview release status are free to use. You don't need to obtain a license to use AI Builder preview features.

AI Builder licenses FAQ

How many AI Builder credits are consumed for each operation?

Each AI Builder capability consumes service credits at a different rate. You can check the consumption rates in the [Microsoft Power Platform Licensing Guide](#) in the AI Builder capacity add-on section.

My Power Automate flow or my Power App fails in AI Builder with an error message that contains one of the following. How do I unblock it?

- EntitlementNotAvailable
- NoCapacity
- QuotaExceeded
- No capacity was found
- Credit usage exceeds allocation

You need to check entitlement allocation of your environment. If there's no allocation, and no allocation at the tenant level, [allocate capacity](#) to the environment or tenant level from other environments or by [purchasing AI Builder add-on](#).

If there are AI Builder credits allocated or available at tenant level, compare the amount with the [AI Builder consumption report](#). Then [allocate additional capacity](#) eventually by [purchasing the AI Builder add-on](#).

Where can I see credit usage for this month?

Get details on your environment consumption by checking the [AI Builder consumption report](#). It gives the amount of consumption per day per user per environment. Adding all the consumptions of the current month of a specific environment will give you the monthly consumption of this environment.

When is credit usage count reset? Is it based on license acquisition date?

Credit usage is counted on a monthly basis, starting the first day of each month. It's not based on the license acquisition date. Credit usage of an environment should not exceed the credit allocation.

What happened to the remaining capacity at the end of the month? Can it be used the following month?

No, unused capacity is not carried over to the next month. Credit usage is counted on a monthly basis and reset on first day of the month. Available capacity is based on allocated capacity.

I exceeded my trial capacity. What can I do?

- You can [purchase AI Builder Add On](#) and [allocate capacity](#) to your environment
- You can wait for your trial to expire, then extend your trial: this will add new capacity.
- You can also share your existing model with another user who still has active trial with capacity.

What happens after my trial license expires?

After your AI Builder trial license expires, or if you exceed capacity:

- Your models are not deleted.
- You have to [purchase AI Builder Add-On](#) to continue using your AI models.
- An administrator must [allocate AI Builder capacity](#) to any environment where you want to use AI Builder.
- You can't create or modify AI Builder models, and no new inference will be possible when the trial expires if you don't purchase a license.
- You can also extend your trial, but only for a limited number of times.

What happens to my data and models when my AI Builder trial expires?

Data and models are deleted only when the environment is deleted.

Where can I find more information about trial environments?

To learn more, go to [About trial environments](#).

Can I block users in my organization from signing up for an AI Builder trial?

Any individual can try out the features of AI Builder for 30 days and incur no costs to your company. This option is available to any user in a tenant and can't be disabled by an admin. Once your company purchases some AI credits (for example, through capacity add-ons or inclusion in some licenses), trials aren't proposed to users anymore.

What can I do with the 5,000 AI credits included in the per user plan with attended RPA?

Each user license grants you 5,000 credits, allowing you to assess the capabilities in AI Builder. For instance, you could use these credits to extract data from a few documents with [document processing](#) or perform hundreds of basic OCR extractions with [text recognition](#).

What happens if a user turns on/off AI Builder per user capacity add-on license?

Some user licenses include seeded AI Builder credits. For example, the Power Apps per user license includes 500 seeded AI Builder credits. If **AI Builder capacity per user add-on** isn't selected, the 500 AI Builder credits are still added to the total number of AI Builder credits owned at your tenant level and can be used.

Where can I learn more about license management in Power Apps and Power Automate?

Learn more about licenses and license management in Power Apps in [About licensing and license management](#).

See also

- [Microsoft Power Apps and Power Automate Licensing Guide](#) ↗
- [AI Builder licensing FAQ](#)
- [Estimate the AI Builder capacity that's right for you](#) ↗
- [AI Builder consumption report](#)
- [Training: Get started with AI Builder licensing \(module\)](#)

AI Builder labs

Article • 05/31/2022

Use AI Builder labs to gain experience with AI Builder and learn more about it. Links are added here as new resources become available.

AI Builder [hands-on labs](#) walk through these features:

- Object detection
- Prediction
- Category classification
- Document processing
- Business card reader

The zip file that contains the hands-on labs includes the following folders and zip files:

- Lab data
- Lab images
- Lab scripts
- AIBuilderLabSolution_1_0_0_0.zip
- ProcessFeedback_Flow.zip

See also

[Work with sample data](#)

Start using AI Builder with sample data

Article • 05/31/2022

Don't have any data of your own to create a model? No problem! We've got you covered.

Sample data is available for several AI Builder model types, together with instructions for working with the sample data. Select one of these options to get started:

- [Use sample data to do prediction](#)
- [Use sample data to do category classification](#)
- [Use sample data to do entity extraction](#)
- [Use sample data to do object detection](#)
- [Use sample data to do document processing](#)

Preview features

Article • 07/08/2022

AI Builder is released and is generally available. But, some of its features are still in preview status. These features display a **Preview** tag near their names:

- On the AI Builder **Explore** page.
- In the Power Apps Studio > **Insert** > **AI Builder** menu.

Block preview scenarios

The environment administrator can decide to hide accesses to preview features. In the Microsoft Power Platform admin center, under **Settings** > **Features**, a toggle switch lets the admin specify whether preview features are displayed.

FAQ

What happens if a model linked to a scenario in preview mode is created and published, then the admin decides to hide all access to scenarios in preview?

In that case, this model will be blocked; it can't be edited or retrained, and it can't be used to compute new predictions. It will be displayed in the list of models, but it will be disabled and the only action that will be allowed is Delete.

AI Builder community

Article • 05/05/2022

The forums are a great resource where anyone can read and discuss topics with other AI Builder users.

Join the discussion

Read and post in the [AI Builder forum](#). Before you post a new question, search the discussion forum to see whether your question already has been answered.

See also

- [Power Apps community forums](#)
- [Power Automate community forums](#)
- [Microsoft Dataverse community forums](#)

AI models and business scenarios

Article • 03/06/2023

In AI Builder, you can choose from several model types that are suited to different business scenarios. Here are some examples:

- If you want to use intelligence to detect your products in images, you can refine a *custom* AI Builder object detection model. With a customizable model, you'll build, train, and publish it for your intended use.
- If you want to use intelligence to automate your expense reports by scanning and processing business receipts, you can use a *prebuilt* AI Builder receipt scanning model. All prebuilt models allow you to go straight to productivity.
- If you want to design a marketing campaign based on patterns in your historical data, you can use a *custom* prediction model tailored to your business, using your own historical data.

These models are only a few of the ways you can use AI Builder to add intelligence to your business processes.

To build a model with AI Builder:

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. On the left pane, select **AI Builder > Explore**.
3. On the right pane, select the model type that matches what you want to do. Then, you're ready to get started.

The screenshot shows the Microsoft Power Automate interface. On the left, there's a sidebar with navigation links like Home, Approvals, My flows, Create, Templates, Connectors, Data, Monitor, and AI Builder. The AI Builder section is expanded, showing sub-options: Explore (which is selected), Models, Document automation, Process advisor, Solutions, Learn, and an Ask a chatbot button. The main area is titled 'Unlock the power of generative AI with GPT' and describes using a GPT model on Azure OpenAI Service. Below this is the 'AI Models > Explore' section. It has tabs for All, Documents, Text, Structured data, and Images. There are eight cards displayed:

- Invoice processing**: Extract information from invoices.
- Text recognition**: Extract all the text in photos and PDF documents (OCR).
- Receipt processing**: Extract information from receipts.
- Identity document reader**: Extract information from identity documents.
- Business card reader**: Extract information from business cards.
- Document processing**: Extract custom information from documents. This card includes a 'Custom model' option.
- Azure OpenAI Service**: Create text, answer questions, summarize documents and more with GPT. This card includes a 'Preview' button.
- Sentiment analysis**: Detect positive, negative, or neutral sentiment in text data.

Model types

The following table lists the data type, models type, and build type.

- The *data type* describes the type of AI that the models use (for example, documents, text, structured data, or images).
- The *build type* indicates whether it's a customizable model that you'll need to build, train, and publish for your intended use, or if it's a prebuilt model that's ready to use. In general, use *custom* AI Builder models for applications where you're working with data that's unique to your business. Use *prebuilt* models for scenarios that are common across different types of businesses.

[Common business scenarios](#) and the model types that are suited to them are described later in this topic.

Data Type	Model type	Build type
Documents	Business card reader	Prebuilt
Documents	Document processing	Custom
Documents	Text recognition	Prebuilt
Documents	Receipt processing	Prebuilt
Text	Azure OpenAI Service (preview)	Prebuilt

Data Type	Model type	Build type
Text	Category classification	Prebuilt (preview) and custom
Text	Entity extraction	Prebuilt and custom
Text	Key phrase extraction	Prebuilt
Text	Language detection	Prebuilt
Text	Sentiment analysis	Prebuilt
Text	Text translation	Prebuilt
Structured data	Prediction	Custom
Images	Object detection	Custom

Common business scenarios

The different types of AI models in AI Builder provide you with a broad range of AI capabilities without the need for coding or data expertise. Here are some common business scenarios, and the preferred AI model types for addressing them:

Business scenario	Model type
Automate customer application processing	Document processing
Automate expense reports	Receipt processing
Categorize user feedback based on their focus	Category classification
Extract insights from product reviews	Entity extraction
Identify language of text	Language detection
Identify and classify customer feedback	Sentiment analysis
Translate support requests into your language	Text translation
Identify fraudulent transactions	Prediction
Get alerted to social media posts referencing your brand	Key phrase extraction
Automate contact list	Business card reader
Automate inventory taking	Object detection
Take a photo of text and save it to a database	Text recognition

Next step

[Build a model](#)

See also

[Feature availability by region](#)

Build a model in AI Builder

Article • 12/13/2022

In AI Builder, we guide you through each step to create your AI model.

The screenshot shows the Microsoft Power Platform interface with the 'AI Builder' section selected. At the top, there's a banner with the text 'Unlock the power of generative AI with GPT' and a 'Try the preview' button. Below the banner, the 'AI Models > Explore' page is displayed. It features a navigation bar with tabs for 'All', 'Documents', 'Text', 'Structured data', and 'Images'. The main area contains eight cards representing different AI models:

- Invoice processing**: Extract information from invoices.
- Text recognition**: Extract all the text in photos and PDF documents (OCR).
- Receipt processing**: Extract information from receipts.
- Identity document reader**: Extract information from identity documents.
- Business card reader**: Extract information from business cards.
- Document processing**: Extract custom information from documents. This card includes a 'Custom model' option.
- Azure OpenAI Service**: Create text, answer questions, summarize documents and more with GPT. This card includes a 'Preview' button.
- Sentiment analysis**: Detect positive, negative, or neutral sentiment in text data.

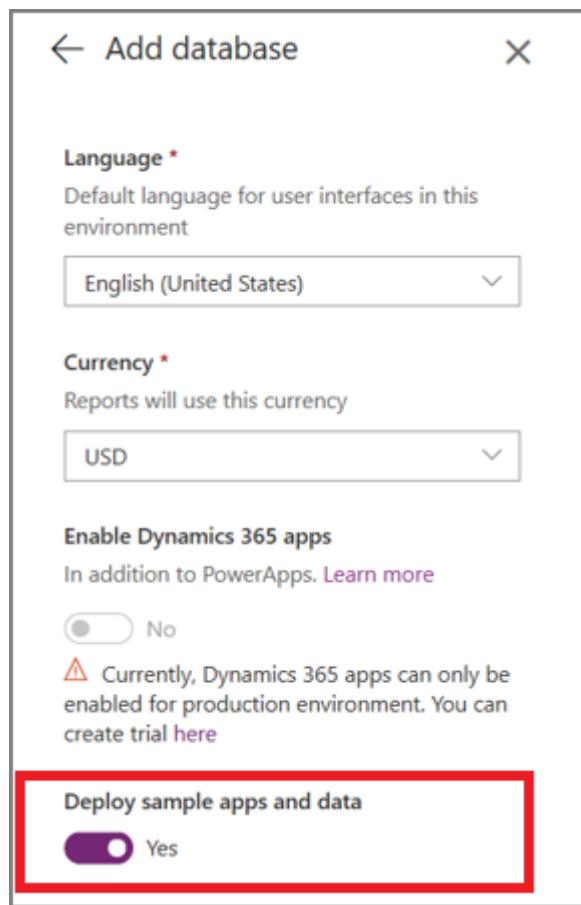
On the left sidebar, under the 'AI Builder' category, there are links for 'Explore', 'Models', 'Document automation', 'Process advisor', 'Solutions', 'Learn', and a 'Ask a chatbot' button.

Prerequisites

- AI Builder requires the use of [Microsoft Dataverse](#), which is the data platform for Microsoft Power Platform that allows you to store and manage business data. Dataverse is the platform on which Dynamics 365 apps are built. This means if you're a Dynamics 365 customer, your data is already in Dataverse.
- AI Builder must be enabled for your environment. Otherwise, you won't have access to AI Builder functionality.

Deploy sample apps and data

Do you want to explore AI Builder by using sample data provided by Microsoft? Enable the **Deploy sample apps and data** setting when you create your environment to add sample data to your environment automatically. You can also [download sample data](#), and then upload it to your environment.



Get started

AI Builder allows you to build models based on data type (for example, documents, text, structured data, or images), and build types. The *custom* build type allows you to build, train, and publish a model for use that is unique to your business. The *prebuilt* build type is ready to use, and offers scenarios that are common across different types of businesses.

For more information about data types and build types, go to [AI models and business scenarios](#).

1. Sign in to [Power Apps](#).
2. In the left pane, select **AI Builder > Explore**.
3. Select a custom model, and then select **Get started**.

Next step

[Train your model in AI Builder](#)

See also

AI Builder actions are disabled/deactivated

Train your model in AI Builder

Article • 04/07/2022

Before you can use your AI model, you have to train it to perform the way you want. After you train your model, [publish it](#) to make it available to other people.

When you create a model in AI Builder, you configure it based on the needs of your business. Each time you save changes to your model in AI Builder, AI Builder saves your progress as a draft. When you're done, confirm the settings with which you want to train your model, and then select **Train** to begin training.

The screenshot shows the 'Model summary' page in the AI Builder interface. On the left, a sidebar lists completed steps: 'Select historical outcome' (Personal Bank Loan > Has Loan Account), 'Select fields' (21 fields selected), and 'Filters'. A 'Model summary' step is shown in pink, indicating it's the current page. The main area displays the following details:

- Model type:** Prediction
- Owner:** [Redacted]
- Data source:** Common Data Service
- Historical outcome:** Personal Bank Loan > Has Loan Account
- Entity Fields:** Personal Bank Loan | 21

On the right, there are three sections: 'Quick tips' (What is training? / What's next?), 'Get help or send feedback' (Get help), and buttons for 'Save and close' and 'Train' (which is highlighted).

ⓘ Note

Training takes time, so you can stay on the page and wait, or you can close the page and come back later.

After you train your model for the first time, you have access to a details page where you can [manage your model](#) and—for some model types—view the model's performance results.

On the details page, training results appear in the **Last trained version** section.

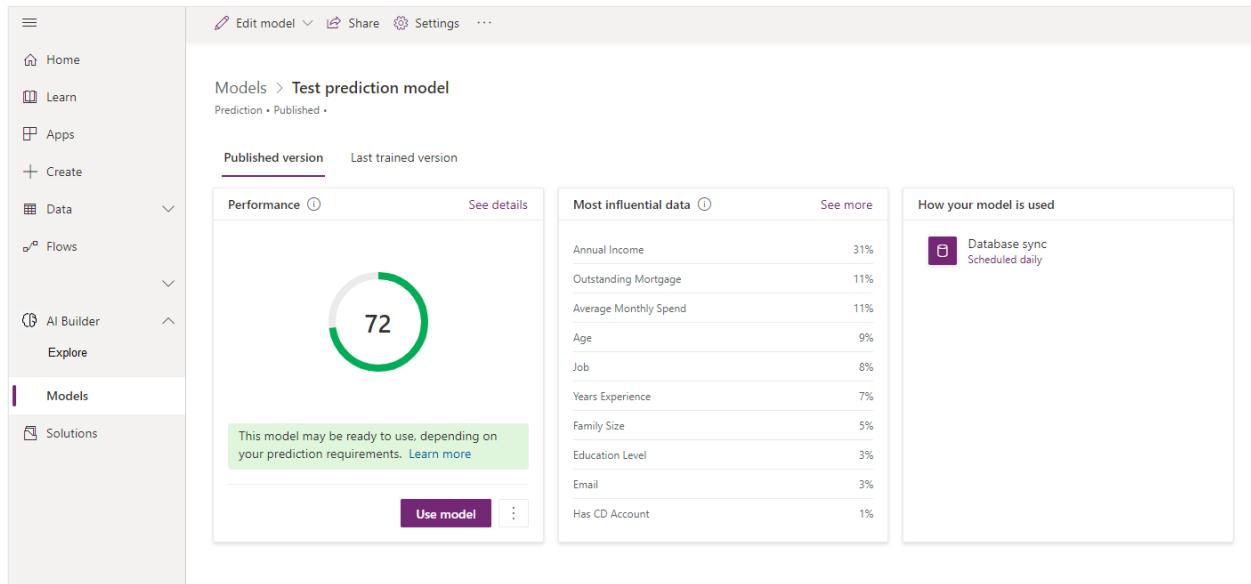
Next steps

- [Manage your model in AI Builder](#)
- [Publish your model in AI Builder](#)

Manage your model in AI Builder

Article • 01/05/2023

Creating the optimal model for your business can be a rather iterative process. Results can vary depending on the configurations you set and the training data you provide. Updating these factors can improve the performance of your model. In some cases, however, performance might be degraded. Each AI model type has a set of guidelines to help you with the process of creating the best model, tailored to your needs.



The screenshot shows the AI Builder interface with the following details:

- Left sidebar:** Home, Learn, Apps, Create, Data, Flows, AI Builder (selected), Explore, Models (selected), Solutions.
- Top navigation:** Edit model, Share, Settings, ...
- Page title:** Models > Test prediction model
- Sub-navigation:** Prediction • Published •
- Published version:** Last trained version
- Performance:** A large green circle with the number 72 in the center. Below it, a tooltip says: "This model may be ready to use, depending on your prediction requirements. [Learn more](#)".
- Most influential data:**

Feature	Percentage
Annual Income	31%
Outstanding Mortgage	11%
Average Monthly Spend	11%
Age	9%
Job	8%
Years Experience	7%
Family Size	5%
Education Level	3%
Email	3%
Has CD Account	1%
- How your model is used:** Database sync (Scheduled daily)

Evaluate your model

After you train your model for the first time, you can evaluate its performance and quality on its details page.

Depending on your AI model type, a performance score might appear for each trained version. You can use this score to quickly compare two versions of the same model. However, remember that the score is based on the configuration for that training. Make sure you take any changes you made between versions into consideration when you compare scores.

Each AI model type has a different explanation for how the score is calculated and how the score should be interpreted. View the tooltip next to **Performance** to learn more.

Some AI model types include a feature to quickly test the performance for your trained version with real data of your choosing. Select **Quick test** to see your model in action.

After you finish evaluating your newly trained model, you have two options:

- **Publish your model:** For more information about when to publish a model, see [When should I publish my model?.](#)
- **Create a new version:** For more information about when to create a new version, see [When should I create a new version?.](#)

Underfit models

An *underfit* model is a model that actually performs worse than a random guess. If your model consistently performs poorly, it's probably an indication that there's a problem with your training data. Are the fields you're using relevant to the type of determination that your model is intended to make? Are there data input errors or other problems that are leading your model astray?

Overfit models

An *overfit* model appears to perform very well—if not perfectly—when run on your training data. That can be because there's a column in your training data that directly corresponds to outcome. For example, let's say you have a prediction model that predicts whether a shipment will arrive on time. If your historical data includes the actual delivery date, your model would predict perfectly when run against your historical data. It probably wouldn't do so well when run on real data in your business environment, because the delivery date column wouldn't be populated yet.

Edit the model name

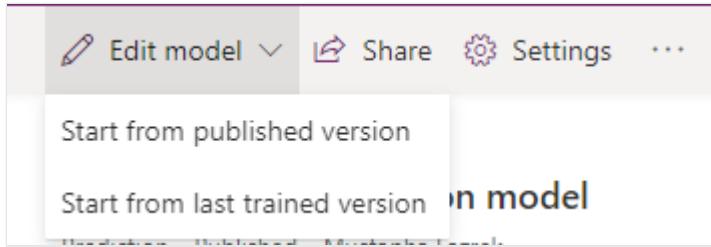
1. At the top of the page, select **Settings**.
2. In the **Model settings** panes on the right, under **Name**, enter a different name.
Depending on your AI model type, you might need to first select the **General** section.
3. Select **Save**.

Create a new version

To create a new version, select **Edit model** at the top of the page.

You can have up to two trained versions available at a time: one **Published version** and one **Last trained version** that isn't published. If you train a new version when a last trained version already exists, the existing last trained version is overwritten.

When you create a new version, your model is based on the configuration from an existing version—your published version, or your last trained version. If you have both, you have to choose which one you want to create the new version from.



A new version is created only after you've successfully trained it. If you leave without finishing your changes and training your model, your progress is saved as a draft. Certain actions, such as creating a new version or retraining, might be disabled until you train or discard your draft. You can only have one draft available at a time, so you have to select either **Resume draft** to pick up where you left off or **Discard draft** to get rid of the changes before you can continue.

After training, your training results appear under the **Last trained version** section of the **Details** page.

If you're satisfied with your last trained version, you can publish your model to make it available. Otherwise, you can always create a new version.

When should I create a new version?

You can create a new version of your model to help improve the model performance or quality. This depends on the AI model type: some models can be improved by updating the configuration, and some models can be improved by updating the training data.

Due to the experimental nature of machine learning, not all new versions you create will result in an increase in model performance. If you aren't satisfied with your model, you can create a new version to try to yield better results.

If you're satisfied with your model, you can [publish it](#) to make it available. Because you can only have two trained versions available at a time, you might want to publish a model that you don't want to be overwritten by a new version.

For more information about the nuances of improving your model performance, see the message underneath your accuracy score.

Retrain and republish existing models

Whereas training creates a new version by updating your configuration, retraining creates a new version that uses the same configuration as your current version. The benefit of retraining is that it will study any new data so that your model stays accurate over time. This action is only applicable to certain AI model types.

1. Sign in to [Power Apps](#).
2. On the left pane, select **AI Builder > Models**.
3. Follow the steps for your model type.

For prediction and category classification models, in the **Performance** section, select the (...) menu, and then select **Retrain now**.

4. This replaces your last trained version. If you're ready, publish this version.

Perform these steps on each of your AI Builder models to get your AI models up and running again.

Next step

[Publish your model in AI Builder](#)

See also

[Training: Manage models in AI Builder \(module\)](#)

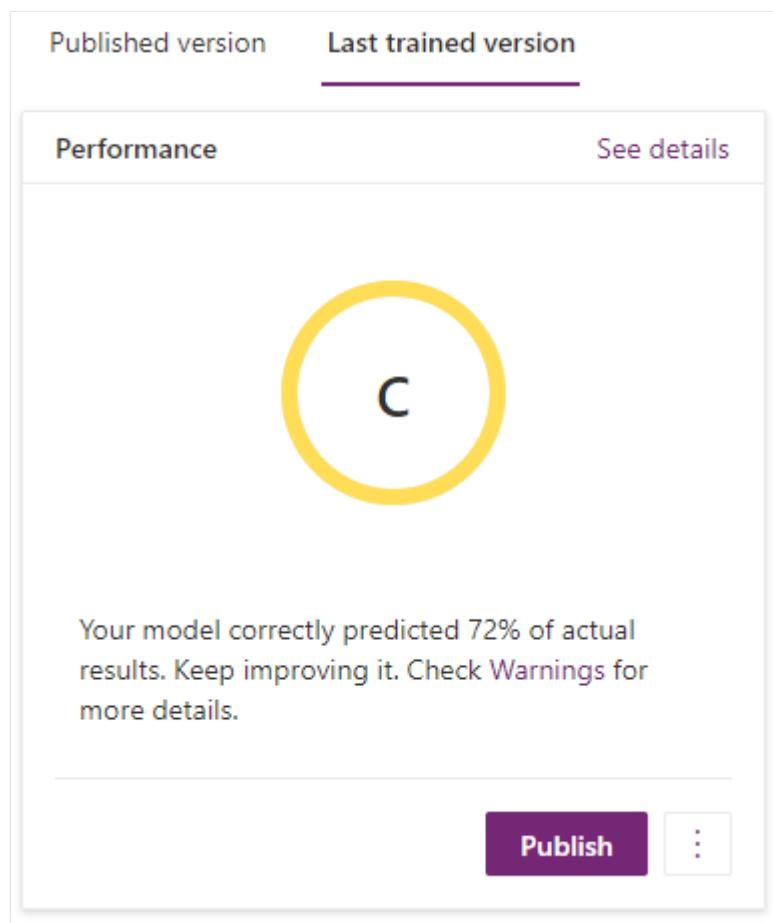
Publish your model in AI Builder

Article • 04/07/2022

After you successfully train your model, you have to publish it to make it available. All users in your current environment will be able to use your published model when you publish it.

Publish your model

On the details page, under **Last trained version**, select **Publish**.



After you publish your last trained version, it appears as the published version. For certain AI model types, you might need to take additional steps to use your model in Power Apps or Microsoft Dataverse.

ⓘ Note

- Any previous published version is overwritten when you publish a new version.

- If you have a published version and a last trained version, you'll lose the published version when you unpublish because the last trained version is more recent.

When should I publish my model?

Publish your model when you want to make it available to users in your Power Apps environment. If you aren't satisfied with your model, you can create a new version to try to yield better results. For information about how to create a new version, see [Manage your model in AI Builder](#).

If you're satisfied with your model, you can publish it to make it available. Because you can only have up to two trained versions available at a time, you might publish a version because you don't want it to be overwritten by a new version.

Next step

[Use your AI model](#)

See also

[AI Builder in Power Automate overview](#)

[AI Builder in Power Apps overview](#)

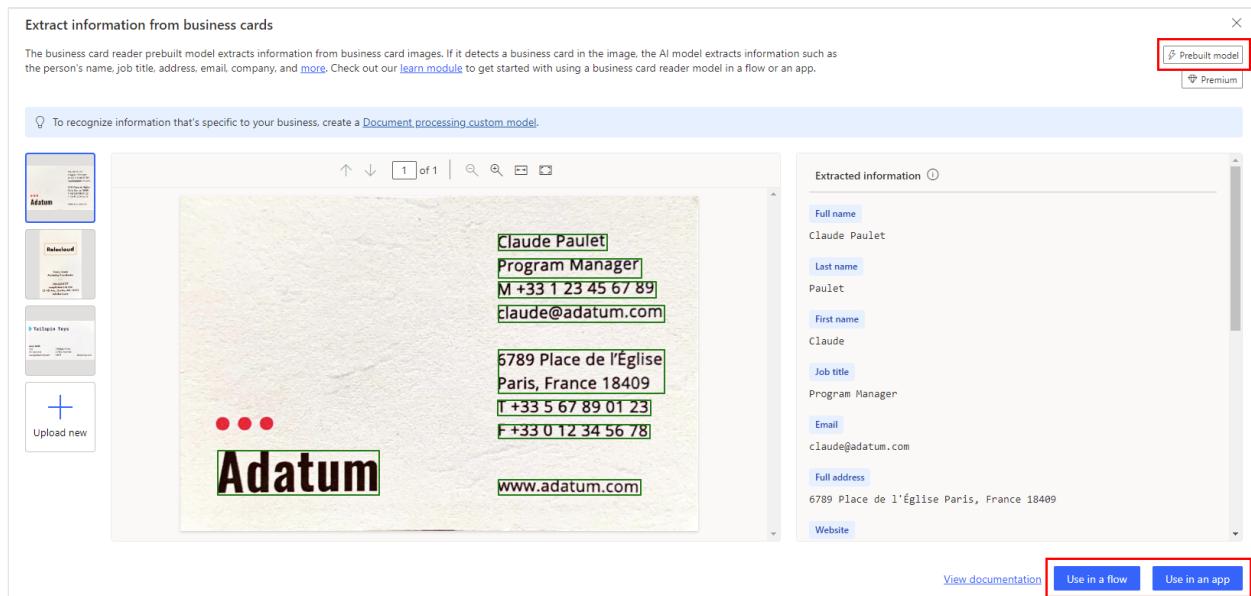
Use your AI model

Article • 07/08/2022

After you publish your model, you can use it across Microsoft Power Platform to create end-to-end solutions that meet your business needs.

Get straight to productivity

When you select a *prebuilt model* from one of the tiles in **Try out AI capabilities for each data type**, you can choose how you want to use the model (in a flow or app), and then AI Builder will provide you with the steps to build it. A prebuilt model tile doesn't have the word **Custom** in the upper-right corner. When you select it, the next screen shows **Prebuilt model** in the upper right. The following example shows the screen for the business card reader model.



Optimize a model for your business outcome

When you select a *custom model* from one of the tiles in **Try out AI capabilities for each data type**, you can customize your own steps. You can learn how your model is used in the detail page.

How to use your model



Power Automate
[See documentation](#)



Power Apps
[See documentation](#)

To learn how to use each AI Builder model in Power Automate or Power Apps, select one of the following links for details:

- [Use AI Builder in Power Automate](#)
- [Use AI Builder in Power Apps](#)

Next step

[Share your AI model](#)

Share your AI model

Article • 04/19/2023

When you create and publish AI models in AI Builder, they're private—only you can run them. This allows you to test them and use them within apps or flows.

If you want other users to use your model in Power Apps or Power Automate, you need to share it. This applies to users who create apps or flows that call your model, and also to users who only run those apps or flows. Both app makers and app users need shared access to your model.

You can also share a model with other users as co-owners of this model. Co-owners can edit, retrain, publish, or share a model. However, co-owners can't delete a model.

Share action

The share action is available on the **Models** page in AI Builder for each model where you're the owner. It's also available for the system administrator in the environment, or for any security role that has Share permissions on AI Builder system tables.

The share action is also available on the model detail screen in AI Builder, with the same conditions.

Share panel

When you select the share action, a share panel appears. The share panel is where you select Microsoft Dataverse users and teams in your organization, and share your model with them.

- Sharing your model with a user or a team as a **User** gives the user or team the ability to see and run your model. However, this user or team won't be able to edit, retrain, publish, share, or delete the model. They also won't be able to access the data you used to train your model.
- Sharing your model with a user or a team as a **Co-owner** gives the user or team the ability to see, run, edit, retrain, publish, or share the model. This user or team will be able to access the data you used to train your model. They won't be able to delete the model.

Model list views

The models you create and the models that are shared with you appear in the model list under two different views:

- My models
- Shared with me

 **Note**

If you're an administrator of the environment, all models in your environment appear in the **Shared with me** view, whether they've been shared or not.

When a model is shared with you as a user, you can use it in Power Apps or Power Automate, but you can't view details or edit the model. No actions are available for models in the **Shared with me** list.

As an admin or owner of a model, you might encounter a model where the only available action is **Delete**. This happens when the model type is no longer supported. It might be that the model type was a preview feature, and the admin [disabled AI Builder preview features](#).

FAQ

Is the sharing step necessary to use an app that includes calls to an AI model? Why is app sharing not enough?

App sharing only allows users to open the app. Things like data access or AI model access are granted separately. AI model access is given through the sharing mechanism. Without it, users can open the app, but the app can't execute a call to the AI model.

Is the sharing step necessary to allow other environment makers to use my model in their apps?

Yes. Your model isn't listed in the AI model control if it isn't shared with the maker of the app. This helps you control access to your work and decide when to release it.

Can I allow other users to edit my model?

Yes. When sharing a model, look for the user or team to share the model with and select the option **Co-owner**.

Users with the **System Customizer** role can edit models from other users by default. Modifying models with **System Customizer** role isn't supported by AI Builder and you might face unexpected issues.

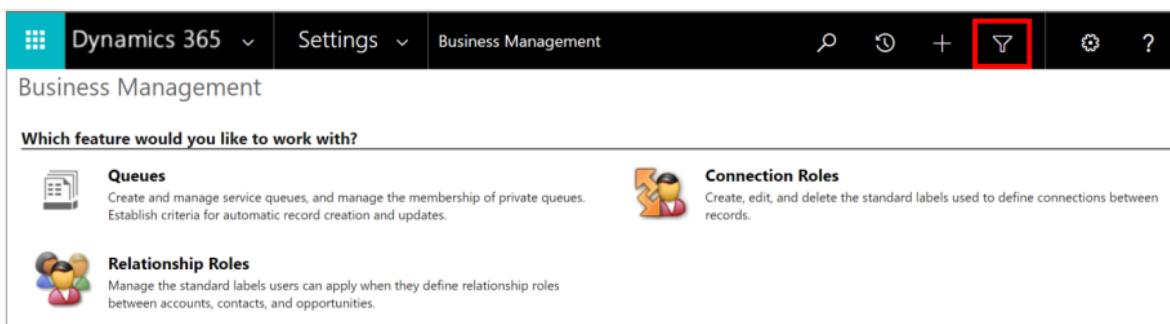
The owner of a model has left the company. How can we allow non-admin users to edit this model?

A user with the **System Customizer** role can perform this action. To learn more about roles, go to [Roles and security](#).

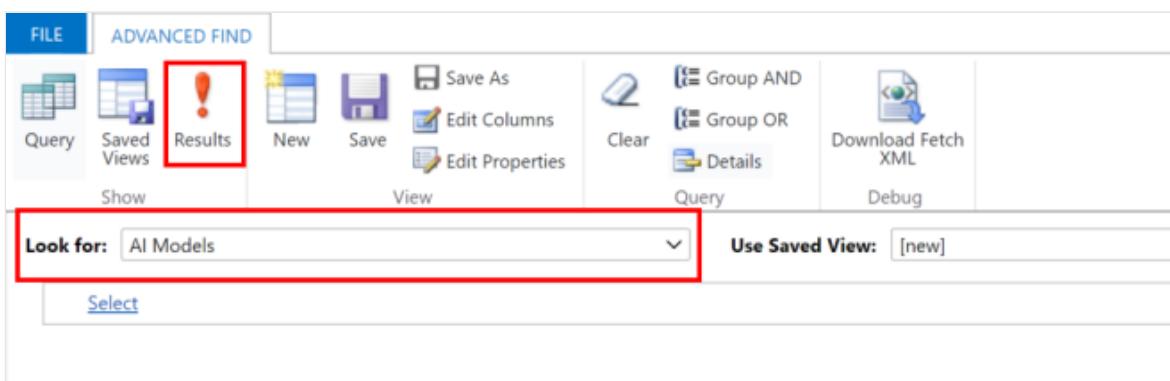
Reassign this model to another user. *Reassign* means changing ownership in the advanced settings of your organization in AI Builder. You also need to give access to the data used to train the model.

To change the ownership of a model:

1. Sign in to the [Power Platform admin center](#).
2. Select the environment where the model you want to change ownership is located.
3. Select **Settings > Resources > All legacy settings**.
4. Select **Advanced Find** on the top right corner.



5. From the **Look for** list, select **AI Models**, and then select **Results**.



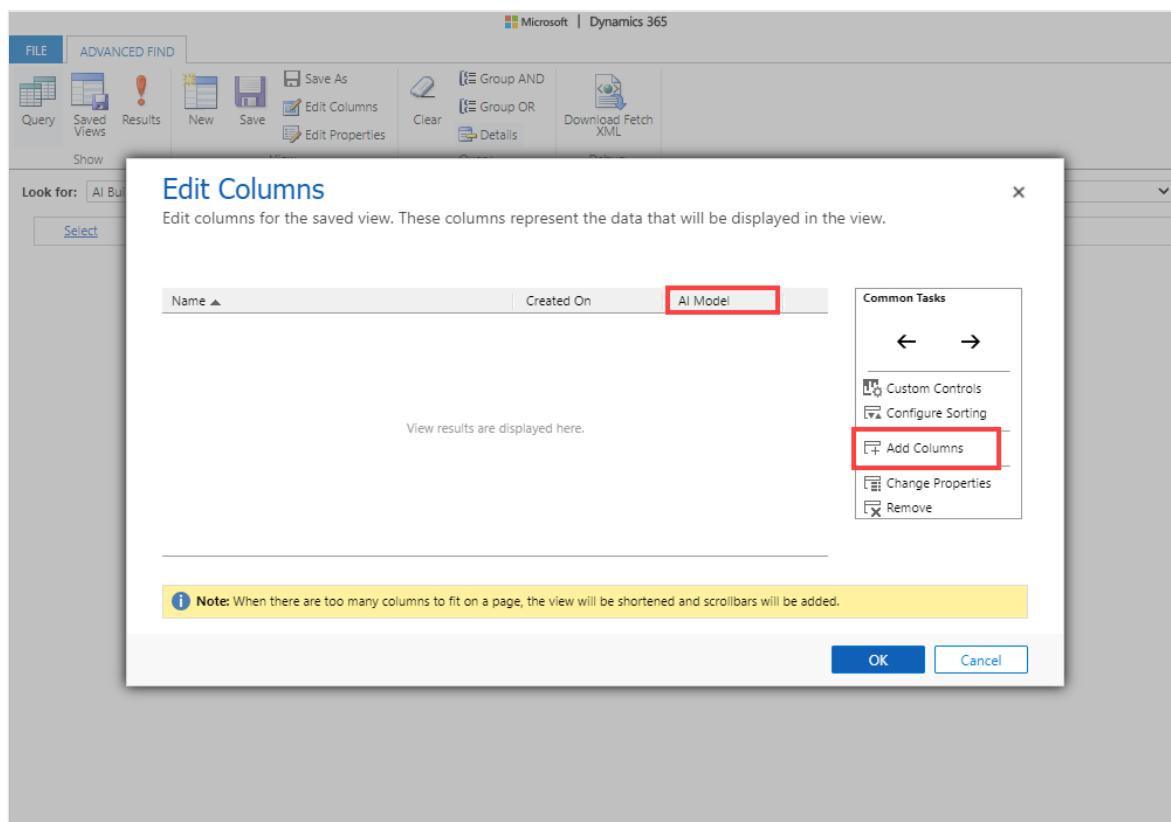
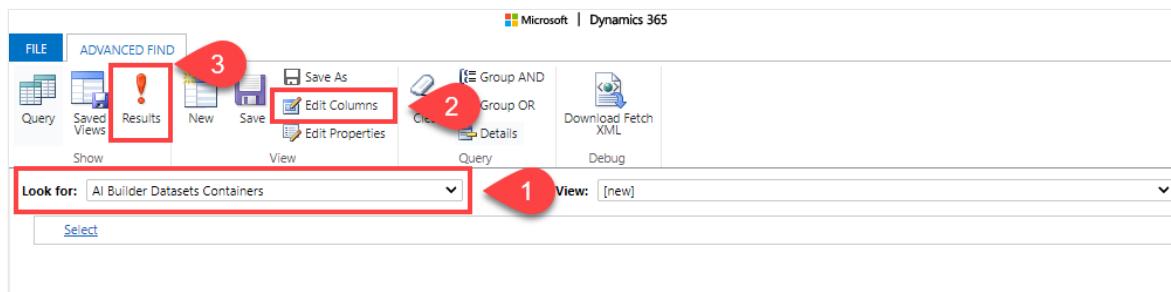
6. You'll get a list of the AI models in the environment. Select the model you want to reassign, and select **Assign AI Models**. A window will open where you can change

the owner of the model.

If the model you're changing the ownership is a *document processing* model or an *object detection* model, you'll also need to reassign the **AI Builder Datasets Container** associated with the model.

To reassign AI Builder Datasets Containers:

1. On the **Advanced Find** tab, select **AI Builder Datasets Containers** from the **Look for** list. To make it easier to identify which AI models it corresponds to, select **Edit Columns > Add Columns > AI Model**.



2. Select the row associated to the model you're reassigning.
3. Select **Assign AI Builder Datasets Containers**. A window will open where you can change the owner.

Can I disable the sharing feature for AI model makers and only allow admins to do it?

Yes. In advanced settings of your organization, an admin must create a security role where the share privilege is disabled for the **AI Model** custom entity. Assign this role to AI model makers.

Why can't I share generic prediction models?

Generic prediction models only work as part of a scheduled run. They can't be used in Power Apps or Power Automate, so the share action isn't available.

Next step

[Distribute your AI model](#)

See also

[Training: Manage models in AI Builder \(module\)](#)

Distribute your AI model

Article • 12/13/2022

You can distribute an AI model as a *solution component*. After you create a model in AI Builder, make it available for other environments to use. Do this by packaging it into a solution, and then exporting it into a zip file. After the solution is imported in the target environment, the packaged AI model is available for use.

For more information, go to [Introduction to solutions](#).

Solution explorer

Use the Power Apps solution explorer to create solutions and add components—such as AI models—to them. You can also export and import solutions by using the solution explorer.

For more information, go to [Use solutions in Power Apps](#).

Recommended process

It's a good idea for you to develop AI models in a sandbox or development environment first. Then, you can deploy them to a production environment with managed solutions. If you need to copy your production environment into a sandbox environment, follow the instructions in [Copy an environment](#).

Using this process, you can use the model immediately after you import it. No other action is required to use it in Power Apps or Power Automate, but it's a good idea to perform a quick test in AI Builder first.

A model can only be added in a solution when it has a published version. When the solution is exported and imported in a new environment, only the published version of the model is installed in the new environment.

Note

When adding an AI model in a solution, only the model executable is included. The training data isn't included with the model.

If you get an error message during importing, go to [AI models fail to be imported in a new environment](#) for a possible resolution.

Automating the process

As AI models can be distributed across environments using solutions, you can automate your model lifecycle in the same way you would for other platform components. To learn more, go to [Application lifecycle management \(ALM\) with Microsoft Power Platform](#).

Changing imported models

Generally, we don't recommend changing imported models because it generates unmanaged customizations. These customizations can prevent the model from being properly updated in the future. Changes include updating model information, creating and training new versions, or republishing the model.

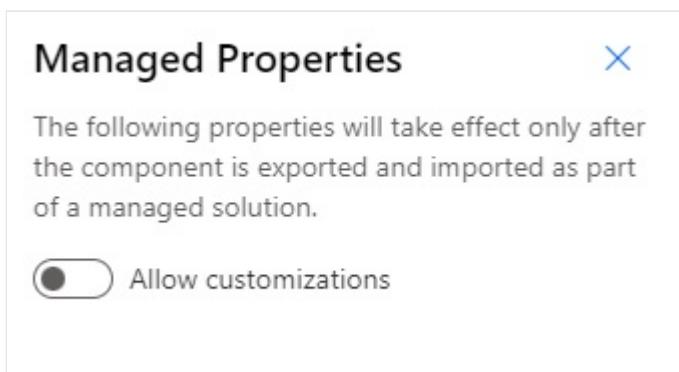
For some models, creating and training new versions have been disabled because training data isn't moved alongside the model. This applies to document processing, object detection, and entity extraction models.

If you accidentally perform actions after you've imported, delete the imported solution and then import the solution again.

To avoid accidental modification of a model after import, it might be a good idea to disable customization in the managed properties of the model before importing it.

To disable customizations, do the following:

1. From within the solution on the menu at the top, select **Managed Properties**.
2. Turn off **Allow customizations**.



After you've disabled customization, your model will include a note that you've limited the possible actions to it.

Importing status

For document processing and object detection models, the import process might continue after the import action is finished. When a model is continuing the import process, **Importing** appears on the list page of the AI Builder model. This is normal and can last several minutes.

Limitations

- You can't create and train a new version of an imported document processing, object detection, or entity extraction model because the training dataset isn't part of the imported solution. You should create a new model instead.
- You can't set a run schedule on imported category classification models.
- Importing an object detection model or document processing model should be done within one month of export. However, you can still import after that period if the source model remains unchanged after its export.
- If you're using a model within an app or a Power Automate flow, you need to explicitly add the app and the model to the solution. The model isn't considered an app or flow dependency.
- You can't create a new AI Builder model in solution explorer.

Continuously improve your model (preview)

Article • 12/13/2022

[This topic is pre-release documentation and is subject to change.]

After the model creation, it's likely that you'll need to improve your model regularly using production data. The *feedback loop* feature will help you automate this continuous process.

ⓘ Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- Only custom document processing models are supported by the feedback loop feature.

Select documents to add in the feedback loop

The first thing you'll have to do is select the data eligible for the feedback loop. This has to be done from within the Power Automate flow that runs your model.

New Power Automate flow

If you don't have a Power Automate flow to run your model yet, you can create a new one from the model detail page after publishing the model.

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. On the left pane, select **AI Builder > Models**.
3. Select the model for which you'd like to set up the feedback loop.
If your model isn't published yet, select **Publish** before continuing.
4. Select **Use model > Build intelligent automations**.

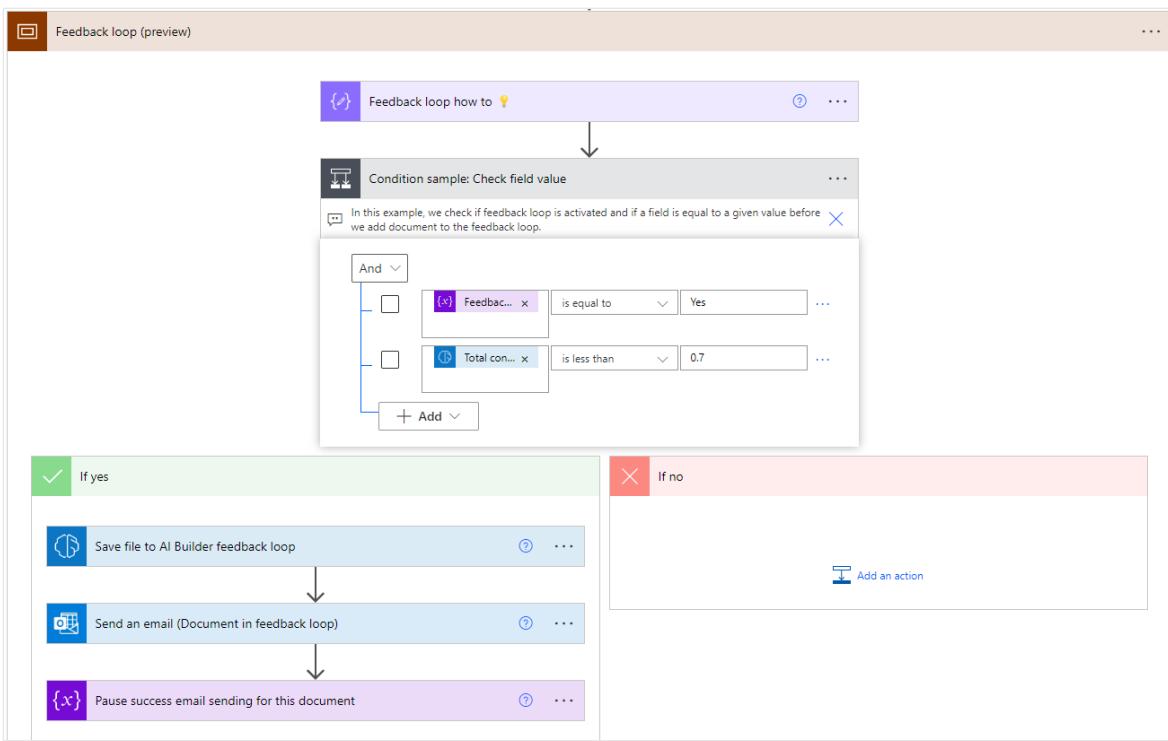
5. Select the template that best fits your needs.

How do you receive the documents to process? X

We'll create a sample cloud flow based on your selection that you can then customize.

-  **Microsoft 365 Outlook work account**
Documents are received attached to emails sent to a Microsoft 365 Outlook work account.
-  **Microsoft 365 Outlook shared mailbox account**
Documents are received attached to emails sent to a Microsoft 365 Outlook shared mailbox account.
-  **SharePoint document library**
Documents are uploaded in a SharePoint document library.
-  **OneDrive for Business folder**
Documents are uploaded in a OneDrive for Business folder.
-  **Other**
Choosing this option will create a sample flow with a manual trigger that can then be customized.

6. Validate the template connections to land in the Power Automate flow authoring experience.
7. Go to the **Feedback loop (preview)** section to edit the conditions that will allow you to add data in the feedback loop storage.



In this example, we updated the condition stating that confidence score of the **Total** field should be less than 0.7 (70%). If it is, the data goes in the feedback loop storage.

! Note

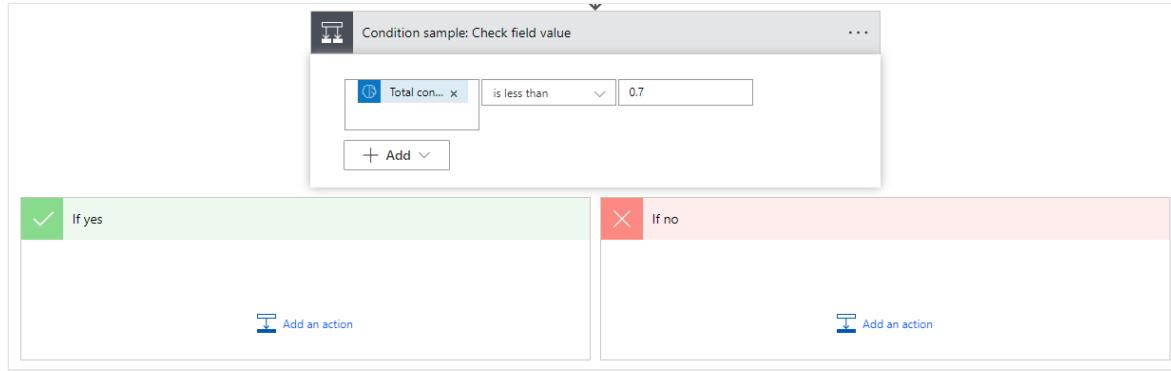
The feedback loop storage is a Microsoft Dataverse table within the same environment in which the flow runs. The table is called **AI Builder Feedback Loop**. Unless you need to delete records in the table, you don't need to access it directly to make the feedback loop work.

Existing Power Automate flow

If you already have a flow running with your AI Builder model, you'll be able to add the feedback loop storage functionality within.

1. Sign in to [Power Automate](#).
2. On the left pane, select **My flows**.
3. Select your Power Automate flow.
4. On the toolbar, select **Edit**.
5. After the AI Builder action card, insert a new step:
 - a. Select the plus sign (+) that appears when you hover over the arrow.
 - b. Select **Add an action**.

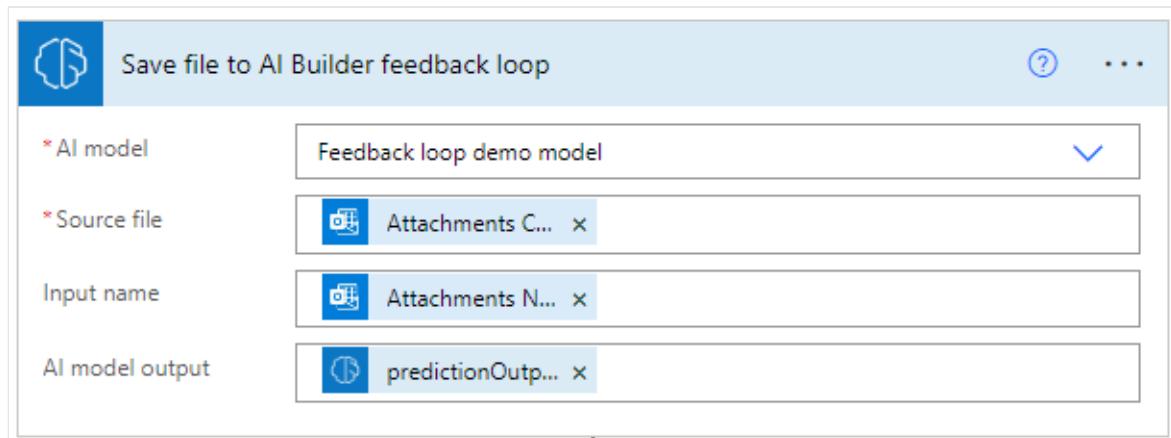
6. In the Actions list, select **Condition**. This is where you'll enter the condition that defines if the data needs to go in the feedback loop (yes) or not (no).



In this example, we added a condition stating that confidence score of the **Total** field should be less than 0.7 (70%). If it is, the data goes in the feedback loop storage.

1. In the **If yes** section:

- Add the action **Save file to AI Builder feedback loop**.
- Select the model you're using to place it in the **AI model** field.
- Select the source file to place it in the **Source file** field.
- Select the expression **predictionOutput** to place it in the **AI model output** field.

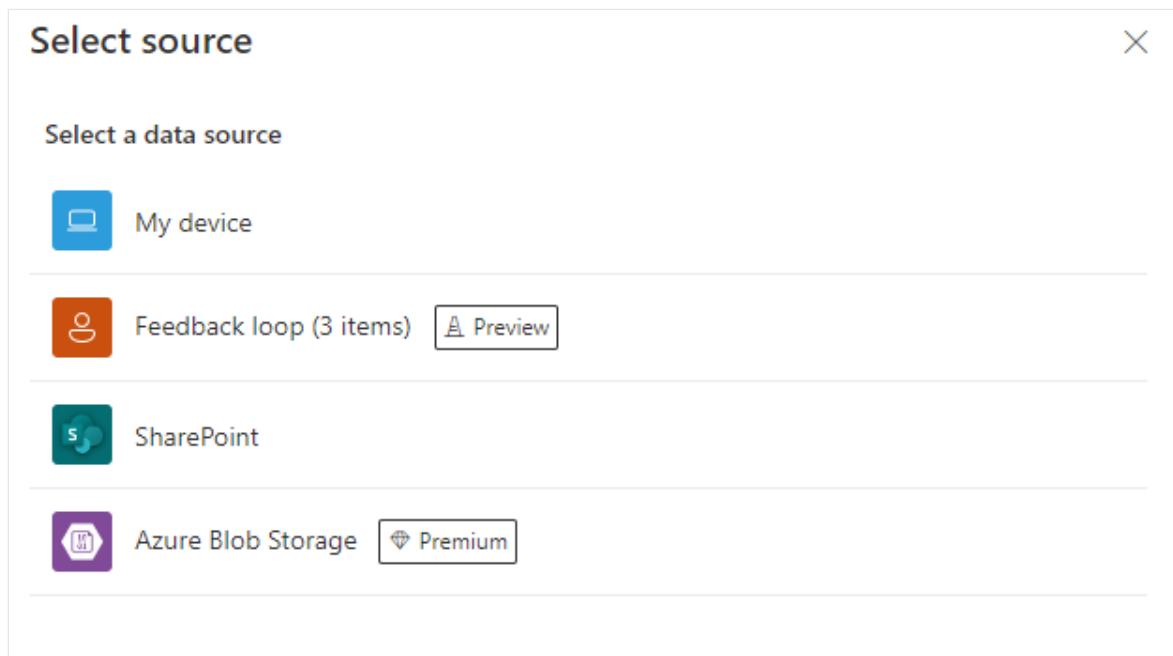


Use data in the feedback loop to improve your model

Now that you have data in the AI Builder feedback loop, you can use it to improve your model.

- Sign in to [Power Apps](#) or [Power Automate](#).
- On the left pane, select **AI Builder > Models**.
- Select the model to improve using data in the feedback loop. Models that can be improved have the status **Documents to review**.

4. Select **Edit model** and continue to the wizard section to add new training data.
5. For document processing models, select the desired collection and select **Add documents**. You'll get the **Feedback loop** data source option. If documents have been added to the feedback loop, they'll appear there.



6. Select **Feedback loop** and add the documents that could improve your model.
7. Tag these new documents and retrain the model.

Your model is now improved with new documents coming from the feedback loop.

Limitations

- The feedback loop feature is limited to custom document processing models.
- Feedback loop data can be added only from Power Automate flows.
- Feedback loop data can't transit between environments currently.
- The owner of the model and owner of the flow containing the feedback loop logic should be the same person.

Monitor model activity (preview)

Article • 05/31/2023

[This topic is pre-release documentation and is subject to change.]

As you use your AI models, you might need to access data to monitor the activity or consumption of your AI models.

ⓘ Important

- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- This feature might not be available in your region yet.

The **AI Builder activity** section of the Power Automate portal provides tables and graphs to monitor AI models and the data processed by the AI models, as well as track AI credits consumption.

The screenshot shows the 'AI Builder activity (preview)' page. On the left, there's a navigation sidebar with options like Home, Approvals, My flows, Create, Templates, Connectors, Data, Monitor (which is selected and highlighted with a red border), Cloud flow activity, Desktop flow activity, and AI Builder activity (preview) (which is also highlighted with a red border). The main content area has a title 'AI Builder activity (preview)' and a subtitle 'This section provides data to better understand your usage of AI models.' Below this are two buttons: 'Last 7 days' and 'All AI models'. A table titled 'Data processed details' follows, showing seven columns: 1. Data processed, 2. Data type, 3. Date, 4. Model, 5. Status, 6. Processed by, and 7. AI Credits. Each row in the table represents a document processed by the AI model 'GptPromptEngineering model' on May 22, 2023, at various times. The status for all rows is 'Processed', and the AI Credits column shows values of 0, 100, 100, 100, 100, 100, and 100 respectively. A 'See more' link is located at the top right of the table header.

Data processed details						
1 Data processed	2 Data type	3 Date	4 Model	5 Status	6 Processed by	7 AI Credits
<[system_start]> You will respond to human request...	Text	May 22, 10:34 AM (3 h ago)	GptPromptEngineering model	Processed	Unknown Manager	0
Document	Pdf	May 22, 04:54 AM (8 h ago)	Custom doc processing model	Processed	Unknown Manager	100
Document	Pdf	May 22, 04:53 AM (8 h ago)	Custom doc processing model	Processed	Unknown Manager	100
Document	Pdf	May 22, 04:53 AM (8 h ago)	Custom doc processing model	Processed	Unknown Manager	100
Document	Pdf	May 22, 04:53 AM (8 h ago)	Custom doc processing model	Processed	Unknown Manager	100
Document	Pdf	May 22, 04:52 AM (8 h ago)	Custom doc processing model	Processed	Unknown Manager	100
Document	Pdf	May 22, 04:47 AM (8 h ago)	Custom doc processing model	Processed	Unknown Manager	100

Legend:

1. **Data processed:** Contains the text input of the AI model predict action for text processing models, **Image**, or **Document** for other models.
2. **Data type:** Data type processed by the AI model: **Text**, **Jpeg**, **Png**, **Bmp**, **Pdf**, or **Unknown**.
3. **Date:** Processing date.
4. **Model:** The name of the AI model used. If the model has been deleted, you see **Deleted**.
5. **Status:** Processing status: **Processing**, **Processed**, or **Failed**.

6. **Processed by:** Name of the person who did the predict. This is typically the Power Automate flow owner, or the person who executed the Power Apps app.
7. **AI Credits:** Number of credits consumed for this AI predict action.

View AI Builder activity

The AI Builder activity screen contains AI models activity, including activity generated in [Power Apps](#).

1. Sign in to [Power Automate](#).
2. On the left navigation panel, select **Monitor > AI Builder activity (preview)**.
3. (Optional) Customize how the data appears by applying a filter for timeframe or by model. Do this by selecting the relevant heading.
By default, data displays for all AI models for the last seven (7) days.
4. (Optional) To display more data, select **See more**.

Monitoring data for makers and admins

The AI Builder activity section is helpful for makers who want to get monitoring on their AI models usage. It's also helpful for environment admins who want to monitor all activity in an environment.

Note

- The monitoring data is stored in the **AI Event** table in your Microsoft Dataverse. It persists in the table even if the model/flow/app are deleted.
- The **AI Event** Dataverse table contains input of the AI model predict actions for text scenarios only.

The data you can display depends on your role.

Role	What you can display
System Administrator	All activity for all AI models.
System Customizer	All activity for all AI models.
Basic User	Only your own activity of the AI models you have access to.

Role	What you can display
Environment Maker	Only your own activity of the AI models you have access to.

Manage AI Builder activity monitoring data

Effective management of historical data generated by your Power Automate Flows, Power Apps or other Power Platform products can ensure that your Dataverse environments remain efficient and cost-effective. By implementing data retention policies and utilizing features like [bulk record deletion of Dataverse](#) and the [Power Platform admin center](#), you can proactively manage the accumulation of historical data.

To create bulk-delete jobs in Dataverse, you need to have the **Bulk Delete** privilege in at least one of the roles that have been assigned to you.

Later in this article, you learn how to purge historical AI Builder activity monitoring data from your environment using Dataverse's built-in bulk-delete feature. This feature allows you to quickly and easily [remove large amounts of data](#) from your environment in compliance with your specific data retention policies. This ensures efficient data storage and performance management. In addition to ad-hoc bulk-delete jobs, you can schedule recurrent bulk-delete jobs that will find and delete records in a table that are, for example, `OlderThanXDays`.

You'll also learn how to:

- Identify the AI Builder activity monitoring data that could be purged.
- Create a bulk delete job to delete the data.

⊗ Caution

When you delete Dataverse data, it's permanently deleted from your environment. There's no way to recover individual records once they've been deleted.

Export a Dataverse table

The following table shows an AI Builder activity monitoring Dataverse table with potentially large data volumes/

Display name	System name	Details

Display name	System name	Details
AI Event	Msdyn_aievent	The AI event table stores activity data about AI models activity (<i>predicts</i>), such as the processed data type, processed data info for text scenarios, processing date, processing status, and credits consumed.

As the data is stored in your **AI Event** Dataverse table, you can export it in a CSV format. To learn how to export, go to [Export data](#).

The screenshot shows the Microsoft Power Platform admin center interface. On the left, there's a navigation pane with options: Home, Create, Learn, Apps, and Tables, with 'Tables' selected and highlighted by a red box. The main area shows 'Tables > AI Event'. Below that is a 'Table properties' section with columns for Name (AI Event), Primary column (Name), and Description (Last modified). At the top of the main area, there are several buttons: '+ New', 'Edit', 'Create an app', 'Using this table', 'Import', 'Export', and 'Advanced'. The 'Export' button is also highlighted with a red box.

Delete AI Builder activity monitoring data

To delete AI Builder activity monitoring data, you need to create a bulk-delete job. To bulk-delete data in Dataverse, follow the steps in this section.

Before performing bulk delete operations, thoroughly test and review your filter results. Bulk-delete operations are irreversible.

1. Sign in to the [Power Platform admin center](#).
2. On the left navigation pane, select **Environments** > select your environment > **Settings** (on the top menu bar).
3. Select **Data management** > **Bulk deletion**.
4. From the **All Bulk Deletion System Jobs** grid, select **New** on the command bar. This will open the **Bulk Deletion Wizard**, which allows you to define a query for the records you want deleted.
5. Select **Next**.
6. In the **Look for** list, select the **AI Events** table.
7. In the search criteria area, add the filter that should return the records that you want to be deleted. Here's an example that finds all AI Builder models activity that is older than six (6) months:

Define Search Criteria



Select search criteria to identify records to delete.

The screenshot shows the 'Define Search Criteria' page. At the top, there are dropdown menus for 'Look for:' set to 'AI Events' and 'Use Saved View:' set to '[new]'. Below these are buttons for 'Clear', 'Group AND', 'Group OR', and a dropdown menu currently showing 'Processing Date'. Under 'Processing Date', there is a sub-menu 'Older Than X Months' with the value '6' selected. A 'Select' button is at the bottom of the search criteria area.

8. Select Next.

9. In the **Name** text box, type a name for the bulk deletion job (for example, **Bulk-delete of AI models monitoring data older than 6 months**).

10. In the **At scheduled time** section, select a date and time for the job start time.
Select a time when users aren't typically online.

11. Select the **Run this job after every** checkbox.

12. In the **days** list, select the frequency you want the job to run.

13. If you want a notification e-mail sent, select the **Send an email to me** (`email@domain.com`) **when this job is finished** checkbox.

14. Select Next.

15. In the **Review and Submit Bulk Deletion Details** screen, review the bulk deletion job, and then select **Submit** to create the recurring job.

Overview of prebuilt AI models

Article • 05/08/2023

AI Builder prebuilt models help you add intelligence to apps and flows without having to gather data and then build, train, and publish your own models. For example, in Power Apps you can add a component based on a prebuilt model that recognizes contact information from business cards. You can use a prebuilt model in Power Automate to analyze whether customer feedback was positive or negative.

Prebuilt models are available in Power Automate and—depending on the model—in Power Apps. The following prebuilt models are currently available in AI Builder.

Model type	Availability
Azure OpenAI Service (preview)	Power Automate and Power Apps
Sentiment analysis	Power Automate and Power Apps*
Language detection	Power Automate and Power Apps*
Text recognition	Power Automate and Power Apps
Invoice processing	Power Automate and Power Apps*
Entity extraction	Power Automate and Power Apps*
Key phrase extraction	Power Automate and Power Apps*
Category classification	Power Automate and Power Apps*
Text translation	Power Automate and Power Apps*
ID reader	Power Automate and Power Apps*
Business card reader	Power Automate and Power Apps
Receipt processing	Power Automate and Power Apps

Power Apps*: To learn more, go to [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

See also

[AI Builder in Power Automate overview](#)

[AI Builder in Power Apps overview](#)

[Feature availability by region](#)

Azure OpenAI Service model overview (preview)

Article • 06/06/2023

[This topic is pre-release documentation and is subject to change.]

The Azure OpenAI Service generative text capability is a powerful tool in AI Builder. This preview capability enables you to quickly and easily build AI-powered applications that use text generated from your input, like these:

- A tool that quickly and accurately summarizes long documents and extracts key information.
- A tool that helps draft responses to user queries.
- A tool that classifies text into different categories.
- A tool that translates text from one language to another.

Text generation is powered by Azure OpenAI Service, which is built on Generative Pre-trained Transformer (GPT) technology. These large language models have been trained on a massive amount of text data, which enables them to generate text that's similar to human-written text.

Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- [View our preview terms ↗](#).
- This capability might not be available in your region yet.
- This capability may be subject to usage limits or capacity throttling.

Explore the model

The Azure OpenAI Service model has a prompt engineering interface with sample prompts you can try. You can also create your own prompts to instruct the model.

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. On the left navigation panel, select **AI Builder > Explore** (Power Automate) or **AI models > Build an AI model** (Power Apps).
3. Select **Text > Azure OpenAI Service** (Power Automate) or **Text > Text generation** (Power Apps).

The screenshot shows the 'AI Models > Explore' page in Power Automate. The left sidebar has sections for Home, Approvals, My flows, Create, Templates, Connectors, Data, Monitor, AI Builder (which is highlighted with a red box), and Explore. The main content area has tabs for All, Documents, Text (which is highlighted with a red box), Structured data, and Images. The 'Text' tab is selected. There are six cards in the main content area, each with a preview button. The first card, 'Azure OpenAI Service', is highlighted with a red box and describes creating text, answering questions, summarizing documents, and more with GPT. Other cards include Sentiment analysis (detecting positive, negative, or neutral sentiment), Category classification (classifying customer feedback into predefined categories), Key phrase extraction (extracting most relevant words and phrases from text), Language detection (detecting the predominant language of a text document), and Text translation (detecting and translating more than 90 supported languages).

The next window that opens is where you can create text that can be used for many tasks. [Learn how text generation in Azure OpenAI Service works.](#)

See also

- [How text generation in Azure OpenAI Service works \(preview\)](#)
- [Use your Azure OpenAI Service model in Power Apps \(preview\)](#)
- [Use your Azure OpenAI Service model in Power Automate \(preview\)](#)

How text generation in Azure OpenAI Service works (preview)

Article • 06/06/2023

[This topic is pre-release documentation and is subject to change.]

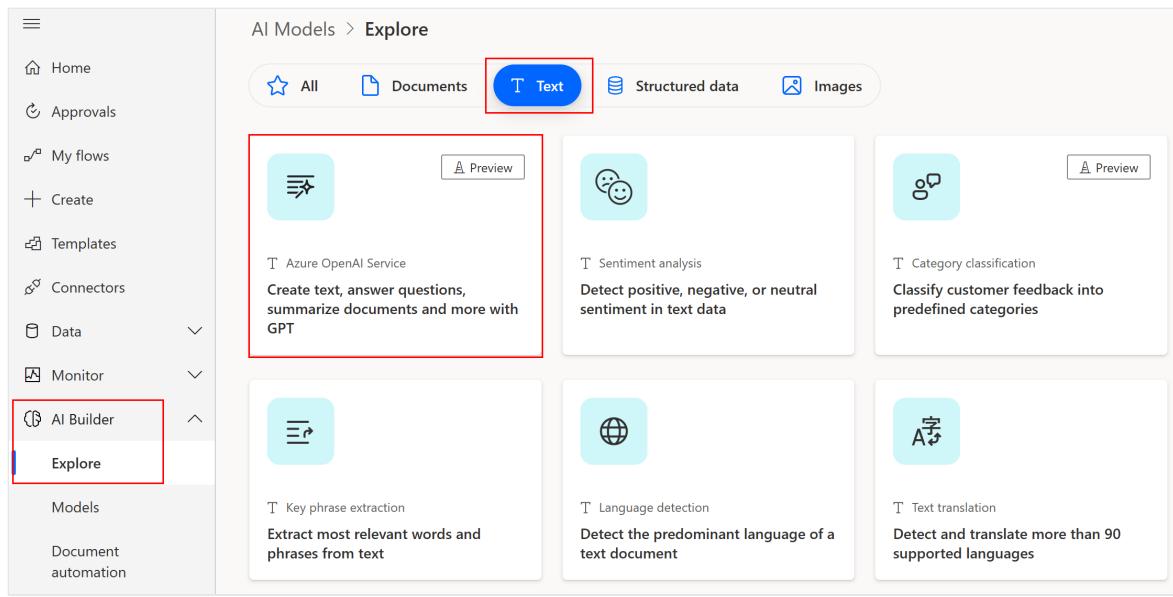
This article explains *prompt engineering* and other key concepts to help you create powerful applications that can generate text from your input. *Prompt* is a natural language instruction that tells the machine learning model to perform a task. The GPT model uses the prompt to determine the structure and content of the text it needs to generate. *Prompt engineering* is the process of creating and refining the input that's used to generate text with the Azure OpenAI Service generative text capability.

ⓘ Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- [View our preview terms ↗](#).
- This capability may not be available in your region yet.
- This capability may be subject to usage limits or capacity throttling.

Open the prompt engineering interface

1. Sign in to [Power Apps ↗](#) or [Power Automate ↗](#).
2. On the left navigation panel, select **AI Builder > Explore** (Power Automate) or **AI models > Build an AI model** (Power Apps).



3. Select **Text** > **Azure OpenAI Service** (Power Automate) or **Text** > **Text generation** (Power Apps).

The prompt engineering window opens. Use this exploratory experience to learn how to instruct the model to generate desired text. The prompt engineering interface comes with sample prompts to help you try out the capability.

Use specific text for more relevant responses

The goal of prompt engineering is to create an input that's as specific as possible to get a more relevant response from the AI model. Your prompts should be specific to a topic and convey your intent.

A prompt might include the following information:

- The topic
- Keywords or phrases that are associated with the topic
- The tone of the response
- The target audience

For example:

Generate a response to the text below. Be apologetic, humble, and creative with the response. The response should restate the problem to acknowledge the issue. The response should indicate that the problem will be addressed shortly.

Create text, summarize documents, and more with GPT

This model runs on Azure OpenAI Service and can be used for many tasks that involve creating text. Try a template to see how to use generative AI in a variety of scenarios. You can also try writing a prompt from scratch. When you're done, you can use the model in an app or a flow.

✖ [Prebuilt model](#) [Preview](#)

Templates

Respond to a complaint

Prompt

Generate a response to the text below. Be apologetic, humble, and creative with the response. The response should restate the problem to acknowledge the issue. The response should indicate that the problem will be addressed shortly.

[Start of text]
My thing arrived broken
[End of text]

288/5000

▶ Test it out

Response

AI-generated content can have mistakes. Make sure it's accurate and appropriate before using it. [Read preview terms](#)

I'm so sorry to hear that your item arrived broken. I understand how frustrating this must be and I apologize for the inconvenience. I'll be sure to look into this right away and do whatever I can to make it right.

Like Dislike

[View documentation](#) [Use prebuilt model](#) ▾

If the generated text is too long or contains irrelevant information, adjust the prompt. A good prompt has the following characteristics:

- **Clear and concise:** It's written in clear and concise language that's easy to understand.
- **Specific:** It's specific enough to guide the GPT model in the right direction.
- **Contextual:** It provides enough context for the GPT model to generate meaningful output.
- **Relevant:** It's relevant to the task and provides the GPT model with enough information to generate meaningful output.

Parts of a prompt

There are generally two parts to a prompt for a GPT model, the *instruction* and the *context*.

- **The instruction** is the first part of the prompt. It should provide clear directions on what the model should do; for example, "Summarize this email in three bullets."
- **The context** is the second part of the prompt. It should provide the information the model needs to generate an appropriate response; for example, "The email contains customer feedback from the past week."

Usually in an automation task, the instruction remains constant. The context gets replaced by dynamic content so that the instruction can be reused in an automated

workflow.

For example, you're building a workflow to summarize customer feedback every week and generate a response. You might build an automation to filter email that contains customer feedback and include a prompt like this:

"Summarize the following customer feedback into bullet points, identify each distinct topic. Additionally, generate a positive tone response, indicating that we'll take action on key points they have highlighted."

The screenshot shows the Azure AI Studio interface. At the top, it says "Create text, answer questions, summarize documents and more with GPT". Below that, it says "This model runs on Azure OpenAI Service and can be used for many tasks that involve creating text. Try a template to see how to use generative AI in a variety of scenarios. You can also try writing instructions from scratch. When you're done, you can use the model in an app or a flow." There are "Prebuilt model" and "Premium" buttons. The main area has "Templates" (with "Summarize text" selected) and "Instructions" (containing the prompt text). The response section shows "Bullet Points:" and "Positive Tone Response:". At the bottom, there are "View documentation", "Use in a flow", and "Use in an app" buttons.

The context is the entire body of text from the weekly feedback emails. The feedback changes weekly, but the instruction part of the prompt remains the same.

Human oversight

Human oversight is an important step when working with content that's generated from a GPT model. Large language models like GPT are trained on huge amounts of data. AI-generated content can contain errors and biases. A human should review it before it's posted online, sent to a customer, or used to inform a business decision. Human oversight helps not only to identify potential errors and biases, but also to make sure the content is relevant to the intended use case and aligns with the company's values.

Human review can also help to identify any issues with the GPT model itself. For example, if the model is generating content that isn't relevant to the intended use case, then you may need to adjust the prompt.

Responsible AI

We're committed to creating [responsible AI](#) by design. Our work is guided by a [core set of principles](#): fairness, reliability and safety, privacy and security, inclusiveness, transparency, and accountability. We're putting these principles into practice across the company to develop and deploy AI that has a positive impact on society. We take a comprehensive approach, combining innovative research, exceptional engineering, and responsible governance. Alongside OpenAI's leading research on AI alignment, we're advancing a framework for the safe deployment of our own AI technologies that's aimed to help guide the industry toward more responsible outcomes.

[Learn more about transparency in the Azure OpenAI Service.](#)

See also

[Azure OpenAI Service model overview \(preview\)](#)

[Use your Azure OpenAI Service model in Power Apps \(preview\)](#)

[Use your Azure OpenAI Service model in Power Automate \(preview\)](#)

Sentiment analysis prebuilt model

Article • 03/24/2023

The sentiment analysis prebuilt model detects positive or negative sentiment in text data. You can use it to analyze social media, customer reviews, or any text data you're interested in. Sentiment analysis evaluates text input, and gives scores and labels at a sentence and document level. The scores and labels can be positive, negative, or neutral. At the document level, there can also be a "mixed" sentiment label, which has no score. The sentiment of the document is determined by aggregating the sentence scores.

Use in Power Apps

Explore sentiment analysis

You can try out the sentiment analysis model before you import it into your flow by using the "try it out" feature.

1. Sign in to [Power Apps](#).
2. In the left pane, select **AI Builder > Explore**.
3. Under **Get straight to productivity**, select **Sentiment Analysis**.
4. In the **Sentiment Analysis** window, select **Try it out**.
5. Select predefined text samples to analyze, or add your own text in the **Add your own here** box to see how the model analyzes your text.

Use the formula bar

You can integrate your AI Builder sentiment analysis models in Power Apps Studio by using the formula bar. For more information, see [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Use in Power Automate

If you want to use this prebuilt model in Power Automate, you can find more information in [Use the sentiment analysis prebuilt model in Power Automate](#).

Supported language and data format

- **Language:** German, Spanish, English, French, Hindi, Italian, Japanese, Korean, Dutch, Norwegian, Portuguese (Brazil), Portuguese (Portugal), Turkish, Chinese (Simplified), Chinese (Traditional)
- Documents can't exceed 5,120 characters.

Model output

If text is detected, the sentiment analysis model outputs the following information:

- **Sentiment:**
 - Positive
 - Negative
 - Neutral
 - Mixed
- **Confidence score:** Value in the range from 0 through 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate.
- **Sentences:** List of sentences from the input text, with analysis of its sentiments.
 - **Sentiment:**
 - Positive
 - Negative
 - Neutral
 - Mixed
 - **Sentence confidence score:** Value in the range from 0 through 1. Values close to 1 indicate greater confidence that the sentiment is accurate.

Limits

The following applies to calls made per environment across the following prebuilt models: language detection, sentiment analysis, and key phrase extraction.

Action	Limit	Renewal period
Calls (per environment)	400	60 seconds

See also

- Training: Analyze the sentiment of text with AI Builder (module)
- Video: How to set up instant Microsoft Teams notifications for negative emails in 1 minute ↗

Language detection prebuilt model

Article • 01/05/2023

The language detection prebuilt model identifies the predominant language of a text document. The model analyzes the text and returns the detected language and a numeric score from 0 through 1. Scores close to 1 indicate higher confidence in the result. The detected language is returned as the "script" of the language. For instance, for the phrase "I have a dog", it will return "en" instead of "en-US". The response for languages that can't be detected is **unknown**.

Use in Power Apps

Use the formula bar

You can integrate your AI Builder language detection model by using the formula bar. For more information, see [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Use in Power Automate

If you want to use this prebuilt model in Power Automate, you can find more information in [Use the language detection prebuilt model in Power Automate](#).

Supported language and data format

- Documents can't exceed 5,120 characters.
- For information about language support, see [Text Analytics API v3 language support](#).

Model output

If text is detected, the language detection model outputs the following information:

- **Results:** A list of languages detected in the input text.
- **Language:** Script version of the language code (for example, "en", "fr", "zh_chs", "ru").
- **Confidence score:** Numeric value from 0 through 1, where values close to 1 indicate higher confidence.

Limits

The following applies to calls made per environment across the following prebuilt models: language detection, sentiment analysis, and key phrase extraction.

Action	Limit	Renewal period
Calls (per environment)	400	60 seconds

See also

[Training: Identify the language of text with AI Builder \(module\)](#)

Text recognition prebuilt model

Article • 03/13/2023

The text recognition prebuilt model extracts words from documents and images into machine-readable character streams. It uses state-of-the-art optical character recognition (OCR) to detect printed and handwritten text in images.

This model processes images and document files to extract lines of printed or handwritten text.

Use in Power Apps

The text recognition prebuilt model is available in Power Apps by using the text recognizer component. More information: [Use the text recognizer component in Power Apps](#)

Use in Power Automate

For information about how to use this model in Power Automate, see [Use the text recognition prebuilt model in Power Automate](#).

Supported language, format, and size

The files you can scan with the text recognition model must have these characteristics:

- **Language for print text:** Afrikaans, Albanian, Angika (Devanagiri), Arabic, Asturian, Awadhi-Hindi (Devanagiri), Azerbaijani (Latin), Bagheli, Basque, Belarusian (Cyrillic), Belarusian (Latin), Bhojpuri-Hindi (Devanagiri), Bislama, Bodo (Devanagiri), Bosnian (Latin), Brajbha, Breton, Bulgarian, Bundeli, Buryat (Cyrillic), Catalan, Cebuano, Chamling, Chamorro, Chhattisgarhi (Devanagiri), Chinese (Simplified), Chinese (Traditional), Cornish, Corsican, Crimean Tatar (Latin), Croatian, Czech, Danish, Dari, Dhimal (Devanagiri), Dogri (Devanagiri), Dutch, English, Erzya (Cyrillic), Estonian, Faroese, Fijian, Filipino, Finnish, French, Friulian, Gagauz (Latin), Galician, German, Gilbertese, Gondi (Devanagiri), Greenlandic, Gurung (Devanagiri), Haitian Creole, Halbi (Devanagiri), Hani, Haryanvi, Hawaiian, Hindi, Hmong Daw (Latin), Ho(Devanagiri), Hungarian, Icelandic, Inari Sami, Indonesian, Interlingua, Inuktitut (Latin), Irish, Italian, Japanese, Jaunsari (Devanagiri), Javanese, Kabuverdianu, Kachin (Latin), Kangri (Devanagiri), Karachay-Balkar, Kara-Kalpak (Cyrillic), Kara-Kalpak (Latin), Kashubian, Kazakh (Cyrillic), Kazakh (Latin), Khaling, Khasi, K'iche',

Korean, Korku, Koryak, Kosraean, Kumyk (Cyrillic), Kurdish (Arabic), Kurdish (Latin), Kurukh (Devanagiri), Kyrgyz (Cyrillic), Lakota, Latin, Lithuanian, Lower Sorbian, Lule Sami, Luxembourgish, Mahasu Pahari (Devanagiri), Malay (Latin), Maltese, Malto (Devanagiri), Manx, Maori, Marathi, Mongolian (Cyrillic), Montenegrin (Cyrillic), Montenegrin (Latin), Neapolitan, Nepali, Niuean, Nogay, Northern Sami (Latin), Norwegian, Occitan, Ossetic, Pashto, Persian, Polish, Portuguese, Punjabi (Arabic), Ripuarian, Romanian, Romansh, Russian, Sadri (Devanagiri), Samoan (Latin), Sanskrit (Devanagari), Santali (Devanagiri), Scots, Scottish Gaelic, Serbian (Latin), Sherpa (Devanagiri), Sirmauri (Devanagiri), Skolt Sami, Slovak, Slovenian, Somali (Arabic), Southern Sami, Spanish, Swahili (Latin), Swedish, Tajik (Cyrillic), Tatar (Latin), Tetum, Thangmi, Tongan, Turkish, Turkmen (Latin), Tuvan, Upper Sorbian, Urdu, Uyghur (Arabic), Uzbek (Arabic), Uzbek (Cyrillic), Uzbek (Latin), Volapük, Walser, Welsh, Western Frisian, Yucatec Maya, Zhuang, Zulu

- **Language for handwritten text:** English, Chinese (Simplified), French, German, Italian, Japanese, Korean, Portuguese, Spanish
- **Format:**
 - JPG
 - PNG
 - BMP
 - PDF
- **Size:** 20 MB maximum
- For PDF documents, only the first 2,000 pages are processed.

Model output

If a document is detected, the text recognition model outputs the following information:

- **Results:** A list of lines extracted from the input text.
- **Text:** Strings containing the line of text detected.
- **BoundingBox:** Four values representing the bounding box, described by using the top and left positions along with its width and height.

Limits

Action	Limit	Renewal period
Text recognition calls (per environment)	480	60 seconds

See also

Training: Recognize text with AI Builder (module)

Invoice processing prebuilt AI model

Article • 04/06/2023

The invoice processing prebuilt AI model extracts key invoice data to help automate the processing of invoices. The invoice processing model is optimized to recognize common invoice elements like invoice ID, invoice date, amount due, and more.

Use in Power Apps

For information on how to use the invoice processing prebuilt model in Power Apps, go to [Use the invoice processing prebuilt model in Power Apps](#).

Use in Power Automate

For information on how to use the invoice processing prebuilt model in Power Automate, go to [Use the invoice processing prebuilt model in Power Automate](#).

Supported languages and files

The following languages are supported: Dutch (Netherlands), English (Australia), English (Canada), English (India), English (United Kingdom), English (United States), French (France), German (Germany), Italian (Italy), Portuguese (Portugal), Spanish (Spain).

To get the best results, provide one clear photo or scan per invoice.

- The image format must be JPEG, PNG, or PDF.
- The file size must not exceed 20 MB.
- The image dimensions must be between 50 x 50 pixels and 10,000 x 10,000 pixels.
- PDF dimensions must be at most 17 x 17 inches, which is the equivalent of the Legal or A3 paper sizes or smaller.
- For PDF documents, only the first 2,000 pages are processed.

Model output

If an invoice is detected, the invoice processing model will output the following information:

Property	Definition
----------	------------

Property	Definition
Amount due (text)	Amount due as written on the invoice.
Amount due (number)	Amount due in standardized number format. Example: 1234.98.
Confidence of amount due	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Billing address	Billing address.
Confidence of billing address	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Billing address recipient	Billing address recipient.
Confidence of billing address recipient	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer address	Customer address.
Confidence of customer address	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer address recipient	Customer address recipient.
Confidence of customer address recipient	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer ID	Customer ID.
Confidence of customer ID	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer name	Customer name.
Confidence of customer name	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer tax ID	The taxpayer number associated with the customer.
Confidence of customer tax ID	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).

Property	Definition
Due date (text)	Due date as written on the invoice.
Due date (date)	Due date in standardized date format. Example: 2019-05-31.
Confidence of due date	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Invoice date (text)	Invoice date as written on the invoice.
Invoice date (date)	Invoice date in standardized date format. Example: 2019-05-31.
Confidence of invoice date	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Invoice ID	Invoice ID.
Confidence of invoice ID	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Invoice total (text)	Invoice total as written on the invoice.
Invoice total (number)	Invoice total in standardized date format. Example: 2019-05-31.
Confidence of invoice total	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Line Items	<p>The line items extracted from the invoice. Confidence scores are available for each column.</p> <ul style="list-style-type: none"> • Line item amount: Amount for a line item. Returned in text and number format. • Line item description: Description for a line item. Returned in text format. • Line item quantity: Quantity for a line item. Returned in text and number format. • Line item unit price: Unit price for a line item. Returned in text and number format. • Line item product code: Product code for a line item. Returned in text format. • Line item unit: Unit for a line item (for example, kg and lb). Returned in text format. • Line item date: Date for a line item. Returned in text and date format. • Line item tax: Tax for a line item. Returned in text and number format. • Line item all columns: Returns all the columns from the line item as a line of text.

Property	Definition
Payment terms	The terms of payment for the invoice.
Confidence of payment terms	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Purchase order	Purchase order.
Confidence of purchase order	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Previous unpaid balance (text)	Previous unpaid balance as written on the invoice.
Previous unpaid balance (number)	Previous unpaid balance in standardized number format. Example: 1234.98.
Confidence of previous unpaid balance	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Remittance address	Remittance address.
Confidence of remittance address	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Remittance address recipient	Remittance address recipient.
Confidence of remittance address recipient	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Service address	Service address.
Confidence of service address	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Service address recipient	Service address recipient.
Confidence of service address recipient	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Service start date (text)	Service start date as written on the invoice.

Property	Definition
Service start date (date)	Service start date in standardized date format. Example: 2019-05-31.
Confidence of service start date	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Service end date (text)	Service end date as written on the invoice.
Service end date (date)	Service end date in standardized date format. Example: 2019-05-31.
Confidence of service end date	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Shipping address	Shipping address.
Confidence of shipping address	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Shipping address recipient	Shipping address recipient.
Confidence of shipping address recipient	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Subtotal (text)	Subtotal as written on the invoice.
Subtotal (number)	Subtotal in standardized number format. Example: 1234.98.
Confidence of subtotal	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Total tax (text)	Total tax as written on the invoice.
Total tax (number)	Total tax in standardized number format. Example: 1234.98.
Confidence of total tax	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Vendor address	Vendor address.
Confidence of vendor address	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).

Property	Definition
Vendor address recipient	Vendor address recipient.
Confidence of vendor address recipient	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Vendor name	Vendor name.
Confidence of vendor name	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Vendor tax ID	The taxpayer number associated with the vendor.
Confidence of vendor tax ID	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Detected text	Line of recognized text from running OCR on an invoice. Returned as a part of a list of text.
Detected key	Key-value pairs are all the identified labels or keys and their associated responses or values. You can use these to extract additional values that aren't part of the predefined list of fields.
Detected value	Key-value pairs are all the identified labels or keys and their associated responses or values. You can use these to extract additional values that aren't part of the predefined list of fields.

Key-value pairs

Key-value pairs are all the identified labels or keys and their associated responses or values. You can use these to extract additional values that aren't part of the predefined list of fields.

To visualize all key-value pairs detected by the invoice processing model, you can add a **Create HTML table** action in your flow as shown in the screenshot and run the flow.

↓

Extract information from invoices

?

...

↓

{▽} Create HTML table

?

...

* From X

* Columns ▾

Header	Value
Key	<input type="button" value="Detected key"/> X
Value	<input type="button" value="Detected value"/> X

Hide advanced options ^

{ } Create HTML table 0s

INPUTS Show raw inputs >

Format

Html

From

```
[{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "DOCUMENTO Factura NÚMERO", "value": "17635", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "NÚMERO", "value": "17635", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}], [{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "FECHA", "value": "10/Junio/2023", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "FECHA", "value": "10/Junio/2023", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}, {"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "Tel ::", "value": "91 123 45 67", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "Tel ::", "value": "91 123 45 67", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}, {"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "CIF:", "value": "A-01234567", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "CIF:", "value": "A-01234567", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}, {"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "CIF:", "value": "B-98765432", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "CIF:", "value": "B-98765432", "confidence": 0.875, "location": "Text", "isPrimary": false}, "isPrimary": false}], [{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "NÚMERO", "value": "17635", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "NÚMERO", "value": "17635", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}], [{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "FECHA", "value": "10/Junio/2023", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "FECHA", "value": "10/Junio/2023", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}], [{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "Tel ::", "value": "91 123 45 67", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "Tel ::", "value": "91 123 45 67", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}], [{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "CIF:", "value": "A-01234567", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "CIF:", "value": "A-01234567", "confidence": 0.875, "location": "Text", "isPrimary": true}, "isPrimary": true}], [{"@odata.type": "#Microsoft.Dynamics.CRM.expando", "key": "CIF:", "value": "B-98765432", "confidence": 0.875, "keyLocation": {"@odata.type": "#Microsoft.Dynamics.CRM.ExpandoKeyLocation", "key": "CIF:", "value": "B-98765432", "confidence": 0.875, "location": "Text", "isPrimary": false}, "isPrimary": false}],

OUTPUTS Show raw outputs >



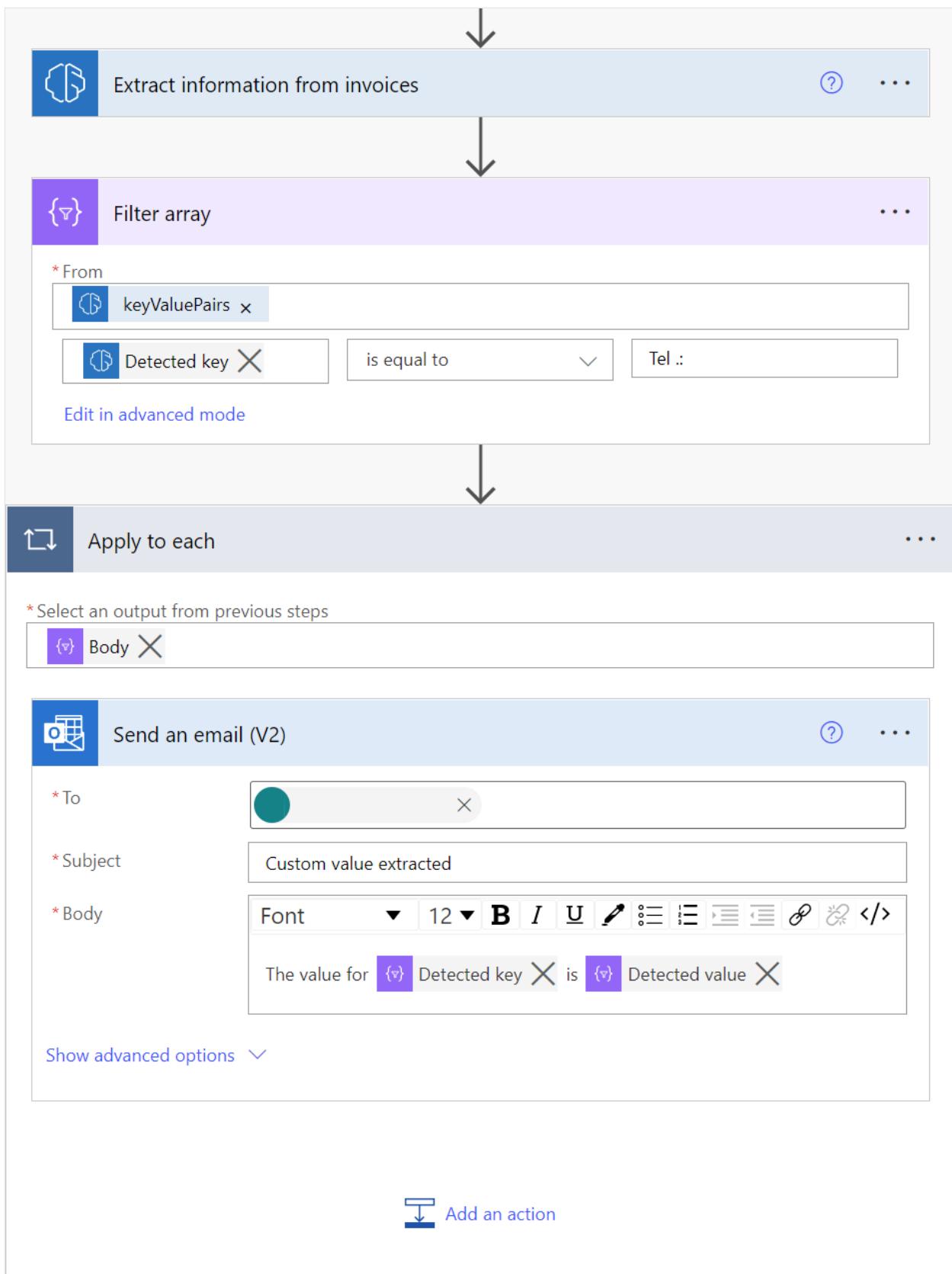
Body



| Key    | Value         |
|--------|---------------|
| NÚMERO | 17635         |
| FECHA  | 10/Junio/2023 |
| Tel :: | 91 123 45 67  |
| CIF:   | A-01234567    |
| CIF:   | B-98765432    |


```

To extract a specific key for which you know its value, you can use the **Filter array** action as shown on the screenshot below. In the example of the screenshot, we want to extract the value for the key **Tel ::**



Limits

The following limit applies to calls made per environment across document processing models including prebuilt models: receipt processing and invoice processing.

Action	Limit	Renewal period
--------	-------	----------------

Action	Limit	Renewal period
Calls (per environment)	360	60 seconds

Create a custom invoice processing solution

The invoice processing prebuilt AI model is designed to extract common fields found in invoices. Because every business is unique, you might want to extract fields other than those included in this prebuilt model. It can also be the case that some standard fields aren't well extracted for a particular type of invoice you work with. To address this, there are two options:

- **View raw OCR results:** Every time the invoice processing prebuilt AI model processes a file you provide, it also does an OCR operation to extract every word written on the file. You can access the raw OCR results on the detected text output provided by the model. A simple search on the content returned by detected text might be enough to get the data you need.
- **Use document processing:** With AI Builder, you can also build your own custom AI model to extract specific fields and tables you need for the documents you work with. Just [create a document processing model](#) and train it to extract all the information from an invoice that doesn't work well with the invoice extraction model.

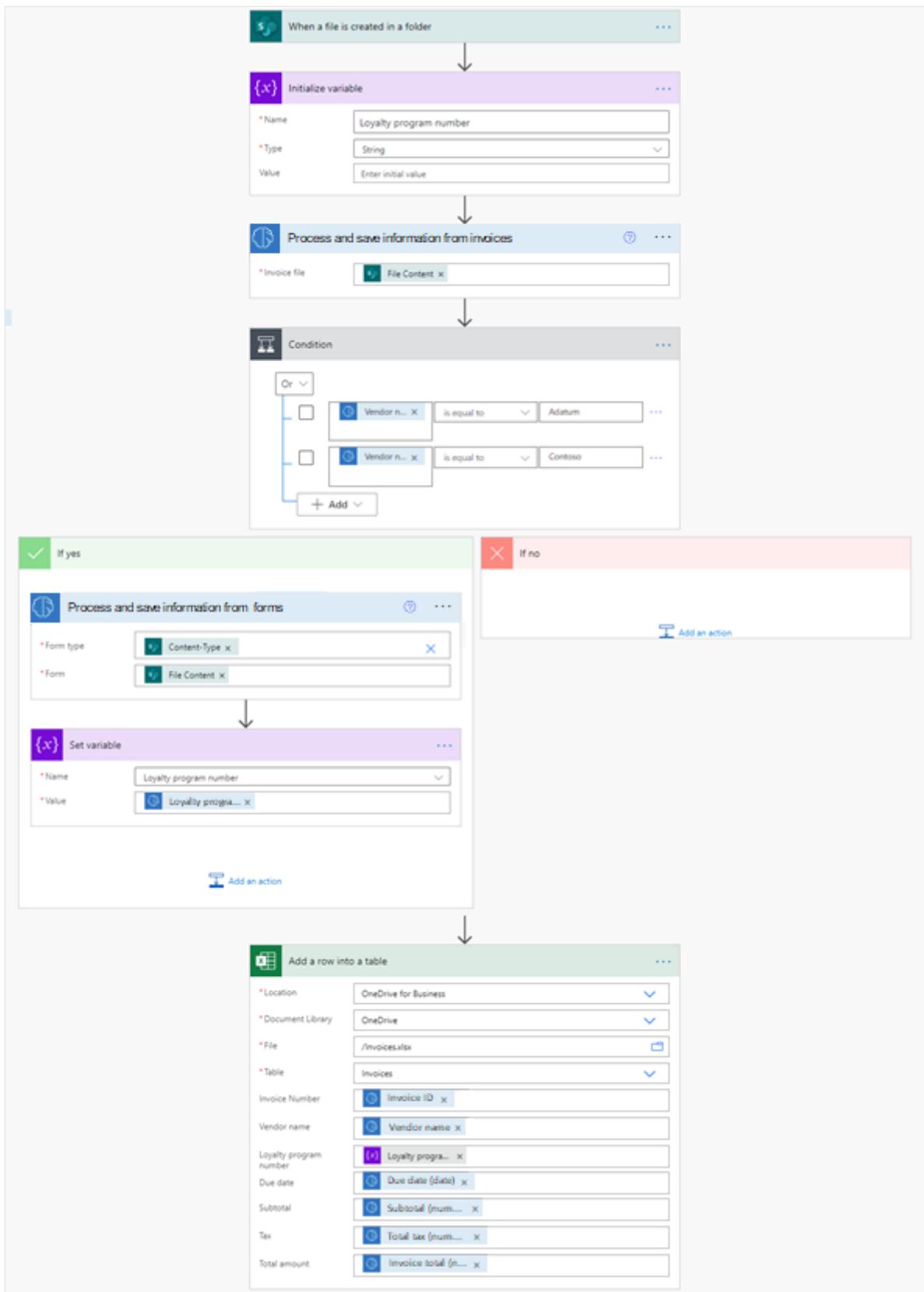
Once you train your custom document processing model, you can combine it with the invoice processing prebuilt model in a Power Automate flow.

Here are some examples:

Use a custom document processing model to extract additional fields that aren't returned by the invoice processing prebuilt model

In this example, we've trained a custom document processing model to extract a *loyalty program number*, only present in invoices from providers Adatum and Contoso.

The flow is triggered when a new invoice is added to a SharePoint folder. It then calls the invoice processing prebuilt AI model to extract its data. Next, we check if the vendor for the invoice that has been processed is either from Adatum or Contoso. If it's the case, we then call a custom document processing model that we've trained to get that loyalty number. Finally, we save the extracted data from the invoice in an Excel file.

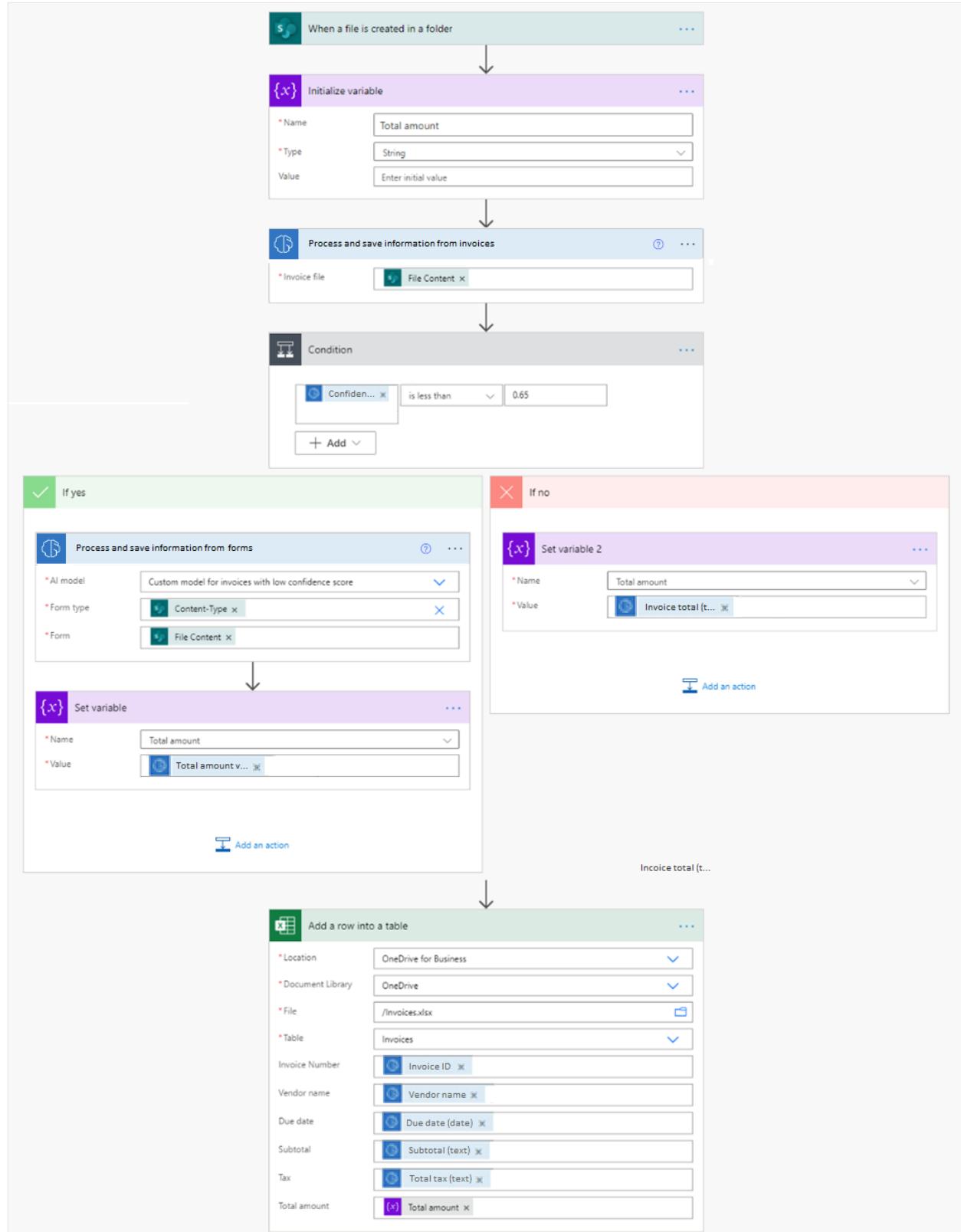


Use a custom document processing model if the confidence score for a field returned by the invoice processing prebuilt model is low

In this example, we've trained a custom document processing model to extract the total amount from invoices where we usually get a low confidence score when using the

invoice processing prebuilt model.

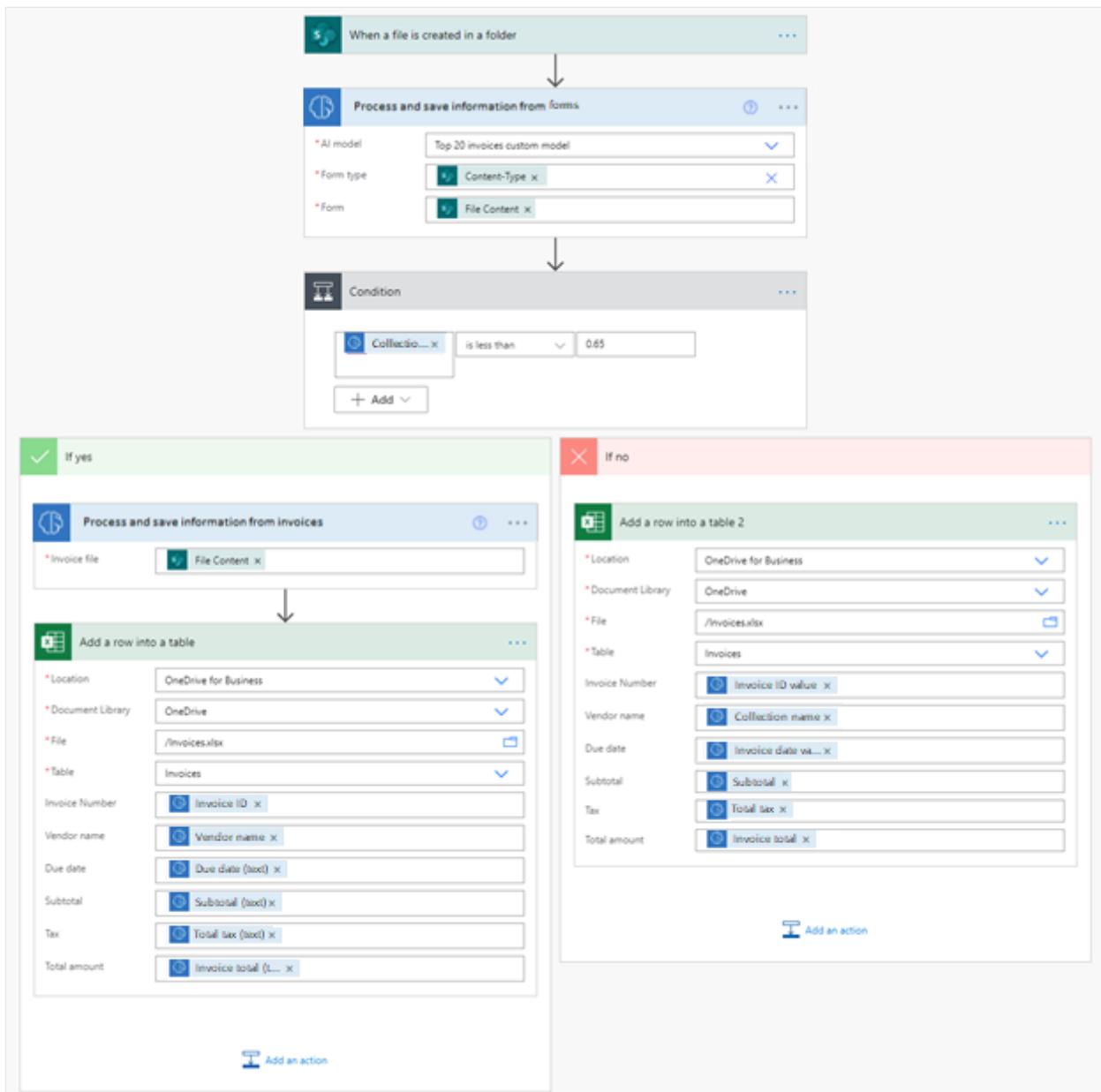
The flow is triggered when a new invoice is added to a SharePoint folder. It then calls the invoice processing prebuilt AI model to extract its data. Next, we check if the confidence score for the *Invoice total value* property is less than 0.65. If it's the case, we then call a custom document processing model that we've trained with invoices where we usually get a low confidence score for the total field. Finally, we save the extracted data from the invoice into an Excel file.



Use the invoice processing prebuilt model to handle invoices that a custom document processing model hasn't been trained to handle

One way to use the invoice processing prebuilt model is to use it as a fallback model to handle invoices that you haven't trained in your custom document processing model. For example, let's say you built a document processing model, and trained it to extract data from your top 20 invoice providers. You could then use the invoice processing prebuilt model to process all new invoices or lower volume invoices. Here's an example of how you could do it:

This flow is triggered when a new invoice is added to a SharePoint folder. It then calls a custom document processing model to extract its data. Next, we check if the confidence score for the detected collection is less than 0.65. If it's the case, it probably means the provided invoice isn't a good match for the custom model. We then call the prebuilt invoice processing model. Finally, we save the extracted data from the invoice in an Excel file.



See also

- [Training: Extract invoice data with AI Builder's prebuilt model \(module\)](#)
- [Video: How to automate invoice data copy to Excel in 1 minute](#) ↗

Entity extraction prebuilt model

Article • 06/22/2022

The prebuilt entity extraction model recognizes specific data from text that's of interest to your business. The model identifies key elements from text, and then classifies them into predefined categories. This can help to transform unstructured data into structured data that's machine-readable. You can then apply processing to retrieve information, extract facts, and answer questions.

The prebuilt model is ready to use out of the box. For information about customizing your entity extraction to suit your specific needs, see [Overview of the entity extraction custom model](#).

Use in Power Apps

Explore entity extraction

You can try out the entity extraction model before you import it into your flow by using the "try it out" feature.

1. Sign in to [Power Apps](#).
2. In the left pane, select AI Builder > Explore.
3. Under **Get straight to productivity**, select Entity Extraction.
4. In the Entity Extraction window, select Try it out.
5. Select predefined text samples to analyze, or add your own text in the **Or add your own here** box to see how the model analyzes your text.

Use the formula bar

You can integrate your AI Builder entity extraction model in Power Apps Studio by using the formula bar. For more information, see [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Use in Power Automate

If you want to use this prebuilt model in Power Automate, you can find more information in [Use the entity extraction prebuilt model in Power Automate](#).

Supported data format and languages

- Documents can't exceed 5,000 characters.
- Supported languages:
 - English
 - Chinese-Simplified
 - French
 - German
 - Portuguese
 - Italian
 - Spanish

Supported entity types

Entity	Description
Age	Age of a person, place, or thing, extracted as a number
Boolean	Positive or negative responses, extracted as a Boolean
City	City names, extracted as a string
Color	Primary colors and hues on the color spectrum, extracted as a string
Continent	Continent names, extracted as a string
Country or region	Country and region names, extracted as a string
Date and time	Dates, times, days of the week, and months relative to a point in time, extracted as a string
Duration	Lengths of time, extracted as a string in standard TimeSpan format
Email	Email addresses, extracted as a string
Event	Event names, extracted as a string
Language	Language names, extracted as a string
Money	Monetary amounts, extracted as a number
Number	Cardinal numbers in numeric or text form, extracted as a number
Ordinal	Ordinal numbers in numeric or text form, extracted as a number
Organization	Names of organizations, associations, and corporations, extracted as a string

Entity	Description
Percentage	Percentages in numeric or text form, extracted as a number
Person name	A person's partial or full name, extracted as a string
Phone number	Phone numbers in the standard US format, extracted as strings
Speed	Speed, extracted as a number
State	Names and abbreviations for states in the United States, extracted as a string
Street address	Numbered addresses, streets or roads, city, state, ZIP or postal code in the standard US format, extracted as a string
Temperature	Temperature, extracted as a number
URL	Website URLs and links, extracted as a string
Weight	Weight, extracted as a number
Zip code	ZIP codes in the standard US format, extracted as a string

Model output

The model output shows the identified entities and their entity types. For example:

Input text: "Utility costs have increased by 7% at our Boston office"

Model output entities:

Entity	Entity type
7%	Percentage
Boston	City

Next step

[Use the entity extraction prebuilt model in Power Automate](#)

Key phrase extraction prebuilt model

Article • 01/05/2023

The key phrase extraction prebuilt model identifies the main points in a text document. For example, given input text "The food was delicious and there was great service!", the model returns the main talking points: "food" and "great service". This model can extract a list of key phrases from unstructured text documents.

Use in Power Apps

Explore key phrase extraction

You can try out the key phrase extraction model before you import it into your flow by using the "try it out" feature.

1. Sign in to [Power Apps](#).
2. In the left pane, select AI Builder > Explore.
3. Under Get straight to productivity, select Key Phrase Extraction.
4. In the Key Phrase Extraction window, select Try it out.
5. Select predefined text samples to analyze, or add your own text in the Or add your own here box to see how the model analyzes your text.

Use the formula bar

You can integrate your AI Builder key phrase extraction model in Power Apps Studio by using the formula bar. For more information, see [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Use in Power Automate

If you want to use this prebuilt model in Power Automate, you can find more information in [Use the key phrase extraction prebuilt model in Power Automate](#).

Supported language and data format

- **Language:** Afrikaans, Bulgarian, Catalan, Chinese-Simplified, Croatian, Danish, Dutch, English, Estonian, Finnish, French, German, Greek, Hungarian, Italian, Indonesian, Japanese, Korean, Latvian, Norwegian (Bokmål), Polish,

Portuguese (Brazil), Portuguese (Portugal), Romanian, Russian, Spanish, Slovak, Slovenian, Swedish, Turkish

- Documents can't exceed 5,120 characters.

Model output

If text is detected, the key phrase extraction model outputs the following information:

- **Results:** A list of phrases from the document
- **Phrase:** Strings denoting the key talking points in the document text

Limits

The following applies to calls made per environment across the following prebuilt models: language detection, sentiment analysis, and key phrase extraction.

Action	Limit	Renewal period
Calls (per environment)	400	60 seconds

See also

[Training: Key phrase extraction prebuilt model \(module\)](#)

Category classification prebuilt model (preview)

Article • 06/22/2022

[This topic is pre-release documentation and is subject to change.]

The prebuilt category classification model is a ready to use AI model that is configured to classify your text into categories that are useful for a specific business scenario. The first prebuilt category classification AI model is built around customer feedback uses. Check back for additional category classification prebuilt models, or check release plans to see what might be coming.

ⓘ Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.

Use in Power Apps

Explore category classification

You can see the category classification prebuilt model in action without the need to build a flow by using the "try it out" feature.

1. Sign in to [Power Apps](#).
2. In the left pane, select **AI Builder > Explore**.
3. Under **Get straight to productivity**, select **Category classification (preview)**.
4. In the **Category classification** window, select **Try it out**.
5. Select predefined text samples to analyze, or add your own text in the **Or add your own here** box to see how the model analyzes your text.

Use the formula bar

You can integrate your AI Builder category classification model in Power Apps Studio by using the formula bar. For more information, see [Use Power Fx in AI Builder models in Power Apps](#)

Power Apps (preview).

Use in Power Automate

For information about how to use this prebuilt model in Power Automate, see [Use the category classification prebuilt model in Power Automate](#).

Supported data format and languages

- Documents can't exceed 5,000 characters.
- Supported languages:
 - English
 - Chinese-Simplified
 - French
 - German
 - Portuguese
 - Italian
 - Spanish

Supported classification categories

Customer feedback prebuilt model

- Issues
- Compliment
- Customer Service
- Documentation
- Price & Billing
- Staff

Next step

[Use the category classification prebuilt model in Power Automate](#)

See also

[Overview of the category classification custom model](#)

Text translation prebuilt model

Article • 03/05/2022

The text translation prebuilt model translates your text data in real time across more than 60 languages. This prebuilt model could help remove language barriers within your company. The text translation model can also detect the language of the text data you want to translate.

Use in Power Automate

If you want to use this prebuilt model in Power Automate, you can find more information here: [Use text translation model in Power Automate](#).

Supported language and data format

- Test can't exceed 10,000 characters.
- For information on language support, see [Language and region support for the text translation API](#).

Model output

If text is detected and the target language defined, the text translation model will output the following information:

- **Text:** Strings containing the translated text.
- **Detected language:** Script version of the language code (ex.: "en", "fr", "zh_chs", "ru") detected in the source text. The model will not detect the source language if it was specified by the user.

ID reader prebuilt model

Article • 01/05/2023

You can use the identity document (ID) reader prebuilt model to extract information from passports and US driver licenses, social security, and green cards. The model will extract information such as the person's first name, date of birth, or gender.

Images such as scans or pictures of the identity documents are deleted once processed by the model.

ⓘ Note

- The design and format of identity documents vary widely. AI Builder is constantly improving the accuracy of the ID reader prebuilt model, but it's possible there could be inaccurate or missing information in some cases. It's a good idea to verify that the output is as you expect.
- The ID reader prebuilt model only supports Latin character extraction at this time.

Licensing requirements

AI Builder is licensed as an add-on to your Power Apps or Power Automate licenses. For information about license capacity, pricing, and restrictions, see [AI Builder licensing](#).

Role requirements

Users need to have the Basic User role to consume the ID reader prebuilt model.

Use in Power Automate

If you want to use this model in Power Automate, you can find more information in [Use the ID reader prebuilt model in Power Automate](#).

Supported language, format, and size

The images you can process with the ID reader prebuilt model must have these characteristics:

- Format:
 - jpg
 - png
 - pdf
- Size: 20 MB maximum (use small images for fastest processing time)
- For PDF documents, only the first 2,000 pages are processed.

Model output

If a valid identity document is detected, the model will try to locate and extract the following properties.

Property	Note
First name	N/A
Last name	N/A
Gender	N/A
Date of birth	N/A
Place of birth	N/A
Region	US driver license only
Country	N/A
Nationality	Passport only
Street address	US driver license only
Identity document number	N/A
Date of issue	N/A
Identity document expiration date	N/A
Category code	US green card only

Limits

Action	Limit	Renewal period
Identity document reader calls (per environment)	24	60 seconds

See also

- [Use the ID reader prebuilt model in Power Automate](#)
- [Training: Extract data from passports and US driver licenses \(module\)](#)

Business card prebuilt model

Article • 03/13/2023

You can use the business card prebuilt model to extract information from business card images. If it detects a business card in the image, the AI model extracts information such as the person's name, job title, address, email, company, and phone numbers.

ⓘ Note

- The design and format of business cards varies widely. AI Builder is constantly improving the accuracy of the business card AI model, but it's possible there could be inaccurate or missing information in some cases. It's a good idea to verify that the output is as you expect.
- The prebuilt business card reader only supports English and Japanese languages extraction at this time. Check back to see what languages are supported going forward.

Licensing requirements

AI Builder is licensed as an add-on to your Power Apps or Power Automate licenses. For information about license capacity, pricing, and restrictions, see [AI Builder licensing](#).

Role requirements

Users need to have the Basic User role to consume the business card reader.

Use in Power Apps

If you want to use this prebuilt model in Power Apps, you use the business card reader component. More information: [Use the business card reader component in canvas apps](#) and [Use the business card reader component in model-driven apps](#)

Use in Power Automate

If you want to use this prebuilt model in Power Automate, you can find more information in [Use the business card reader prebuilt model in Power Automate](#).

Supported language, format, and size

The images you can process with the business card model must have these characteristics:

- Languages: English and Japanese
- Format:
 - JPG
 - PNG
 - BMP
 - PDF
- Size: 50 MB maximum

Model output

If a business card is detected, the business card model will try to locate and extract the following properties.

Property	Definition
AddressCity	The city address
AddressCountry	The country address
AddressPostalCode	The postal code address
AddressPostOfficeBox	The post office box address
AddressState	The state address
AddressStreet	The street address
BusinessPhone	The first phone or fax number
CompanyName	The company name
Department	The organization department found
Email	The contact email found in the business card, if any
Fax	The third phone or fax number
FirstName	The contact's first name
FullAddress	The contact's full address
FullName	The contact's full name

Property	Definition
JobTitle	The contact's job title
LastName	The contact's last name
MobilePhone	The second phone or fax number
OriginalImage	The original image before processing
Website	The website

Limits

Action	Limit	Renewal period
Business card reader calls (per environment)	24	60 seconds

Receipt processing prebuilt model

Article • 03/13/2023

Receipt processing is a prebuilt model that uses state-of-the-art optical character recognition (OCR) to detect printed and handwritten text and extract key information from receipts.

Use in Power Apps

The receipt processing prebuilt model is available in Power Apps by using the receipt processor component. For more information, see [Use the receipt processor component in Power Apps](#).

Use in Power Automate

For information on how to use the receipt processing prebuilt model in Power Automate, see [Use the receipt processing prebuilt model in Power Automate](#).

Supported languages, markets, and files

The following languages are supported: Dutch (Netherlands), English (Australia), English (Canada), English (India), English (UAE), English (UK), English (United States), French (Canada), French (France), German (Germany), Italian (Italy), Portuguese (Brazil), Portuguese (Portugal), Spanish (Spain).

To get the best results, provide one clear photo or scan per receipt.

- The image format must be JPEG, PNG, or PDF.
- The file size must not exceed 20 MB.
- The image dimensions must be between 50 x 50 pixels and 10,000 x 10,000 pixels.
- PDF dimensions must be at most 17 x 17 inches, which is the equivalent of the Legal or A3 paper sizes or smaller.
- For PDF documents, only the first 2,000 pages are processed.

Model output

If a receipt is detected, the receipt processing model will output the following information:

Property	Definition
MerchantName	Merchant name
MerchantAddress	Merchant address
MerchantPhone	Merchant phone number
TransactionDate	Transaction date
TransactionTime	Transaction time
PurchasedItems	<p>The list of purchased items</p> <ul style="list-style-type: none"> • Name: Name of the purchased item • Price: Price of the purchased item • Quantity: Quantity of the purchased item • TotalPrice: Total price of the purchased item
Subtotal	Subtotal
Tax	Tax
Tip	Tip
Total	Total
DetectedText	The list of all recognized lines of text on the receipt

Limits

The following applies to calls made per environment across document processing models including prebuilt models: receipt processing and invoice processing.

Action	Limit	Renewal period
Calls (per environment)	360	60 seconds

See also

[Training: Process receipts with AI Builder \(module\)](#)

Overview of the prediction model

Article • 08/05/2022

AI Builder prediction models analyze patterns in historical data that you provide. Prediction models learn to associate those patterns with outcomes. Then, we use the power of AI to detect learned patterns in new data, and use them to predict future outcomes.

Use the prediction model to explore business questions that can be answered as one of the following ways:

- From two available options (binary).
- From multiple possible outcomes.
- Where the answer is a number.

Binary prediction

Binary prediction is when the question asked has two possible answers. For example: yes/no, true/false, on-time/late, go/no-go, and so on. Examples of questions that use binary prediction include:

- Is an applicant eligible for membership?
- Is this transaction likely to be fraudulent?
- Is a customer a good candidate for a marketing campaign?
- Is an account likely to pay their invoices on time?

Multiple outcome prediction

Multiple outcome prediction is when the question can be answered from a list of more than two possible outcomes. Examples of multiple outcome prediction include:

- Will a shipment arrive early, on-time, late, or very late?
- Which product would a customer be interested in?

Numerical prediction

Numerical prediction is when the question is answered with a number. Examples of numerical prediction include:

- How many days for a shipment to arrive?

- How many calls should an agent handle in a day?
- How many items do we need to keep in inventory?
- How many leads should a sales team convert in a month?

See also

[Feature availability by region](#)

[Prediction model prerequisites](#)

Prediction model prerequisites

Article • 07/08/2022

For information about AI Builder requirements that aren't specific to this AI model, go to [AI model prerequisites](#).

What skills do I need?

- The person creating the prediction model should know enough about the business in question to understand the meaning of a dataset.
- Because you don't need coding skills to create an AI model in AI Builder, you don't need to be a developer or data scientist to use the AI Builder prediction model. If you aren't familiar with a certain concept, select the **View documentation** or **Help** links in AI Builder.

What data do I need?

- Your data must be in [Microsoft Dataverse](#).
- Make sure your administrator has assigned you a security role with Read privilege over your data.
- You need at least 10 rows of historical outcome for each class of the **Label** data column to train a prediction model.
- The minimum for training is 50 rows, but for best results you should have at least 1,000 rows.

Work with sample data

To help you get started quickly with prediction, AI Builder provides sample data that you can use to get started. For more information, go to [Use sample data to do prediction](#).

Next step

[Data preparation](#)

See also

AI model prerequisites

Data preparation

Article • 03/05/2022

Before you create your prediction model, you'll want to make sure your data is in Microsoft Dataverse and that it's in the correct format.

Create your custom table

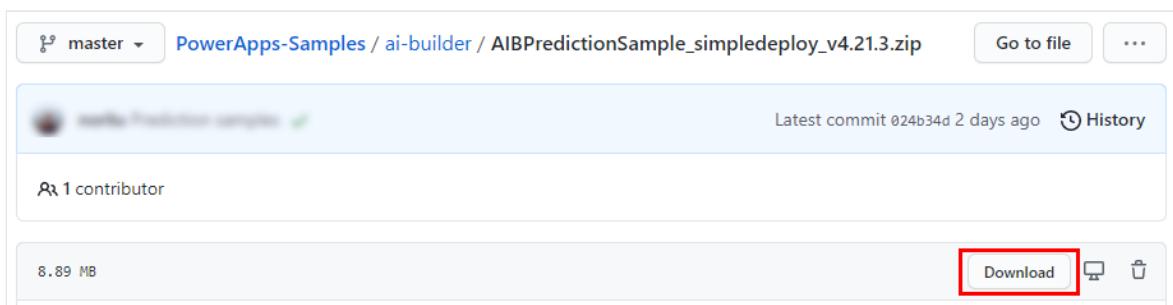
Do you have data that you want to import into Dataverse for training in AI Builder? First, you have to create a table. In this example, we'll provide a solution that has predefined custom tables. To use your own data, [create a custom table](#) and use it instead of the example used here.

ⓘ Note

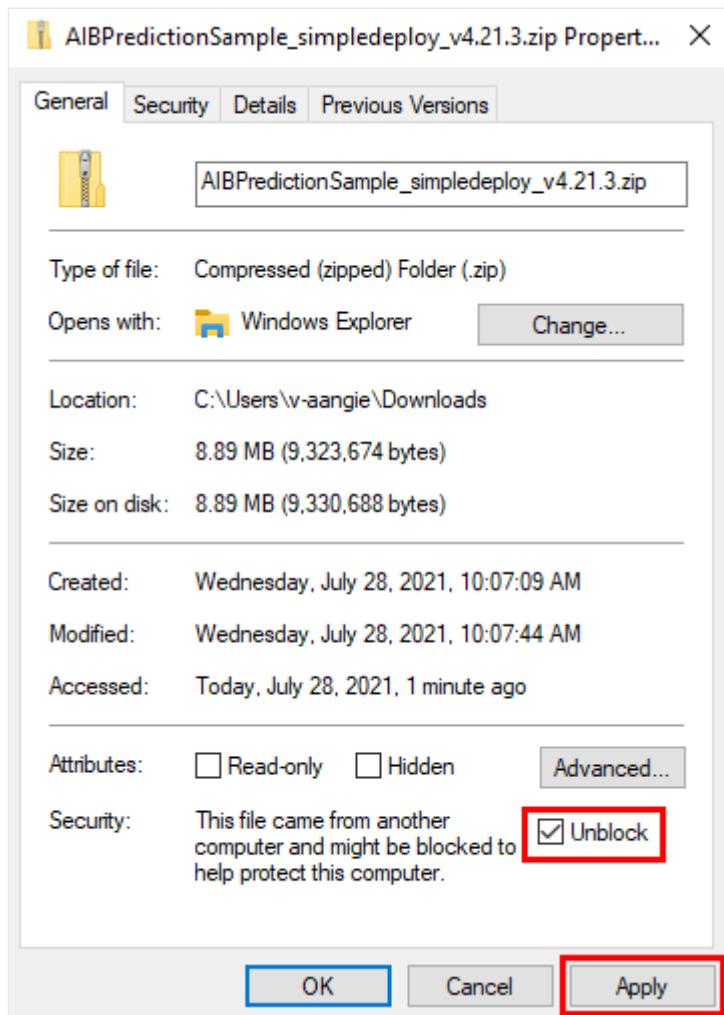
For best results, use a dataset that is less than 1.5 GB in size. Otherwise, AI Builder will use only 1.5 GB of your data to train and predict. Because you can't control which data that exceeds the 1.5 GB limit won't be used, you should optimize your data to stay under 1.5 GB.

Sample dataset for prediction model

1. Download the AI Builder sample dataset package:
 - a. Select [AIBPredictionSample_simpledeploy_v4.21.3.zip](#).
 - b. Select the **Download** button.



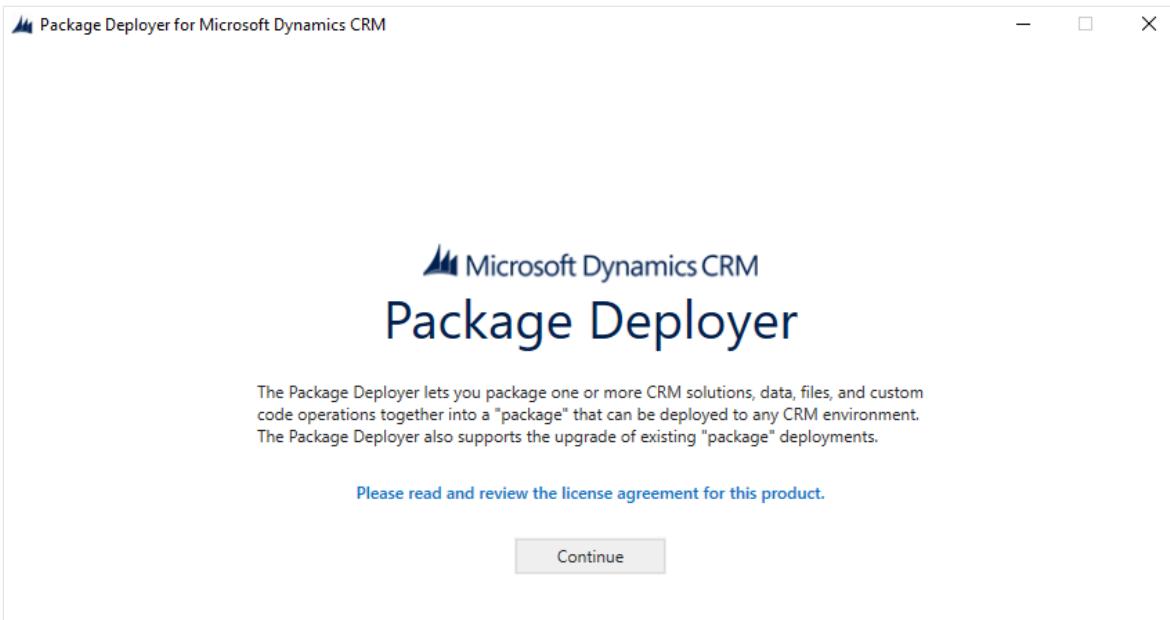
2. Ensure that the file isn't blocked after you download. To do this:
 - a. In the Downloads folder, find the downloaded zip file, right-click, and then select **Properties**.
 - b. On the General tab, select the **Unblock** checkbox, and then select **Apply**.



3. Extract the .zip file, and look for **PackageDeployer.exe** in the extracted folder.

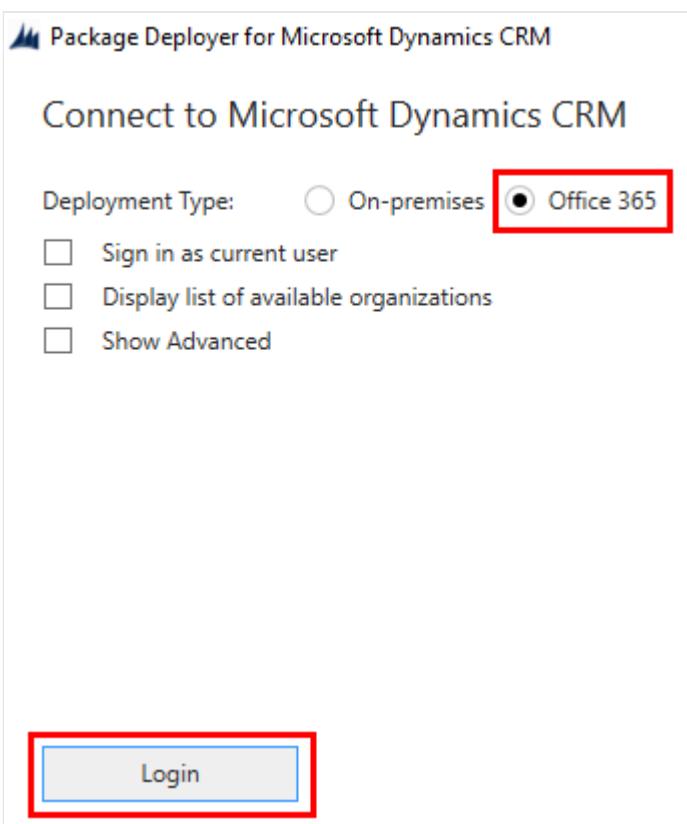
Name	Date modified	Type	Size
Microsoft.Xrm.Tooling.PackageDeployment.CrmPackageCore....	4/10/2021 12:05 AM	Application exten...	43 KB
Microsoft.Xrm.Tooling.PackageDeployment.CrmPackageCore....	4/10/2021 12:03 AM	XML Document	197 KB
Microsoft.Xrm.Tooling.PackageDeployment.CrmPackageExten...	4/10/2021 12:06 AM	Application exten...	35 KB
Microsoft.Xrm.Tooling.PackageDeployment.CrmPackageExten...	4/10/2021 12:02 AM	XML Document	52 KB
Microsoft.Xrm.Tooling.Ui.Styles.dll	4/10/2021 12:05 AM	Application exten...	151 KB
Newtonsoft.Json.dll	4/10/2021 12:14 AM	Application exten...	648 KB
Other Redistributable.txt	4/9/2021 11:59 PM	Text Document	1 KB
PackageDeployer.exe	4/10/2021 12:05 AM	Application	244 KB
PackageDeployer.exe.config	4/9/2021 11:59 PM	CONFIG File	9 KB
pacTelemetryUpload.exe	4/10/2021 12:05 AM	Application	27 KB
System.Diagnostics.DiagnosticSource.dll	5/15/2018 1:29 PM	Application exten...	58 KB

4. Run **PackageDeployer.exe**. The following screen will appear.



5. Select **Continue**.

6. Select **Office 365**, and then select **Login**.



7. Enter the credentials that you use to sign in to the Power Apps maker portal, and then select **Next**.

Sign in to your account

X



Sign in

Email, phone, or Skype

No account? [Create one!](#)

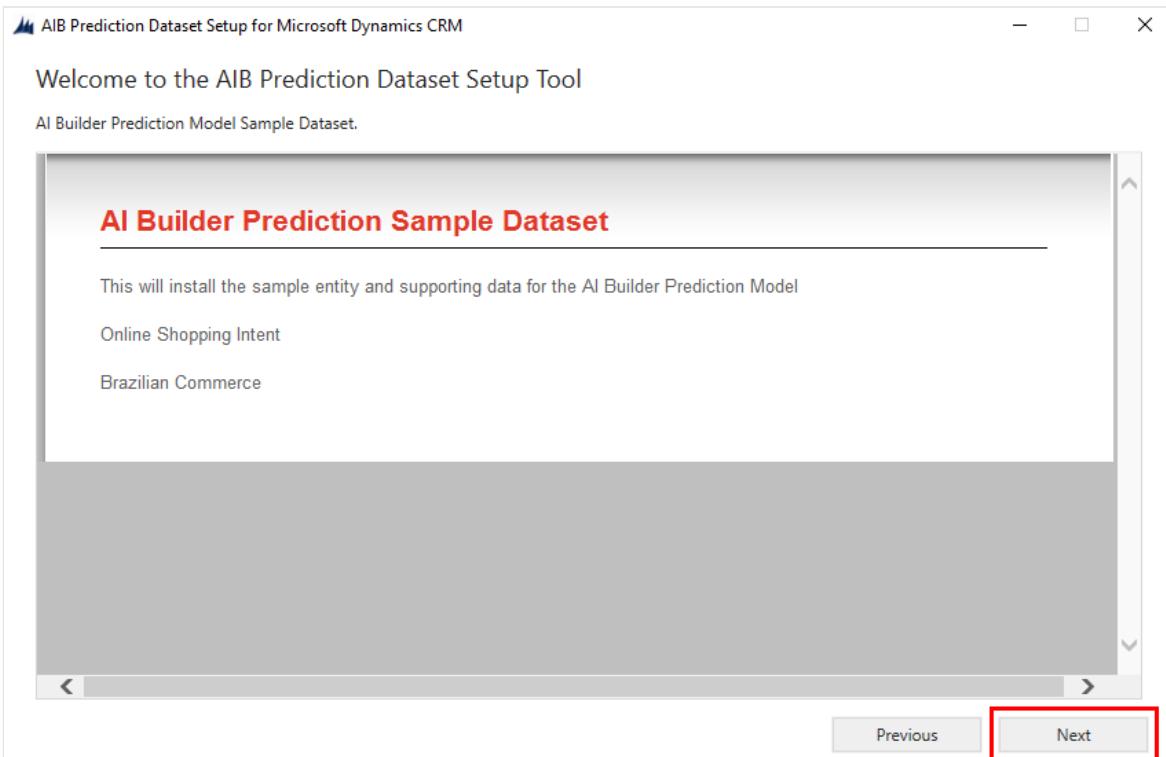
[Can't access your account?](#)

Next

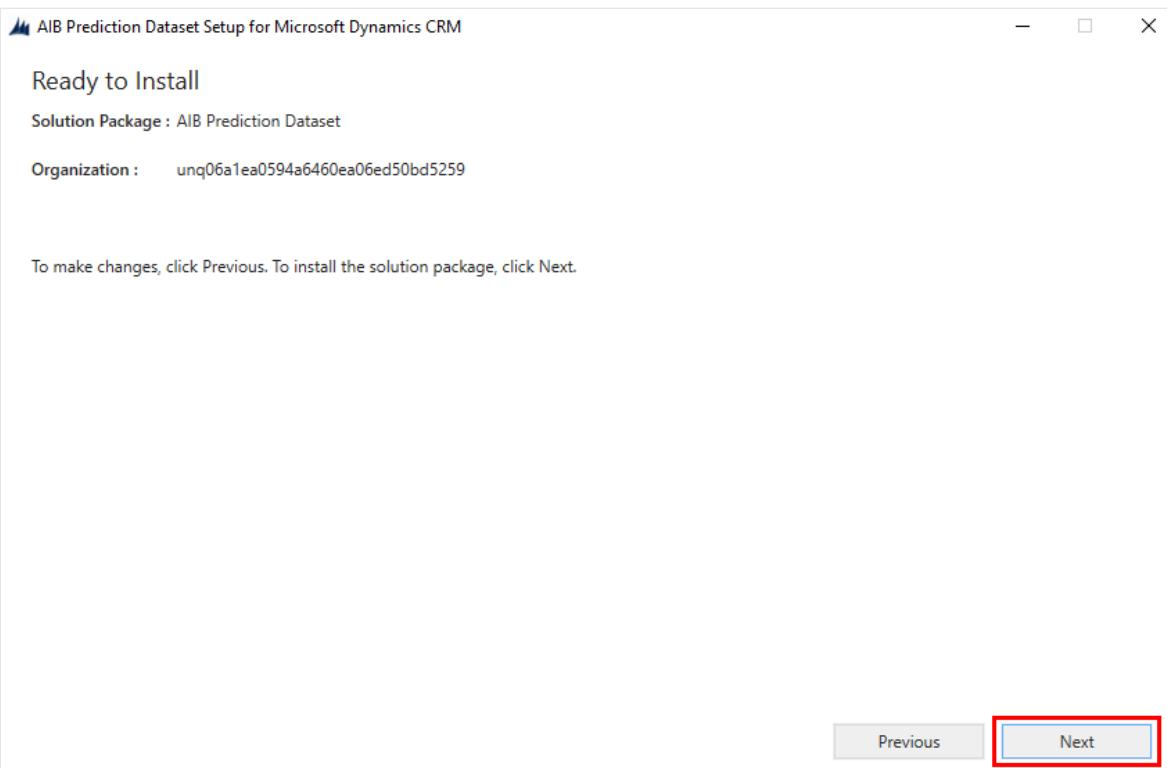


Sign-in options

8. If the sign-in is successful, you'll see the Welcome screen. Read the message, and then select **Next**.

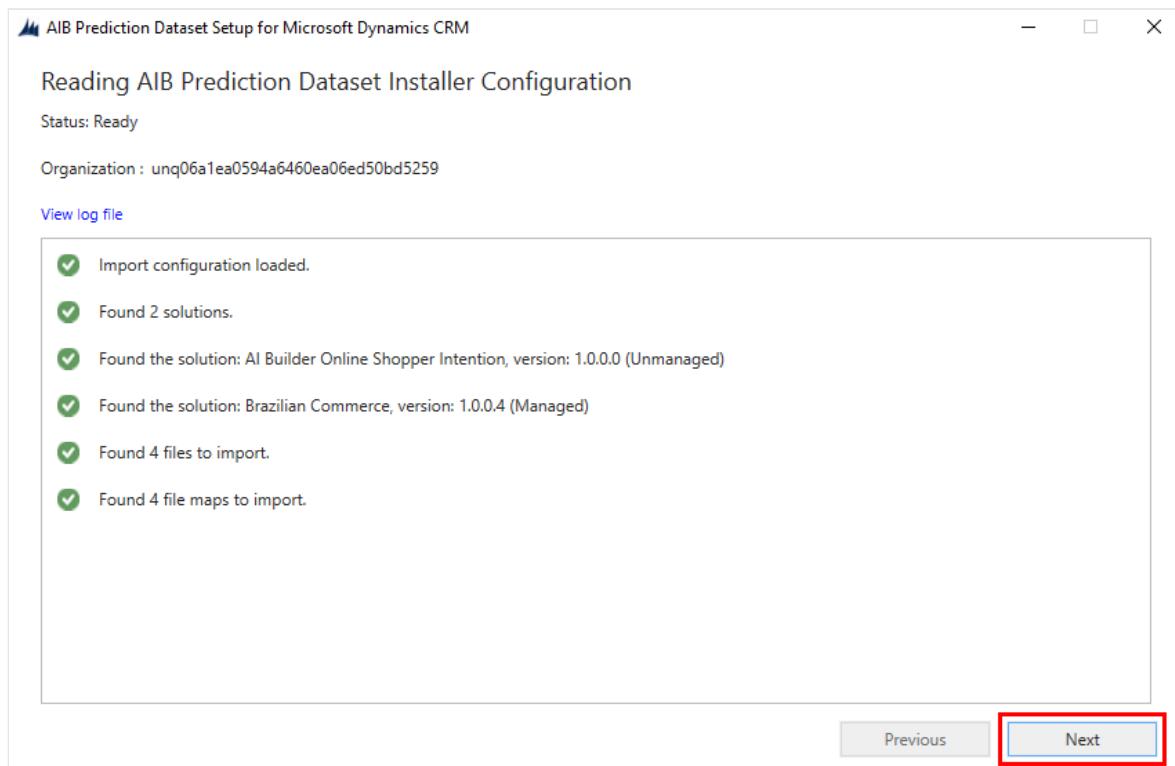


9. On the Ready to Install screen, make sure you're installing the solution in the correct environment and then select **Next**.

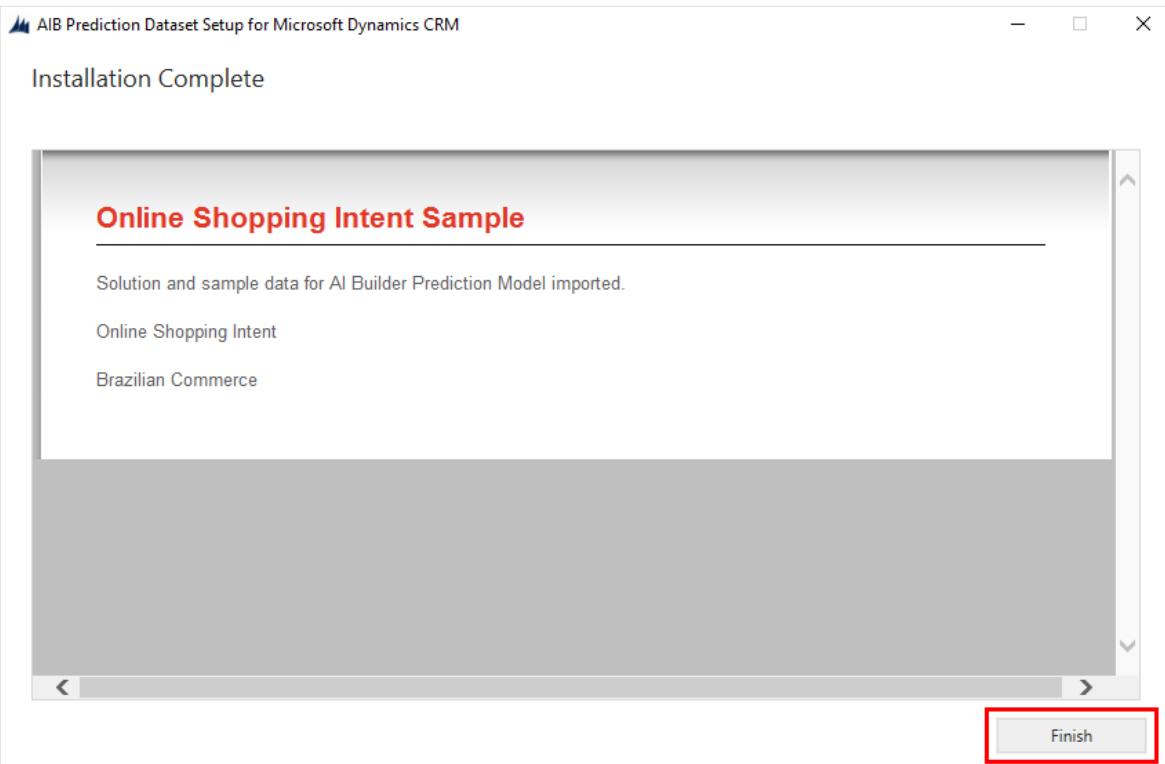


10. On the **Reading AIB Prediction Dataset Installer Configuration** screen, read the summary information for the data and solutions being imported and then select **Next**.

It will take a few minutes to import the data. As each step is completed successfully, you'll see a green circle with a check mark next to the step.



11. On the **Installation Complete** screen, select **Finish**.



How you can use the solutions you installed

The sample dataset installs two solutions in your environment. It also installs sample data for the included entities:

- **Brazilian Commerce:** Use for predicting multiple outcomes. Select **BC Orders** as the table, and **Delivery Timeliness** as the column when you pick what you want to predict.
- **AI Builder Online Shopper Intention:** Use for binary prediction and numerical prediction.
 - Select **Online Shopper Intentions** as the table, and **Revenue (Label)** as the column if you want to try out binary prediction.
 - Select **Online Shopper Intentions** as the table. Also select **ExitRates** or **BounceRates** as the column if you want to try out numerical prediction.

If you need help with creating a prediction model, follow the instructions in [Create a prediction model](#).

Solutions						
Display name	Name	Created ↓	Version	Managed externally?	Solution check	
Brazilian Commerce	... BrazilianCommerce	4/28/2021	1.0.0.4	□	Checked by publisher	
AI Builder Online Shopper Intention	... AIBuilderOnlineShop...	4/28/2021	1.0.0.0	□	Hasn't been run	

You're now ready to go to the next step.

Next step

[Create a prediction model](#)

Create a prediction model

Article • 06/23/2022

This example creates a Power Apps prediction AI model that uses the Online Shopper Intention table in Microsoft Dataverse. To get this sample data into your Microsoft Power Platform environment, enable the **Deploy sample apps and data** setting when you create an environment as described in [Build a model in AI Builder](#). Or, follow the more detailed instructions in [Data preparation](#). After your sample data is in Dataverse, follow these steps to create your model.

1. Sign in to [Power Apps](#), and then select **AI Builder > Explore**.
2. Select **Prediction**. Enter a name for your model, and then select **Create**.

Select your historical outcome

Think of the prediction you want AI Builder to make. For example, for the question "Will this customer churn?", think about questions like these:

- Where is the table that contains information about customer churn?
- Is there a column there that specifically states whether the customer has churned?
- Are there unknowns in a column that might cause uncertainty?

Use this information to make your selections. Working with provided sample data, the question is "did this user who interacted with my online store make a purchase?" If they did, there should be revenue for that customer. Therefore, whether there's revenue for this customer should be the historical outcome. Wherever this information is empty is where AI Builder can help you make a prediction.

1. In the **Table** drop-down menu, select the table that contains the data and the outcome you want to predict. For the sample data, select **Online shopper intention**.
2. In the **Column** drop-down menu, select the column that contains the outcome. For the sample data, select **Revenue (Label)**. Or, if you want to try out predicting a number, select **ExitRates**.
3. If you selected an option set that contains two or more outcomes, consider mapping it to "Yes" or "No" because you want to predict whether something will happen.

4. If you want to predict multiple outcomes, use the Brazilian e-commerce dataset in the sample, and select **BC Order** in the **Table** drop-down menu and **Delivery Timelines** in the **Column** drop-down menu.

ⓘ Note

AI Builder supports these data types for the outcome column:

- Yes/No
- Choices
- Whole number
- Decimal number
- Floating point number
- Currency

Select the data columns to train your model

After you select the **Table** and **Column** and map your outcome, you can make changes to the data columns used to train the model. By default, all relevant columns are selected. You can deselect columns that might contribute to a less accurate model. If you don't know what to do here, don't worry. AI Builder will try to find columns that provide the best model possible. For the sample data, just leave everything as is and select **Next**.

Data column selection considerations

The most important thing to consider here is whether a column that isn't your historical outcome column is indirectly determined by the outcome.

Let's say you want to predict whether a shipment is going to be delayed. You might have the actual delivered date in your data. That date is only present after the order is delivered. So, if you include this column, the model will have close to 100 percent accuracy. The orders that you want to predict won't have been delivered yet, and won't have the delivered date column populated. So, you should deselect columns like this before training. In machine learning, this is called *target leakage* or *data leakage*. AI Builder tries to filter columns that are "too good to be true," but you should still check them.

ⓘ Note

When you're selecting data fields, some data types—like Image, which can't be used as input to train the model—are shown. In addition, system columns like Created On are excluded by default.

Use data from related tables

If you have related tables that might improve the performance of the prediction, you can include those as well. As you did when you wanted to predict whether a customer will churn, you should include additional information that might be in a separate table. AI Builder supports many-to-one relationships at this time.

Filter your data

After you select data columns for training, you can filter on your data. Your tables will contain all rows. However, you might want to concentrate on training and predicting on a subset of rows. If you know that there's irrelevant data within the same table you're using to train a model, you can use this step to filter it.

For example, if you apply a filter to look at only the US region, the model will train on rows where the outcome is known only for the US region. When this model is trained, it will only make a prediction for rows where the outcome is not known for only the US region.

The filtering experience is the same as in the Power Apps view editor. Start by adding:

- A row, which contains a single filter condition.
- A group, which allows you to nest your filter conditions.
- A related table, which allows you to create a filter condition on a related table.

Select the column, the operator, and the value that represents a filter condition. You can use the checkboxes to group rows or to bulk-delete rows.

Next step

[Train and publish your prediction model](#)

Train and publish your prediction model

Article • 03/05/2022

Before you can use your prediction model, you have to train it to perform the way you want. After you train your model, publish it to make it available.

Train

After you've selected your data columns, select **Next** to view a summary of training information, and then select **Train** to train your model.

ⓘ Note

Training takes time, so you can stay on the page and wait, or you can close the page and come back later.

Publish

After the model is trained, you can [evaluate](#) its performance. Then [publish](#) your model to start running predictions. After that, predictions run daily.

See also

[Train your model in AI Builder](#)

[Publish your model in AI Builder](#)

Use sample data to do prediction

Article • 06/23/2022

To explore the possibilities of prediction in AI Builder, you can get started by building and training a prediction model that uses sample data provided by Microsoft.

Note

This sample data is added to your Microsoft Power Platform environment automatically if you enable the **Deploy sample apps and data** setting when you create your database.

Get the sample data

You can [configure your new environment](#) to deploy sample data automatically when you create it. Or, you can download the AI Builder samples provided by Microsoft from <https://github.com/microsoft/PowerApps-Samples/tree/master/ai-builder>. Follow the instructions in the [Data preparation](#) topic to import the sample data into your Power Platform environment, and then follow the instructions in [Create a prediction model](#) to create an AI model that uses this sample data.

Prediction model performance

Article • 03/05/2022

After each training, AI Builder uses the test data set to evaluate the quality and fit of the new model. A summary page for your model shows your model training result. These results are expressed as a performance grade of A, B, C, or D.

Measuring performance

Performance grade

After each training, AI Builder shows a grade to help you evaluate your model's accuracy. The decision about whether your model is ready to publish is one you have to make based on your unique needs and circumstances. AI Builder provides the following performance grades to help you make that judgment call.

How to interpret each grade

Grade	Guidance
A	It might still be possible to improve the model, but this is the best grade you can get.
B	The model is correct in a lot of the cases. Can it be improved? That depends on your unique circumstances, data, and requirements.
C	The model is doing slightly better than a random guess. It might be acceptable for some applications, but in most cases, this is a model that you'd continue to tweak and improve.
D	Something's wrong. Your model is either performing worse than we'd expect a random guess to perform (underfit model). Or, it's performing so well (at or near 100%) that you've probably got a data column that is directly correlated to the result (overfit model)

- More information about [underfit models](#)
- More information about [overfit models](#)

Accuracy range varies depending on your data

If you are predicting 2 or more outcomes, the actual accuracy rates that correspond to the above grades can vary depending on the data distribution of your historical data.

The difference accounts for the fact that the improvement relative to your baseline rate changes when you move that baseline.

Let's say your model predicts whether a shipment will arrive on time. If your historical on-time rate is 80 percent, a performance score of 92 would correspond to a B grade. But, if your historical on-time rate is only 50 percent, 92 would correspond to an A grade. That's because 92 is a much better improvement over 50 percent than it is over 80 percent, and you'd expect a random guess to be close to those percentages.

Binary historical data example

This example shows the accuracy ranges for each grade when the historical data contains different on-time rates for a binary prediction.

Grade	Accuracy range for historical 25% on-time rate	Accuracy range for historical 50% on-time rate	Accuracy range for historical 80% on-time rate	Accuracy range for historical 95% on-time rate
A	92.5 – <99.3%	90 – 98%	93 – <99%	98.1 – <99.8%
B	81.3 – <92.5%	75 – <90%	84 – <93%	95.3 – <98.1%
C	66.3 – <81.3%	55 – <75%	71 – <84%	91.5 – <95.3%
D	<66.3% or ≥99.3%	<55% or ≥98%	<71% or ≥99%	<91.5% or ≥99.8%

Multiple outcome historical data example

Accuracy rates that correspond to each grade can also vary when you're predicting more than 2 outcomes. Let's say your model predicts more than two options for delivery: early, on time or late.

The accuracy ranges for each grade changes when your historical on-time rates change.

Grade	Early (33.3%)	Early (20%)	Early (10%)
	On time (33.3%)	On time (40%)	On time (80%)
	Late (33.4%)	Late (40%)	Late (10%)
A	86.7 – <98.7%	87.2 – <98.7%	93.2 – <99.3%
B	66.7 – <86.7%	68.0 – <87.2%	83.0 – <93.2%
C	40.0 – <66.7%	42.4 – <68.0%	69.4 – <83.0%

Grade	Early (33.3%)	Early (20%)	Early (10%)
D	33.3 – <40.0%	36.0 – <42.4%	66.0 – <69.4%

Numerical prediction example

For numerical prediction, AI Builder uses the R-squared statistical measure to calculate your models accuracy grade. The following table shows the grades that correspond to each grade:

Grade	R-squared
A	85% - <99%
B	60% - <85%
C	10% - <60%
D	≥99% or <10%

Performance details

For training details, select **See details** on the model's grade box. On the **Performance** tab, the following information is available:

 **Note**

For information about additional features planned for this area, see [release plans](#).

- Accuracy score
- R-squared

Accuracy score

AI Builder calculates the accuracy score for your model based on prediction result of the test dataset. Before training, AI Builder separates your dataset into separate training data and testing data sets. And after training, AI Builder applies your AI model to the testing dataset, and then calculates your accuracy score. For example: if your test dataset has 200 rows, and AI Builder correctly predicts 192 of them, AI Builder shows an accuracy score of 96 percent.

For more information, see [Evaluate your model](#).

R -squared

For numerical prediction, AI Builder calculates an r-squared score after each training. This score measures your model's 'goodness of fit', and is used to determine your model's performance grade.

Let's say you're predicting the number of days to fulfill, ship, and deliver an order. The model predicts a set of numbers. The r-squared value is based on the distances between predicted values and actual values in your training data. This is expressed as a number between 0 – 100%, with higher values indicating the predicted value is closer to the real value. Typically, a higher score means the model performs better. Remember though, that perfect or near-perfect scores ([overfit models](#)) are usually indicative of a problem with your training data.

On the **Summary** tab, the following performance information is available:

- Training date
- Data source
- Historical outcome
- Table list used to do prediction.

Improve your prediction model performance

After you've trained and evaluated your model, it's time to tweak your model to improve its performance. Here are some things you can try to help improve your model's predictive power.

Review errors and issues

- If there are any errors after you finish training, fix them and retrain the model.
- If there are no errors, check the training details. Try to address as many issues as possible, and then retrain the model.

Review top influencers

After each training, a list of top influencers appears on the model details page. Each column used in the training has a score to represent its influence on the training. These scores combine to equal 100 percent.

This helps show whether your model is trained as you expect. For example, if you want to predict online shoppers' intention and you're expecting Age, Product as the most influential column, you should see that in the most influential column list in model

details page. If not, it might indicate that the training result is not as expected. In this case, you can either deselect the irrelevant or misleading columns and retrain the model, or check your training issues to see further details.

Add more data

The minimum requirement for training data is 50 rows, but this doesn't mean 50 data rows will train a highly predictive model. Try to provide 1,000 or more data rows, correctly labeled, with a realistic distribution between options.

Check your data distribution

For example, if you're using two option labels of *Yes* or *No*, and most of your data rows only have *Yes* in this column, it's hard for your model to learn from this data. Try to have a distribution of options in your data that roughly reflects the distribution of the options you might expect to see. For example, if you're looking at data columns for *cat_owner* and *dog_owner*, use a data distribution somewhere around 50 percent. If you're looking at fraudulent transactions, use a more imbalanced distribution—perhaps 95 percent to 5 percent. Look to industry standards for this type of information if you don't know what to expect.

Add more column

For example, you want to predict which customers are more likely to return and buy your products. You can add more columns to make the training data richer. For example:

- How do they rate the product?
- How much do they use the product?
- Are they an existing customer?

Narrow selected columns to relevant information

You might already have a lot of correctly labeled training data, with many data columns. Then why might the model still not perform well? It could be that you're selecting columns that lead to unwanted bias. Make sure all the columns you select are relevant to influence what you want to predict. Deselect irrelevant or misleading columns.

Validate data

- Make sure the data columns don't have high rate of missing values (greater than 99 percent). Populate the missing values with default data or remove the data column from the model training.
- If a data column has a high correlation with prediction outcome, remove the data column from the model training.

Next step

[Use your prediction model in Power Apps](#)

Prediction model training errors and warnings

Article • 03/05/2022

While training the prediction model, you might come across the messages in this article that AI Builder might report. Messages are either *errors* or *warnings*. Each is represented by an icon.

Message	Icon
Error	
Warning	

When an error occurs, you can't continue until you resolve it. If the system is unable to correct a problem, it will show you an error.

Warnings are messages reported as informational. They don't stop you from proceeding. They warn you of possible performance issues when training the model.

Errors	See more
Fix these problems and train your model again. Learn more about these errors	
The model needs at least 10 historical outcome rows of each outcome value to train. Add data or select another table.	
The model might produce better performance with optimum rows of 1000 or more to train the model. Online Shopper Intention has 949 rows. Add data for better model performance.	
Online Shopper Intention. Online Shopper Intention > Status Reason has a single value and might not contribute to training the model.	

Error: No AI Builder license

You need an AI Builder license to use this feature. Start or extend a trial or contact your admin to upgrade.

Cause

You either don't have an active [AI Builder license](#), or the existing trial has expired.

Resolution

To use AI Builder models, make sure you have the AI Builder license assigned.

Error: Insufficient number of rows to train

The model needs at least 50 rows to train. <TableName> has only <ActualValue> rows. Add data or select another table.

 The model needs at least 50 rows to train. Account has only 0 rows. Add data or select another table.

Model summary

Review your model's details below. If everything looks good, select Train. [Learn more about training](#)

Model type
Prediction

Cause

The table that you've selected as the historical outcome doesn't have enough rows in it for the model to train itself to predict future outcomes.

Resolution

Add a minimum of 50 rows to the table. Use a minimum of 1,000 rows for better prediction results. More information: [Prediction model prerequisites](#)

Error: Insufficient historical outcome rows to train

The model needs at least 10 historical outcome rows of each outcome value to train. Add data or select another table.

Errors	See more
<p>Fix these problems and train your model again. Learn more about these errors</p> <p>✖ The model needs at least 10 historical outcome rows of each outcome value to train. Add data or select another table.</p>	

Cause

The column that you selected to let AI Builder study the historical outcome doesn't have enough rows for each possible outcome. For example, in a Boolean field where the possible outcomes can be either true or false, there should be a minimum 10 rows of historical data where the outcome is set to **True** and another 10 rows set to **False**.

Resolution

Make sure you have 10 rows of each possible outcome value that you would like the model to predict. More information: [Prediction model prerequisites](#)

Warning: Add data to improve model performance

The model might produce better performance with optimum rows of 1,000 or more to train the model. Online Shopper Intention has <Actualcount> rows. Add data for better model performance.

ⓘ The model might produce better performance with optimum rows of 1000 or more to train the model. Online Shopper Intention has 949 rows. Add data for better model performance.
--

Cause

The AI model found that the number of rows might not result in optimal model performance.

Resolution

We recommend 1,000 rows or more of historical data with outcomes to predict outcomes with a high level of accuracy. However, 50 rows is the minimum limit to process the prediction model.

Warning: Column might be dropped from training model

<TableName>. <ColumnName> might get dropped from training as it has a single value and does not contribute to training the model.

- ⓘ Online Shopper Intention. Online Shopper Intention
 - > Status Reason has a single value and might not contribute to training the model.

Cause

The AI model processes data in the columns related to the outcome that will influence the prediction. Of the various columns selected, it found that the specified column had only a single value across all rows in the table. Because of this, it won't impact the prediction and won't help training the model.

Resolution

Make sure that all columns selected as being related to the outcome column have multiple values in the column. This will help with the training of the model.

Warning: High ratio of missing values

<TableName>. <ColumnName> has a high ratio of missing values, greater than <ThresholdValue> percentage and might not contribute to train the model.

Cause

The AI model processes the data in the columns related to the outcome that will influence the prediction. Of the various columns selected, the model found that the specified column had data in few rows in the table. Since the data won't impact the prediction, it won't help training the model.

Resolution

Make sure the columns that are selected as being related to the outcome have data for them across most of the rows in historical data.

Warning: High percent correlation to the outcome column

<TableName>. <ColumnName> has <ThresholdValue> percent correlation <CorrelationName> with <OutcomeAttributeName> and model might suspect to cause target leak.

Cause

The AI model processes the data in the columns related to the outcome that will influence the prediction. Of the various columns selected, it found that the specified column has a high correlation with the outcome column, which might impact the prediction result. Because of this, it won't be included in training the model.

Resolution

Make sure the columns selected as being related to the outcome don't have a high correlation with the outcome column for a fair prediction.

See also

[Common issues and resolutions for AI Builder](#)

Use your prediction model

Article • 01/04/2023

After your model is trained and published, it's ready for you to use.

Run your prediction model

To run your prediction model, go to the model settings page and select **Run now**.

Schedule retrain and run (preview)

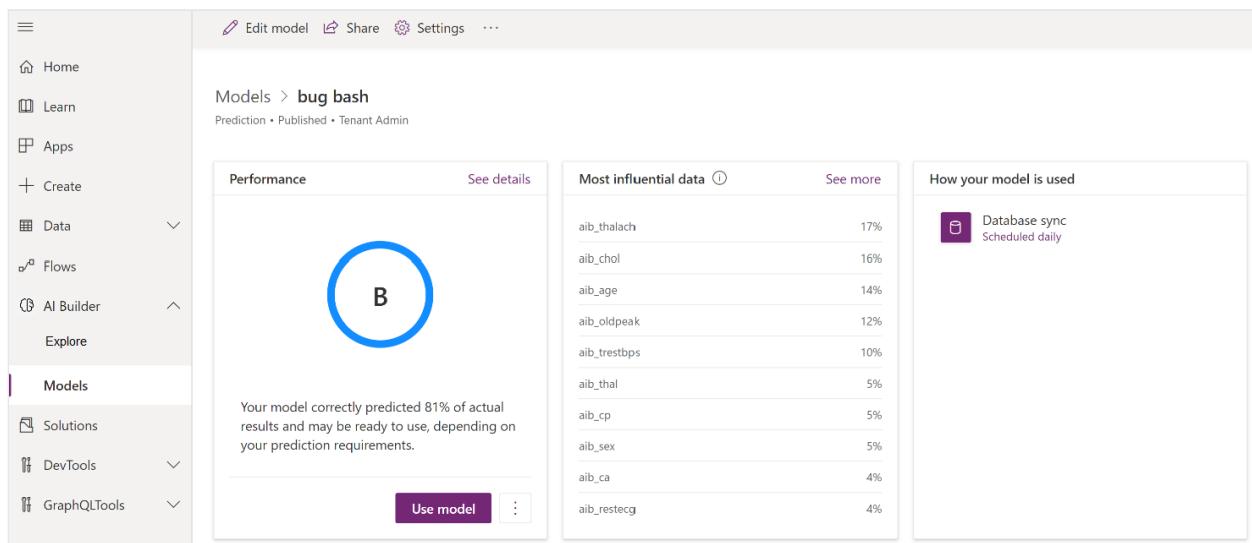
[This topic is pre-release documentation and is subject to change.]

You can use the schedule retrain feature to train prediction models so that their performance doesn't degrade over time. Users can use the schedule run feature to infer rows based on their own conditions and app use. This helps users to use AI Builder units more effectively.

For example, if your data changes weekly, you can schedule the model to train on a weekly basis. It doesn't make sense to keep inferring the same data every day, so you'd probably schedule weekly inference to align with the training schedule.

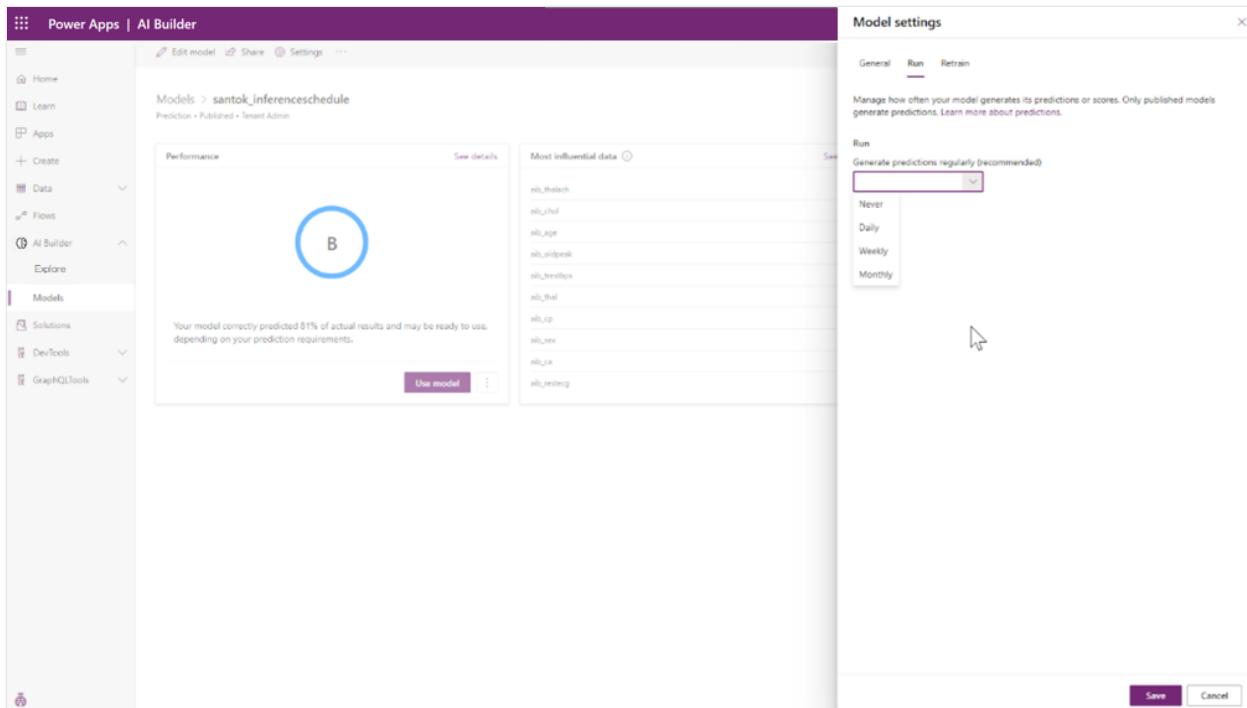
Open the **Schedule** screen from the model detail page under **Database Sync**.

Alternatively, select the **Settings** tab in the menu bar on the model details page.



The screenshot shows the Microsoft Power BI AI Builder Model Details page. On the left, there's a navigation sidebar with options like Home, Learn, Apps, Create, Data, Flows, AI Builder (which is expanded), and Models (which is selected). The main content area shows a model named "bug bash" which is Published and assigned to Tenant Admin. The "Performance" section indicates the model correctly predicted 81% of actual results. Below this, a large blue circle contains the letter "B". A callout text states: "Your model correctly predicted 81% of actual results and may be ready to use, depending on your prediction requirements." There are "Use model" and "..." buttons. To the right, there are three tables: "Most influential data" (listing variables like aib_thalach, aib_chol, aib_age, etc., with percentages) and "How your model is used" (listing "Database sync Scheduled daily").

When you select the link under **Database Sync**, the screen below lists schedule options for **Run**. Switch between the **Run** and **Retrain** tabs to view or modify their respective schedules.



You can set the run or retrain cadence to daily, weekly, or monthly. If you don't want to run prediction or retrain at this time, you can set it to **Never**. Other reasons to set the cadence to **Never** might be that you have data issues upstream and don't want to negatively affect business processes, or you simply don't want to use the model.

! Note

When you schedule your model to retrain, it creates a flow to schedule the training. It's a good idea not to update the flow associated with the AI Builder retrain schedule. AI Builder manages the flow through the schedule retrain experience.

If you make any changes to the AI Builder **Schedule > Retrain** settings, the associated flow is replaced with a new one.

Real-time prediction

In addition to scheduled prediction inferences, AI Builder supports real-time prediction. Although scheduled prediction on a daily cadence might work for some scenarios, there are times when prediction needs to run in real time. Let's say that your retail company wants to know whether an order will be delivered on time. If the order creation and fulfillment processes occur in a short timeframe, you need the delivery prediction during the fulfillment process. Then you can decide whether to use a faster shipment type. Real-time prediction provides you the information at the time you need it.

AI Builder prediction models created after April 2, 2020, are enabled for real-time prediction. A real-time prediction for a model is a synchronous call to AI Builder. AI

Builder supports real-time prediction by value and real-time prediction by reference. The predict operation accepts a single input observation in the request payload and returns the prediction synchronously in the response.

What about existing models?

Prediction models created before April 2, 2020, don't support real-time prediction. Re-create any models created before this date to use real-time prediction capability.

How to use real-time prediction

To use real-time prediction, create a flow in Power Automate. More information: [Use a prediction model in Power Automate](#)

See also

[Train your model in AI Builder](#)

[Publish your model in AI Builder](#)

Overview of the category classification custom model

Article • 01/05/2023

The volume of text data is increasing exponentially for organizations. Channels such as email, documents, and social media contribute increasing amounts of text data. This data carries valuable information that—when extracted and acted on—helps you provide better products and services to your customers. Dealing with this ever-growing volume of data is often time-consuming and error-prone, and can lead to missed business opportunities and increased costs.

Category classification is one of the fundamental natural language processing (NLP) challenges. With category classification, you can identify text entries with tags to be used for things like:

- Sentiment analysis
- Spam detection
- Customer request routing
- Other business needs

Automate and scale your business processes with AI Builder category classification in [Power Automate](#) and [Power Apps](#). AI Builder models help free your employees to act on new insights. Use the results as an input for other AI capabilities, like subscription user churn and predictive analysis. AI Builder learns from your previously labeled text items and enables you to classify unstructured text data stored in Microsoft Dataverse into your own business-specific categories.

Next steps

- [Before you build a category classification model](#)
- [Create a category classification model](#)

See also

- [Use a category classification custom model in Power Automate](#)
- [Use the text recognizer component in Power Apps](#)
- [Feature availability by region](#)
- [Training: Get started with AI Builder category classification \(module\)](#)

Before you build a category classification model

Article • 12/13/2022

Before you build your category classification model, make sure your data is in Microsoft Dataverse and it's structured in the correct format.

Prerequisites

- This model requires the training data to be available within a Dataverse table.
Support for data from external sources is currently unavailable.
- Make sure your administrator has assigned you a security role with Read privilege for the table that has the training data.
- Make sure you have appropriate permissions to create tables in your Power Platform environment. You can use either the System Customizer or System Administrator [built-in security roles](#).

Supported languages

AI Builder category classification supports the following languages. If you try to classify text in other languages, your model might not work properly.

- English
- French
- German
- Italian
- Spanish
- Portuguese

Data preparation

The training data used to train the model from the Dataverse table should conform to the following:

- Store text and tags as two columns in the same table. Each row must have data in the **Text** column.

- You can provide one or more tags to data in the same row in the **Text** column. You can also leave the **Tags** column empty.
- If you've identified multiple tags within the text sample, provide them as delimited text in the **Tags** fields. Currently, commas (,), semicolons (;), and tab characters are supported separators.

Text	Tags
Great clean and quiet room with a free to-go breakfast	Dining, Room
Small but well-orchestrated room that was comfy	Room
I love the view from the 13th floor	(none)

- Make sure to have a minimum of 10 distinct text samples for each tag to be extracted. Tags with fewer than 10 samples won't be trained. In the previous example, there should have been a minimum of 10 rows each that have been tagged with the **Dining** and **Room** tags.
- If **Room** has been tagged in fewer than 10 rows in the data, it will be ignored. The model won't be trained to categorize data for that tag.
- For every tag that is used, provide a minimum of 10 text samples where it *isn't* used.

Text	Tags
Great clean and quiet room with a free to go breakfast	Room
Small but well-orchestrated room that was comfy	Room
(none)	Room

If all rows in the table are tagged to **Room**, and there are no rows—or fewer than 10 rows—that have been tagged to another label, the model will fail the training process.

- A table must have at least two tags, and each one must have 10 text samples.
- You can define up to 200 distinct tags. Each tag is a category that will be identified and extracted from the given text.
- Each sample of text data must have fewer than 5,000 characters.

If you don't have training data and want to try AI Builder category classification, follow these [instructions](#) to use sample data.

Examples of training data format

This section provides examples of the training data format in a Dataverse table.

Columns	Data type	Size
Comments	Text	3,000
Tags	Text	100

Comments	Tags
During my stay, I was completely ignored. The staff failed to pick up on me aspirating and having a UTI. I also had pneumonia.	Care
I was seen very soon after arriving each time and all the staff, nurse, doctor, and anesthetist were very helpful. There seems to be a good sense of teamwork.	Staff, Check-in
The equipment seemed up to date. The nurse/healthcare assistant seemed quite caring.	Facilities, Staff

ⓘ Note

If you don't have your own training data and want to try AI Builder category classification, you can get started by downloading sample data for the category classification model. More information: [Use sample data to do category classification](#)

Import your data into Dataverse

Because training data for a category classification model needs to be available as a Dataverse table, let's begin with preparing data in Dataverse table.

Dataverse includes a powerful set of connectors to help you import data from many sources. More information: [Add data to a table in Microsoft Dataverse by using Power Query](#).

As an example, let's look at how to import training data from an Excel workbook. This example uses a file containing what's shown in the following table.

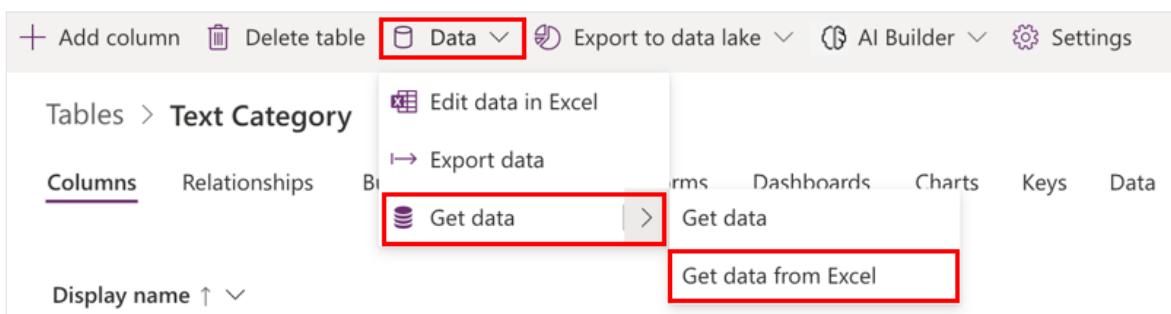
Id	Tags	Text
1	Dining	Breakfast was a bit of a hassle.
2	Dining, Room	Great clean and quiet room with a free to-go breakfast.
3	Room, Dining, Location	The staff we dealt with was very friendly and helpful. The hallways and our room were clean and comfortable. Breakfast (included) was muffins and bagels.
4	Location, Dining	Surrounding area is full of bars and restaurants.
5	Service	Staff was respectful.

In the example, the tags are separated by a comma (,). As an alternative, you can use a semicolon (;) or tab character.

1. Sign in to [Power Apps](#).
2. Select the environment you want to work in.



3. Select Data > Tables.
4. Select your table. If you don't have a table already, follow the steps in [Create a custom table](#).
5. Select Data > Get data > Get data from Excel from the ribbon of the selected table.



6. On the **Import data** screen, select the Excel file that has the data referred to in the [Examples of training data format](#) section earlier in this topic, and then select **Upload**.

Import data

Choose the file you want imported for each table, and we'll map the columns that match. [Learn more](#)

Name	File	Upload
Text Category	aib_categoryclassification.xlsx	<input type="button" value="Upload"/>

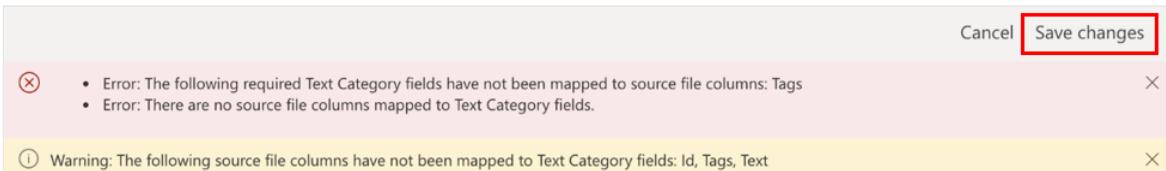
7. To review the field mappings on the **Column mappings for Text Category** screen, select **Map Columns**.

Column mappings for Text Category	
Filter by table column name:	No column name filter applied
Show:	Mappable columns
Text Category columns Source values	
Created By (Azure AD Object ID)	Not set
Created By (Delegate) (Azure AD Object ID)	Not set
Import Sequence Number	Not set
Modified By (Azure AD Object ID)	Not set
Modified By (Delegate) (Azure AD Object ID)	Not set
Owning Business Unit	Not set
Status Reason Value	Not set
Status Value	Not set
Tags *	Tags
Text	Text
Text Category	Id

The left side lists all columns defined in the table. The dropdown list on the right shows the columns available in the Excel file.

Map the **Tags**, **Text**, and **Id** columns from Excel to the respective columns in the table.

8. After you've mapped the columns, go back to the import step by selecting **Save changes** in the upper-right corner.



9. After you see the **Mapping status** as successful, begin the import process by selecting **Import** in the upper-right corner.

The screenshot shows a user interface for importing data. At the top right is a red-bordered 'Import' button. Below it, a message says 'Import data' and 'Choose the file you want imported for each table, and we'll map the columns that match. [Learn more](#)'. A table row is shown with 'Name' (Text Category) and 'File' (aib_categoryclassification.xlsx). To the right of the file name is an 'Upload' button with an upward arrow icon. Next to it is a green circular icon with an 'i' and the text 'Mapping was successful'. On the far right is a blue pencil icon with the text 'Map columns'.

10. The import process might take a few minutes depending on the volume of data being imported. After a few minutes, refresh the **Data** tab of the table to find all the records imported from the Excel file.

You're now ready to go to the next step.

Next step

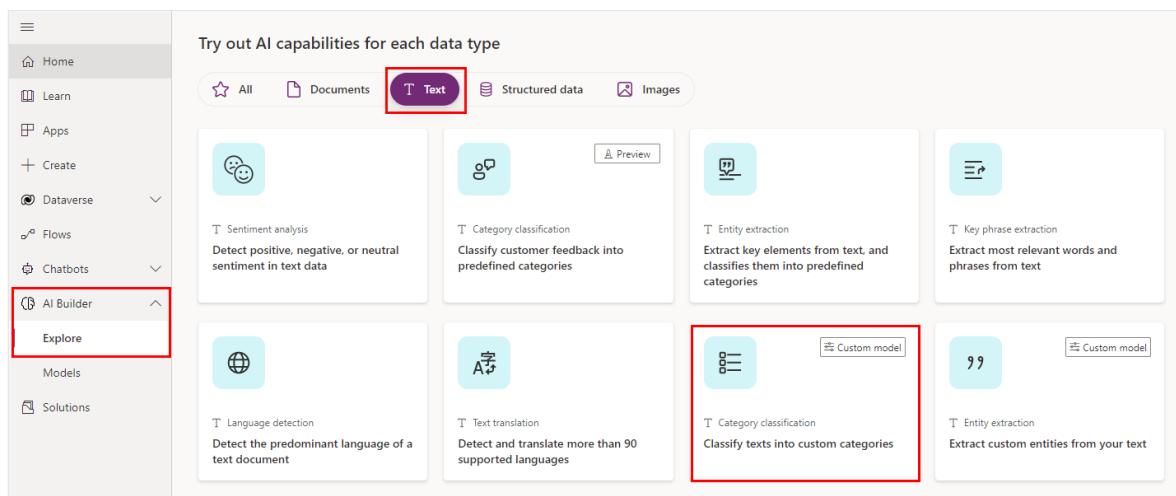
[Create a category classification model](#)

Create a category classification custom model

Article • 12/13/2022

Now that you have your training data in Microsoft Dataverse, you can create a new model and configure it.

1. Sign in to [Power Apps](#), and then select **AI Builder > Explore**.
2. Select **Text**.
3. Select **Category classification - Classify texts into custom categories**.



4. Read the **Classify texts into custom categories** page, and then select **Get started**.
5. Choose **Select text**, select the table, select the column where your training text is stored, and then choose **Select column**.
6. Select **Next**.
7. Choose **Select tags**, select the column where the tags are stored, and then choose **Select column**.
8. (If not pre-selected) Select the separator you used for your tags, and then select **Next**.
9. Review your text and tags to verify the data and the configuration you applied, and then select **Next**.
10. Select the language you want to use for training, and then select **Next**.

That's it! Now you can train your AI model.

Next step

[Train your category classification model](#)

Train your category classification model

Article • 03/05/2022

Before you can use your category classification model, you must train it to perform the way you want. After you train your model, publish it to make it available to other people.

Train

You're now ready to run your training:

- To begin training your category classification model, review your configuration, and then select **Train**. When it's ready, you'll be notified.
- To retrain an existing version of the model, select **Retrain now** from the ellipsis in the model summary performance.

Retraining will train the model based on the same set of configurations you've done when you're creating your category classification model, and on the same dataset. If you want to change the training configurations before retrain, select **Edit model**.

- To train a new version of the model, select **Edit model**. This will create a new draft of your model that you can configure and train.

Troubleshooting tips

If you have difficulty training your model, try these suggestions:

- Make sure your data meets the guidelines listed in [Before you build a category classification model](#).
- Learn how you can [improve category classification model performance](#).
- Download [sample material](#) and use it for testing.

Next step

[Understand category classification model performance](#)

Category classification model training errors and warnings

Article • 12/13/2022

While training the category classification model, you might come across the messages in this article that AI Builder might report. Messages are either *errors* or *warnings*. Each is represented by an icon.

Message	Icon
Error	
Warning	

When an error occurs, you can't continue until you resolve it. If the system is unable to correct a problem, it will show you an error.

Warnings are messages reported as informational. They don't stop you from proceeding. They warn you of possible performance issues when training the model.

Error: InvalidTrainingInput

Errors	See more
Fix these problems and train your model again. Learn more about these errors	
Training report issue: InvalidTrainingInput	

Cause

Following are the possible causes for receiving this error:

- You've supplied fewer than 10 distinct training records per tag in your table.
- You've supplied fewer than two tags, where each has 10 or more distinct training records in your table.
- For each of your tags, you've supplied fewer than 10 distinct training records that don't contain the tag.

Resolution

Add a minimum of 10 distinct training records for each of the tags to be identified. Follow the guidelines in [Before you build a category classification model](#) to do the data preparation.

Error: No tag separator recommended

In the training step, you're prompted to choose **No separator** as the tag separator.

The screenshot shows a user interface for selecting tags. At the top, it says "Select your tags" and "Category Classification Error > Tag". Below that, it says "Tag separator" and notes "We have automatically detected your separator and selected it for you." There are four options shown:

- No separator (one tag per text snippet)**: This option is highlighted with a purple border. It shows a row of five tags: Service, Room, Dining, Location, and Room, Dining.
- Semicolon**: Shows a row of five tags: Service, Room, Dining, Location, Room, and Location, Dining.
- Comma**: Shows a row of ten tags: Service, Room, Dining, Location, Room, Dining, Room, Location, and Dining.
- Tab**: Shows a row of five tags: Service, Room, Dining, Location, Room, and Location, Dining.

Cause

This error will occur if the tag separator used a mix of more than one separator.

Resolution

If you know you have data tagged with multiple tags, recheck the tag separator for each. You must use a single tag separator across all data rows.

Warning: Missing tags for some records

For the new records being created, you find that tags are missing for some records.

Cause

This will happen if you didn't provide a minimum of 10 sample text records for the tag while training the model.

Resolution

Add more sample text for the tag with missing data and retrain the model.

See also

[Train your category classification model](#)

Understand category classification model performance

Article • 04/07/2022

After each training, AI Builder uses the test dataset to evaluate the quality and accuracy of your AI model. A summary page for your model shows your model training results, including a **Performance** score.

AI Builder calculates your model's performance score based on the precision and recall of the prediction results, as well as the F1 score:

- **Performance score:** This score is calculated using precision, recall, and F1 scores. Performance score values are from 0 and 100. Generally, the higher the performance score, the better your model is.
- **Precision:** The fraction of correct predictions among all the positive predictions.
- **Recall:** The fraction of correct predictions among all true positive cases.

Quick test

You can also select **Quick Test** to assess the quality of the model. Just enter text that you want to tag. More information: [Evaluate your model](#)

Next step

[Improve the performance of your category classification model](#)

Improve the performance of your category classification model

Article • 03/05/2022

If your model performance isn't where you want it to be, there are a few things you can try. These tips can help you tweak your model to improve its predictive power.

Add more correctly labeled training data

The more correctly labeled training data you have, the better your model will perform. For example, let's say you have a Yes/No label. If most of your data only has a Yes in this column, your AI model probably won't learn much from this data. If your data isn't correctly labeled, the model is probably not going to learn very well. It's ideal to begin with a small set of correctly labeled examples - perhaps 100 or less. From there, you can continue to double the number of examples iteratively and retrain each time, noting the performance change. Generally speaking, more data is better, but there are diminishing returns for adding data the larger your dataset gets.

More tips

- Make sure your use of tags is balanced in your training data. For example: You have four tags for 100 text items. The two first tags (*tag1* and *tag2*) are used for 90 text items, but the other two (*tag3* and *tag4*) are only used on the remaining 10 text items. The lack of balance might cause your model to struggle to correctly predict *tag3* or *tag4*.
- Make sure you train your model using data that's similar to the what you expect to use the model for.

Next step

[Publish your category classification model](#)

See also

[Category classification prebuilt model](#)

Publish your category classification model

Article • 02/14/2022

When you're satisfied with your model, you can publish it. After it's published, your model is ready to run on your Microsoft Dataverse data or be used through Power Automate. It can take up to two hours for the results to be available.

In the **Model settings** pane on the right side of the screen, you can turn on Dataverse and get important information about where the suggested tags will be stored.

ⓘ Important

- Currently, you can only schedule Dataverse runs by using the input table and column that you used for training. AI Builder creates a destination table where the model will store the suggested tags and their confidence scores. This destination table will have an N:1 (many-to-one) relationship with your source/input table.
- AI Builder uses Power Automate to schedule and run tagging on your data. When you publish your model to run on your Dataverse data, this will use Power Automate runs from your subscription.

More information: [Publish your model in AI Builder](#)

Where will suggested tags be stored?

The following table will be created when you publish a model: **TC_{Model_Name}**

Tags will be stored in a column called **new_Tags** under this table.

ⓘ Note

Due to the character limitation of the column name, if the provided syntax creates text that is over the character limit, AI Builder cuts off the end of the text. This means that the model name might be incomplete or missing. You should choose shorter names so everything is visible.

Next step

[View predictions](#)

Use sample data to do category classification

Article • 06/23/2022

Get started exploring AI Builder category classification by using sample data to build and train a category classification model. The sample data uses customer feedback for a hospital. The goal is to train a model that can predict the category of newly received feedback. This model can help the hospital administrator free up time from categorizing patient feedback, leaving more time to act on it and provide a better experience to patients.

ⓘ Note

This sample data is added to your environment automatically if you enable the **Deploy sample apps and data** setting when creating your database.

Set up an environment with data

1. Download [AIBuilder_Lab.zip](#), which contains category classification sample data.

ⓘ Note

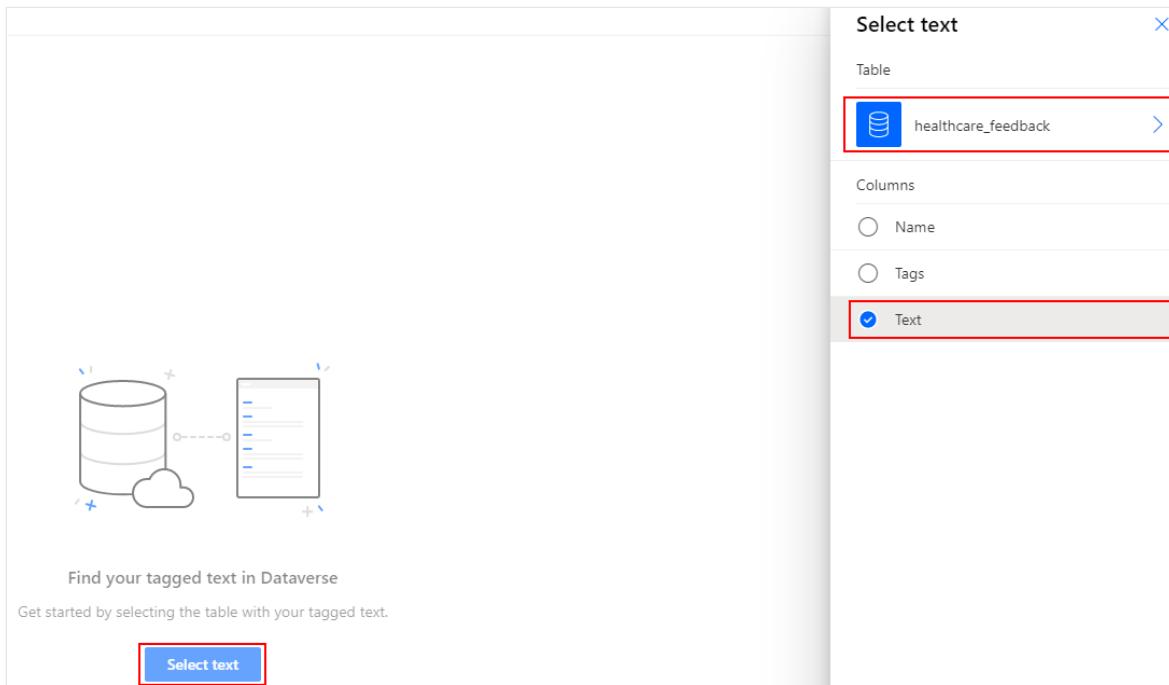
The AIBuilder_Lab.zip file also contains sample files for working with other AI Builder model types, in addition to some hands-on labs that you can use to learn more about AI Builder. For more information about the contents of the zip file, go to the [readme.txt](#) file that's included in the zip file.

2. Import the **AIBuildetTextSample_1_0_0** solution to your Microsoft Power Platform environment. To learn more, go to [Import, update, and export solutions](#).
3. Go to the Lab Data/Text Classification folder within the lab files and then upload data from `pai_healthcare_feedbacks`.

Create your model

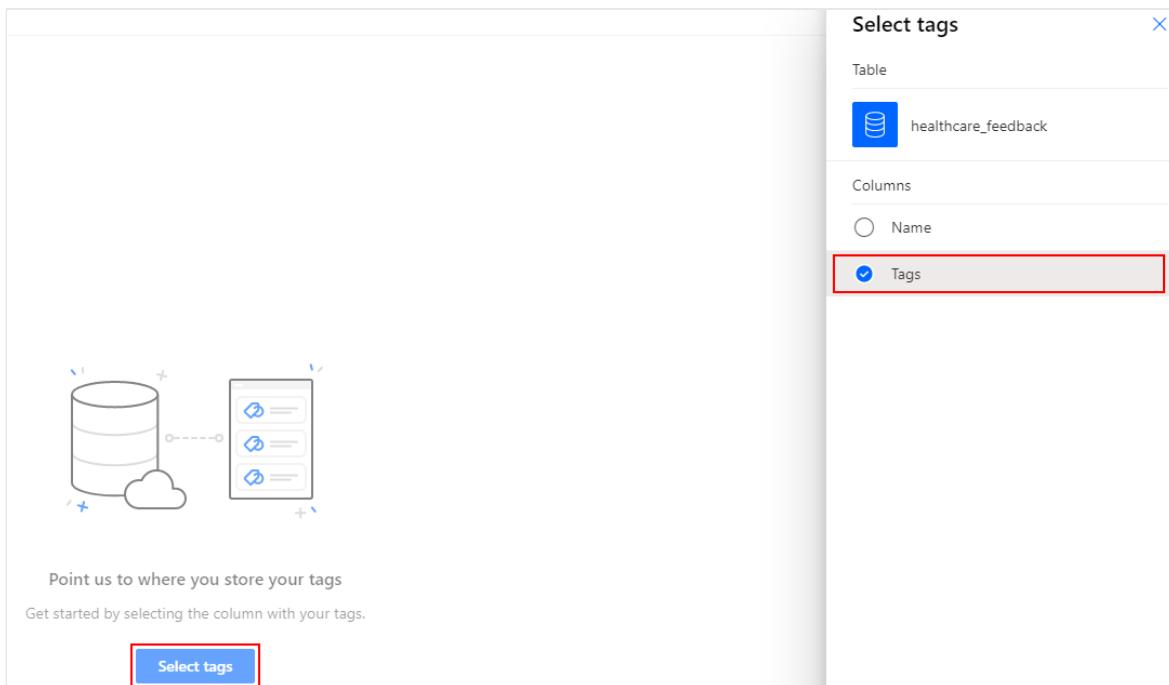
1. Go to the AI Builder build screen, and select **Category classification**.
2. Enter a name, and then create your model.

3. Choose **Select text**, select the **healthcare_feedback** table, and then select the **text** column.



4. Choose **Select column**, preview the tagged text, and then select **Next**.

5. Choose **Select tags** and then select the **tags** column.



6. Choose **Select column**, verify that the correct separator (comma) is chosen, and then select **Next**.

7. Review your text and tags, and select **Next**.

8. Select **English** as the text language, and then select **Next**.

9. Review the model summary, and then select **Train** to train your model.

View results

Article • 04/07/2022

This topic shows you how to see the output of your prediction model.

1. After you select **Use model** and configure it to run on Microsoft Dataverse, the output location appears in the **Settings** pane under the **Run** pivot.

The name shown in **Tags output** is the name of the entity and attribute that's created after publishing. It's a link that takes you to the entity viewer section where the new fields added by AI Builder appear.

2. In the **Views** section, see values of the output fields for the different records. Use the **Filter by** function in the lower-right side pane to filter for only the records that don't have a label.
3. Next, you can build a basic model-driven app to use the output. For information about how to build a model-driven app in Power Apps, see [model-driven app overview](#).

Next step

[Use a category classification model to generate tags](#)

Use a category classification model to generate tags

Article • 01/04/2023

Use in Power Automate

If you want to use your trained model in Power Automate, see [Use a category classification custom model in Power Automate](#).

Use in Power Apps

You can integrate your AI Builder category classification models in Power Apps Studio by using the formula bar. For more information, see [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Set a run schedule on Microsoft Dataverse (preview)

[This topic is pre-release documentation and is subject to change.]

Create a schedule to generate predictions regularly in Microsoft Dataverse.

1. To set the run schedule, go to the **Run** view in the **Model settings** panel.
2. To configure your model to run on your database and generate predictions, select **Generate predictions when new data is added**.

Your model runs whenever a new row is added to its table.

For more information, see [Run your prediction model](#).

Note

You can't set run schedule for imported category classification models.

What if the model isn't writing new tag suggestions?

- Check that you didn't exceed the number of runs for your Power Automate subscription.
- Turn off the Dataverse run setting, and then turn it back on.

Overview of the entity extraction custom model (preview)

Article • 12/13/2022

AI Builder entity extraction models recognize specific data in text that you target based on your business needs. The model identifies key elements in the text and then classifies them into predefined categories. This can help you transform unstructured data into structured data that's machine-readable. You can then apply processing to retrieve information, extract facts, and answer questions.

AI Builder features two types of entity extraction models: prebuilt and custom. [Prebuilt models](#) are ready to use, don't require training or publishing, and are appropriate for many uses where customization isn't needed. Custom entity extraction models must be built, trained, and published before you can use them. By using your own training data and design parameters, you can create an entity extraction model that's purpose-built for your unique requirements.

See also

[Entity extraction prebuilt model](#)

Entity extraction custom model requirements and limitations

Article • 12/13/2022

Data format

- Documents can't exceed 5,000 characters.

Supported languages

- English
- Chinese-Simplified
- French
- German
- Portuguese
- Italian
- Spanish

Next step

[Create an entity extraction custom model](#)

Create an entity extraction custom model

Article • 12/13/2022

Create your model

To create your custom entity extraction model:

1. Provide at least 10 examples of your text data.
2. Review the results from existing, prebuilt entities.
3. Refine your results by creating your own custom tables or modifying existing, prebuilt tables.
4. Review your model, and train it.
5. Evaluate your model (optional).

Upload examples of your text data

You need at least 10 data examples to start customizing the model. AI Builder will identify some tables out of the box. You can customize your model by creating new entity types with a small set of examples or by modifying the existing entity types.

To upload your data:

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. In the left pane, select **AI Builder** > **Explore**, and then select **Entity Extraction**.
3. Enter a name for your model, and then select **Create**.
4. Select the Microsoft Dataverse **Table** and the **Column** that contains your data.

Here's an example of a selected table and column:

Add data

Find the sentences you want to analyze by selecting a table and column. The model will start by identifying basic information in those examples. [Learn more](#)

Table	Column
Account	Account Name

Review and refine entities

AI Builder will automatically extract the prebuilt entity types from your uploaded examples. If the results are satisfactory, you don't need to customize a model and can use the [prebuilt entity extraction](#) model out of the box.

You can customize your entity extraction model in these ways:

- Create a new entity type.

You need to provide at least five examples to create a new entity type. For example, to create a new entity type named **size**, you can add an example like "The suitcase was {large}." The braces designate that "large" is of entity type **size**.

- Modify an existing entity type.

You need to add least five examples to an existing entity type to modify it.

- Deselect any prebuilt entities that you don't want to include in the model.

You can select/deselect prebuilt entity types from the settings.

Next step

[Train and publish your entity extraction custom model.](#)

Train and publish your entity extraction custom model

Article • 12/13/2022

After you create your entity extraction model, you can train and publish it to make it available.

Train and validate your model

1. After you create your model, select **Next** to check your selected entities. If everything looks good, select **Train** to train your model.
2. On the **Training complete** screen, select **Go to Details page**.

Quick-test your model

You can use the **Details** page to test your model before you publish or use it:

1. On the **Details** page, enter the text you want to test.
The quick test should only take a few seconds before displaying the results.

Publish your model

If you're happy with your model, you can select **Publish** to publish it. When publishing is completed, your model is promoted as published and is ready to be used. More information: [Publish your model in AI Builder](#)

After you've published your model, you can use it in a Power Apps canvas app or in Power Automate.

Edit your model

You can create a new version of your published model by selecting **Edit model** on the model details page. You can create new entity types or modify existing entity types.

See also

[Feature availability by region](#)

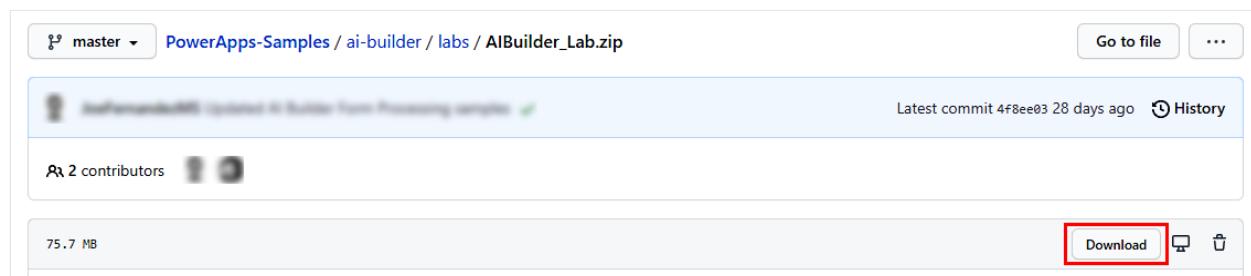
Use sample data to do entity extraction

Article • 12/13/2022

You can get started with the entity extraction model using sample data. The sample data provided refers to feedback or conversations related to travel. The intent is to use the entity extraction AI model to extract and identify entities related to places, persons, organizations, date, time, and a custom entity type: **Location Type**.

Get the sample data

1. Download **AIBuilderLab.zip**.



1. Look for **AIBuilderLabSolution_1_0_0_1.zip** that contains the sample table, *Travel Feedback*.

Sample data for this table can be found in the Lab Data\EntityClassification\ai_b_travelfeedback.csv.

2. Sign in to [Power Apps](#) and choose the environment where you've imported the solution from step 1.
3. Select Solutions > **AIBuilderLabs** > Tables > **Travel Feedback**.

Solutions > AI Builder Lab Solution		
Display name	Name	Type
Object Detection Product	... aib_objectdetectionproduct	Table
Online Shopper Intention	... aib_onlineshopperintention	Table
Travel Feedback	... aib_travelfeedback	Table
healthcare_feedback	... pai_healthcare_feedback	Table

4. Select Data > Get data > Get data from Excel from the ribbon.

Solutions > AI Builder Lab Solution > Travel Feedback

Columns Relationships Business rules Views Forms Dashboards Charts

Display name ↑ ↓	Name ↓	Data type ↓	
Comments	aib_comments	Text	Custom ✓
Created By	createdby	Lookup	Standard ✓
Created By (Delegate)	createdonbehalfby	Lookup	Standard ✓
Created On	createdon	Date an...	Standard ✓

You'll see the Import data screen:

Name	File	Mapping status
Travel Feedback	File not uploaded	Not mapped

5. Select **aib_travelfeedback.csv**, which has the data to be imported in this table.

File	Mapping status
aib_travelfeedback.csv	Mapping errors exist

Map the data

1. Review the column mappings by selecting **Map columns**.
2. Map the **Name** and **Comments** columns in the **Source values** column, and select **Save changes**.

Column mappings for Travel Feedback

Filter by table column name: No column name filter specified Show: Mappable columns Table column filter has been applied

Travel Feedback columns	Source values
Comments	Comments
Created By (Azure AD Object ID)	Not set
Created By (Delegate) (Azure AD Object ID)	Not set
Import Sequence Number	Not set
Modified By (Azure AD Object ID)	Not set
Modified By (Delegate) (Azure AD Object ID)	Not set
Name *	Name
Owning Business Unit	Not set

You'll return to the Import data screen with status updated to reflect that the mappings are complete.

Import

Import data

Choose the file you want imported for each table, and we'll map the columns that match. [Learn more](#)

Name	File	Mapping status
Travel Feedback	aib_travelfeedback.csv	Upload (i) Mapping was successful

Import the data

1. Select **Import** on the top right for the import process to begin.

It might take a few minutes for the import process to complete, depending on the rows included in the file provided.

2. Select the **Data** tab for the Travel Feedback table in the solution to view all the records imported.

Add record Refresh data Add subcomponents Delete table Data Settings Switch to classic

Solutions > AI Builder Lab Solution > **Travel Feedback**

Columns Relationships Business rules Views Forms Dashboards Charts Keys **Data**

	Name	Comments
	1	We are looking for a beach holiday this summer with our two kids
	10	Could you suggest a hotel in San Francisco close to a park? I will be traveli...
	11	I got the tickets to Walt Disney theme park for this weekend
	12	For booking enquiries dial (555)555-5555 or email at queries@query.com
	2	I spent the evening at Central Park last Saturday
	3	The Grand Canyon is a steep-sided canyon carved by the Colorado River i...

See also

[Use Power Fx in AI Builder models in Power Apps \(preview\)](#)

Overview of the object detection model

Article • 01/05/2023

Object detection can help expedite or automate business processes. In retail, it can help streamline inventory management, allowing retail leaders to focus on onsite customer relationships. In manufacturing, technicians can use it to speed the repair process by quickly accessing the manual for a piece of machinery for which the UPC/serial number isn't obvious.

Organizations of any size can use AI Builder object detection to add these capabilities for their own custom objects to their apps.

Next step

[Collect images](#)

See also

- [Feature availability by region](#)
- [Use an object detection model in Power Automate](#)
- [Training: Detect objects with AI Builder \(module\)](#)

Collect images

Article • 12/13/2022

To train an object detection model to recognize your objects, you have to gather images that contain those objects. Adhere to guidelines for image quantity and quality for better results.

Format and size

The images you'll feed your object detection model need these characteristics:

- Format:
 - JPG
 - PNG
 - BMP
- Size:
 - 6 MB maximum for training
 - minimum width / height of 256 pixels x 256 pixels

Data quantity and data balance

It's important to upload enough images to train your AI model. A good starting point is to have at least 15 images per object for the training set. With fewer images, there's a strong risk that your model will learn concepts that are just noise, or not relevant.

Training your model with more images should increase the accuracy.

Another consideration is to make sure your data is balanced. If you have 500 images for one object and only 50 images for another, your training dataset isn't balanced. This can cause the model to be better at recognizing one of the objects. For more consistent results, maintain at least a 1:2 ratio between the object with the fewest images versus the one with the most. For example, if the object with the greatest number of images has 500 images, the object with the fewest images should have at least 250 images for training.

Use more diverse images

Provide images that are representative of what will be submitted to the model during normal use. For example, let's say you're training a model to recognize apples. If you only train images of apples on plates, it might not consistently recognize apples in trees.

Including different kinds of images will make sure that your model isn't biased and can generalize well. The following are some ways you can make your training set more diverse.

Background

Use images of your objects in front of different backgrounds—for example, fruit on plates, in hands, and on trees. Photos in context are better than photos in front of neutral backgrounds because they provide more information for the classifier.



Lighting

Use training images that have different lighting, especially if the images used for detection might have different lighting. For example, include images taken with flash, high exposure, and so on. It's also helpful to include images with varied saturation, hue, and brightness. Your device camera probably lets you control these settings.



Object size

Provide images in which the objects are of varied sizes, capturing different parts of the object—for example, a photo of bunches of bananas and a closeup of a single banana. Different sizing helps the model generalize better.



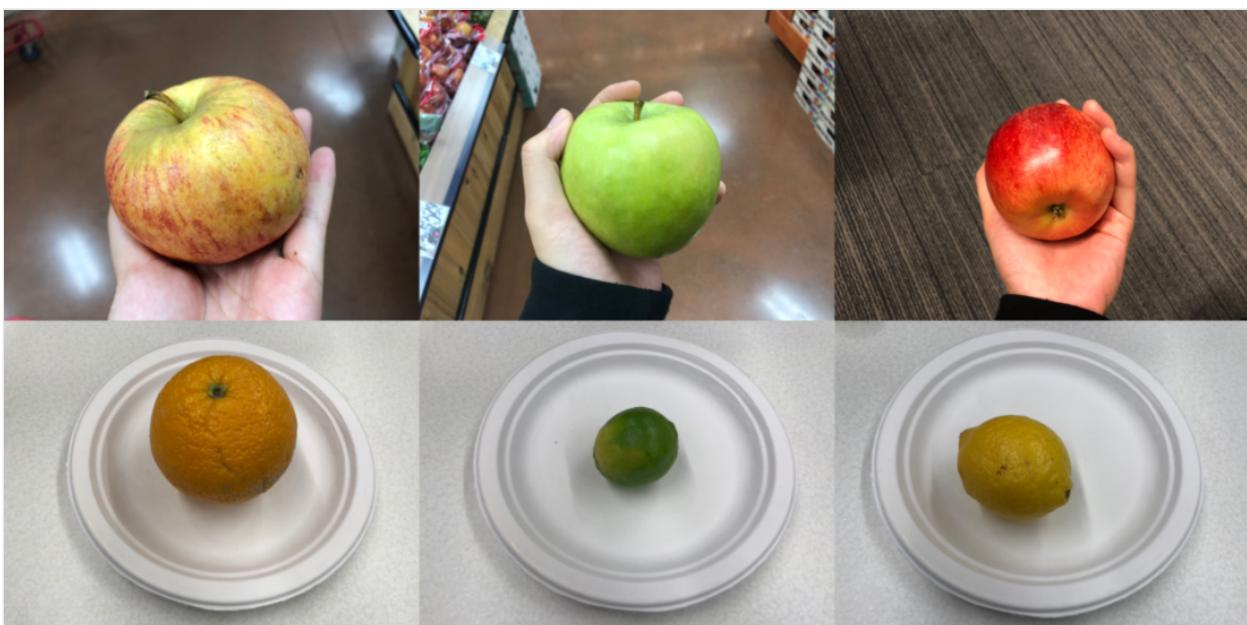
Camera angle

Try to provide images taken from different angles. If all your photos are from a set of fixed cameras such as surveillance cameras, assign a different label to each camera. This can help avoid modeling unrelated objects such as lampposts as the key feature. Assign camera labels even if the cameras capture the same objects.



Unexpected results

AI models can incorrectly learn characteristics that your images have in common. Let's say you want to create a model to distinguish apples from citrus. If you use images of apples in hands and of citrus on white plates, the model might train for hands versus white plates instead of apples versus citrus.



To correct this, use the above guidance on training with more varied images: provide images with different angles, backgrounds, object size, groups, and other variants.

Need help collecting images?

You can use [Trove](#) to gather images for your projects. Trove is an app that connects you directly with photo takers, allowing you to collect more relevant and accurate photos. Using Trove, you can post your project descriptions, outline the types of photos you are looking for, and only approve the photos that you want. Trove provides licensing and privacy frameworks, so you can collect high quality data responsibly and safely.

To use Trove, [sign up and add your AI project](#).

Next step

[Get started with object detection](#)

Build your object detection custom model

Article • 07/08/2022

1. Sign in to [Power Apps](#), and then select **AI Builder > Explore**.
2. Select **Images > Object detection** AI Builder custom model type.
3. Select **Get started**.

Select the model domain

The first thing you'll do when you create an AI Builder object detection model is to define its domain. The domain optimizes the model for specific use cases. There are three domains:

- **Common objects:** The default value. Use this if your use case doesn't fit the specific applications below.
- **Objects on retail shelves:** Detects products densely packed on shelves.
- **Brand logos:** Optimized for logo detection.

Select a model for your domain and click **Next**.

ⓘ Note

Domain specific object detection uses more AI Builder credits than common object detection, and therefore costs more to use. To estimate the impact of using domain specific versus common object on your organization, use the [AI Builder calculator](#).

Provide object names

Next, provide the names of the items you want to detect. You can provide up to 500 object names per model.

There are two ways to provide object names:

- Enter object names directly in AI Builder.
- Select names from your Microsoft Dataverse table.

Important

If you change input mode, you lose any existing object names. In other words, if you type object names in AI Builder and then change to selecting from a database, all object names typed and their associated bounding boxes are deleted from your model. This is not recoverable. The same applies if you switch from selecting object names from a database to typing object names.

To choose objects from a Dataverse table, choose **Select from database** above **Choose objects for your model to detect**, and then choose **Select object names**. If you change your mind before you select your table, you can select **Add objects manually** to switch back.

Enter names in AI Builder

To provide object names directly in AI Builder, just enter the name in the space where the object is detected in the image. Then press **Enter** or select **Add new object** to continue.

- To edit an object name, select the object name and then make your change.
- To remove an object name, select the trashcan icon.

Select names from a database

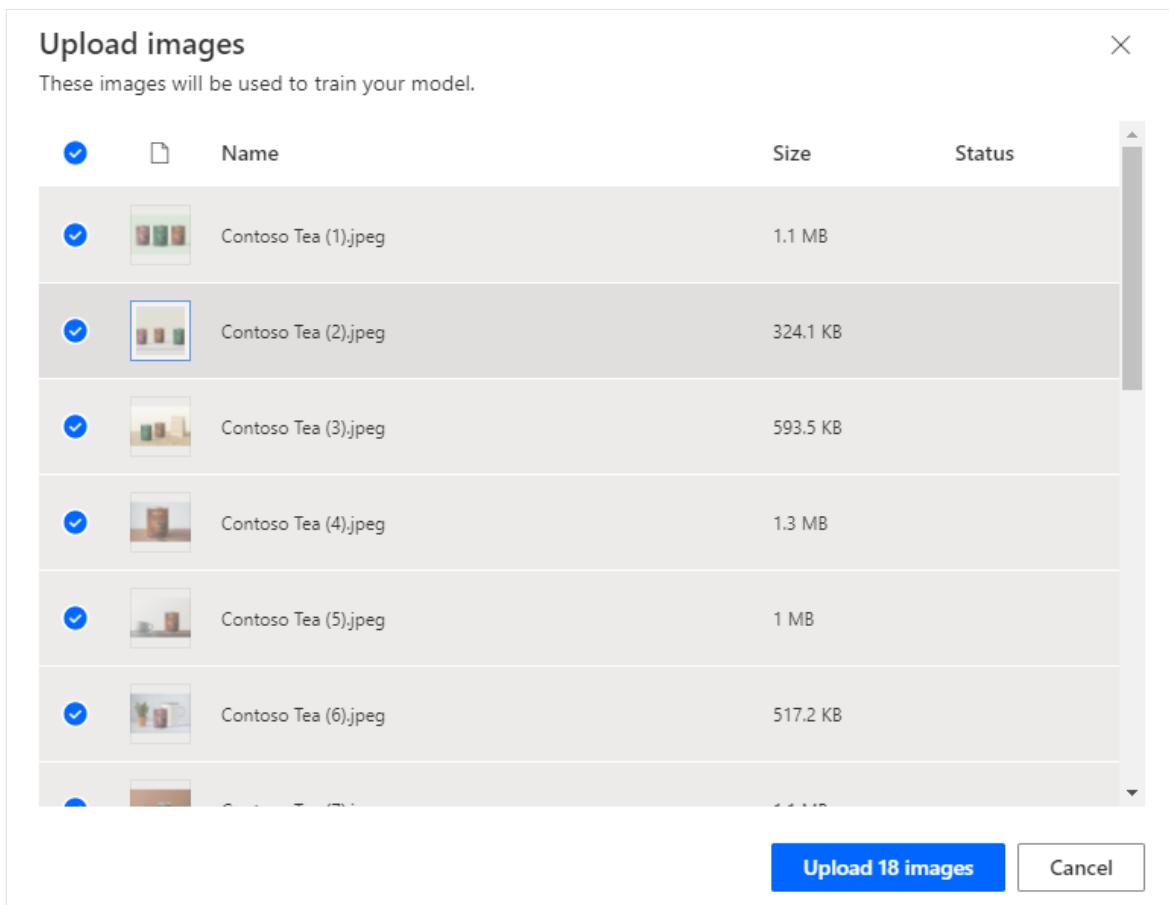
If your data isn't in Dataverse, go to [Prerequisites](#) for information about how to import data into Dataverse.

1. Choose **Select from database** to see tables in your environment.
2. On the panel to the right, find and select the table that contains your object names.
3. Select the column that contains the names of your objects, and then choose **Select column**.
4. From the list of objects in your table, select the ones that represent the objects you want to detect.
5. At the bottom of the screen, select **Next**.

Upload images

Now let's move on to the image upload step. The pictures you collected ahead of time will now come in handy because you need to upload them to AI Builder.

1. Prepare your images in the storage location where you want to add them from.
Currently you can add images from local storage, SharePoint, or Azure Blob Storage.
2. Make sure your images follow the qualitative and quantitative [guidance](#).
3. In AI Builder, select **Add images**.
4. Select the data source where your images are stored, and then select the images that contain your objects.
5. Confirm the images that appear in AI Builder before you upload them. Deselect any image you want to exclude.

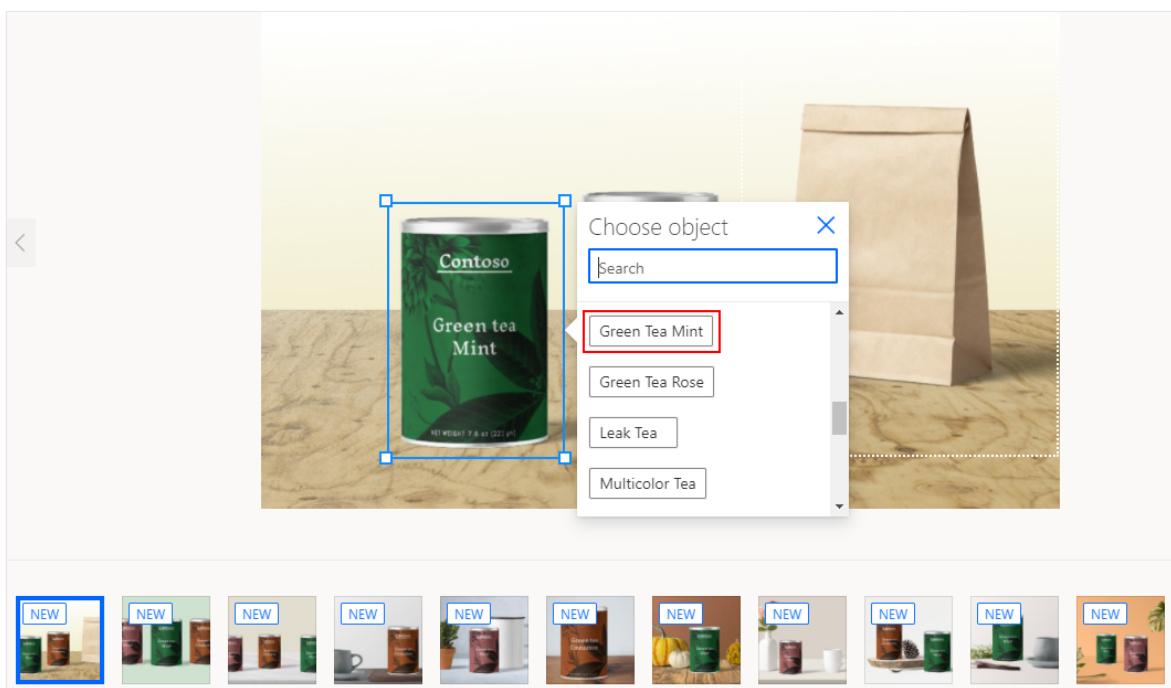


6. Select **Upload <number> images**.
7. When the upload is complete, select **Close**, and then select **Next**.

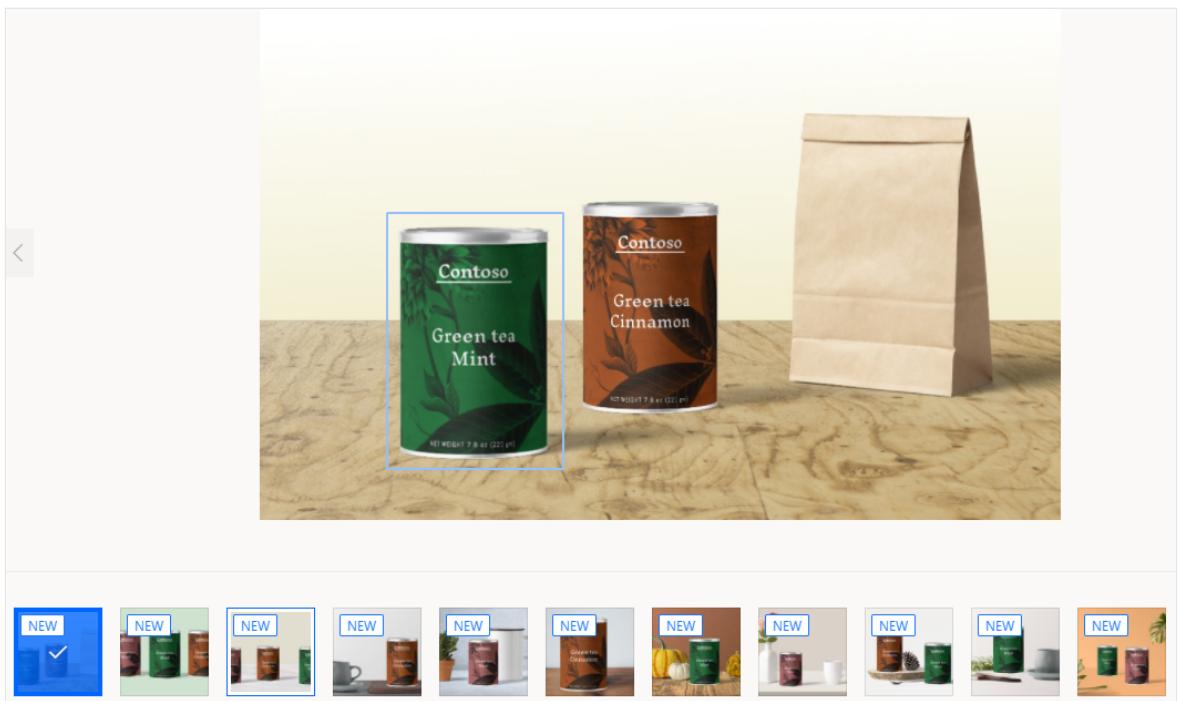
Tag images

This section explains the tagging process that's a key part of object detection. You draw rectangles around the objects of interest, and then assign a name to the rectangle that you want the model to associate with this object.

1. On the **Tag the objects in your images** screen, select the first image in your gallery.
2. Draw a rectangle around the object. To do this, press and hold your mouse at the upper-left corner of the object, and then drag down to the lower-right corner of the object. The rectangle should fully encompass the object you want your model to recognize.
3. After you draw a rectangle, you can associate a name to the object from the list of names you already selected.



4. Your tag is created when you see it surrounding an object.



5. Navigate from image to image, and tag at least 15 images per object name to build a model.
6. After you're done tagging your images, select **Done tagging**. Your data is saved as you create rectangles.
7. In the grid view, you can view a summary of all the tags you created and which images you created. This lets you know how much more work is needed to move forward.
8. Until you reach the minimum for content quantity, you can't move forward. After you have at least 15 images per object name, you'll be able to select **Next** at the bottom of the screen.

That's it! Congratulations, you've created a training set for object detection.

Next step

[Train and publish your object detection model](#)

See also

[Use an object detection model in Power Automate](#)

Train and publish your object detection model

Article • 04/07/2022

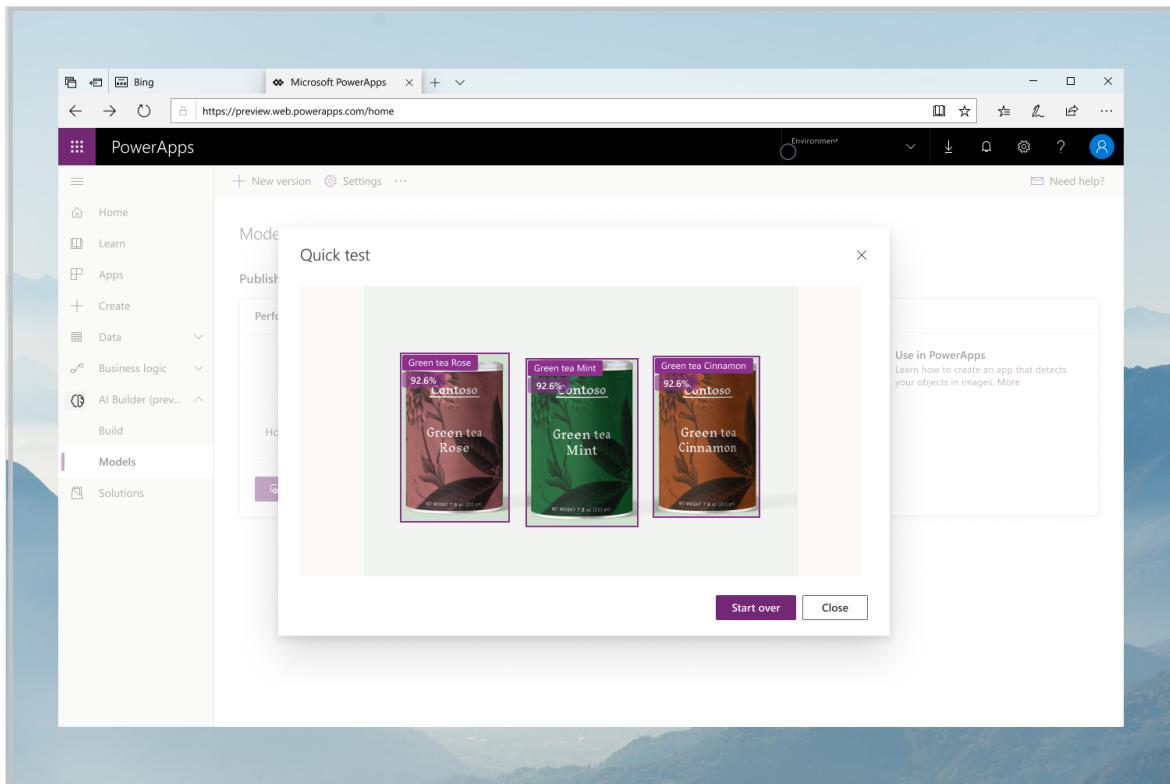
In this section, you'll learn how to verify your data and then how to train, test, and publish your model.

1. On the **Train your model** screen, verify that you have the correct names and the correct number of images.
2. If the data is all good, select **Train** to kick off the training of your object detection model.

Quick-test your model

After your model is trained, you can see it in action from its details page. More information: [Manage your model in AI Builder](#)

1. From your model details page, select **Quick Test** in the **Last trained version** section.
2. Upload an image that contains your objects to test your model.
3. Your model will be applied to the image you uploaded. This step might take a while.
4. After your model has finished running, the rectangles found will be drawn directly on the picture.



How to interpret your model performance score

If you quick-test your model after it's trained, a performance score appears on its details page. This performance score indicates how well the model did on the images you uploaded. This score isn't an indication of how well it will perform on your future images because it hasn't seen them yet.

If you upload fewer than 50 images for a label, you're more likely get a high score—as high as 100 percent. This doesn't mean your model is bulletproof. It means your model has made no mistake on a subset of the images you provided (called the *test set*). The smaller the training set, the smaller the test set, and the more likely it is that your model will be right when the performance score is calculated.

Model performance scores are more reliable when you have more than 50 images per label, and when these scores remain stable even when you change the training set.

Publish your object detection model

From here, you can run more tests with other pictures. If you're happy with the results, you can [publish](#) your model to use it in Power Apps or Power Automate.

Limits

Action	Limit	Renewal period
Object detection calls (per environment)	4800	60 seconds

See also

[Manage your model in AI Builder](#)

Transition to the newest object detection model version

Article • 06/23/2022

AI Builder is including improvements to object detection in a newest model version. Your object detection preview models created before March 5, 2020 are deprecated. More information: [Document processing and object detection preview models in AI Builder are deprecated](#)

Starting March 5, 2020, you have to re-create your existing object detection models if you want to continue to use them. Models created after this date will automatically use the new model version. It's a good idea to transition to the new model version as soon as possible.

What do I need to do?

You must re-create object detection models created before March 5, 2020

1. Identify any object detection models created before March 5, 2020 that you want to continue using.
2. For each model, [create a new object detection model](#) with the same training images you used in the original model.
3. After the new model is trained and published, make sure you update any apps or flows to use the new model.
4. After you're done with your deprecated object detection model, you can delete it.

We understand you might have spent significant time building your object detection models. [Contact us](#) if you need help with migrating larger complex models.

Why this change?

Object detection models are being upgraded for general availability. Your existing preview models won't be compatible with the new model version.

Until March 5, 2020, object detection models stored images as attachments in the Note entity. With this update, these models now use the new File and Image data types in Microsoft Dataverse, which enables a better and more optimized usage of capacity.

See also

Manage your model in AI Builder

Use sample data to do object detection

Article • 03/05/2022

To explore the possibilities of object detection in AI Builder, you can get started by building and training an object detection model using sample pictures and labels.

ⓘ Note

This sample data is added to your environment automatically if you enable the **Deploy sample apps and data** setting when you create your database.

Get the sample data

Download [AIBuilder_Lab.zip](#) file, which contains object detection sample images and labels.

ⓘ Note

The AIBuilder_Lab.zip file also contains sample files for working with other AI Builder model types, in addition to some hands-on labs that you can use to learn more about AI Builder. For more information about the contents of the zip file, see the [readme.txt](#) file that's included in the zip file.

Add labels in Microsoft Dataverse

1. Go to [Power Apps](#), and then select **Data > Tables** in the left pane.
2. Prepare a table with one column in a text format, either by using an existing table or creating a new one. If you need to create a new table, see [Create a table](#).
3. On the top menu, select **Get data**.
4. In the list of data sources, select **Excel**.
5. Select **Browse** to upload your Excel workbook, and then select the sheet or sheets that your data is in. You might have to allow third-party cookies for your browser to complete this step.
6. On the **Edit Queries** screen, select **Transform table > Use first row as headers**.
7. Select **Next > Load to new entity**.
8. Use the drop-down menu to select your target table, and then map your columns to the destination fields.
9. Select **Next** to finish.

See also

[Use the object detection component in Power Apps](#)

Overview of the document processing model

Article • 01/05/2023

Document processing lets you read and save information from standard documents such as invoices or tax documents. When you automate this process, you can save valuable time by reviewing, extracting, organizing, and saving the data automatically by using Power Automate and Power Apps.

[Train](#) your model and define the information to be extracted from your forms. You only need five form documents to get started. Quickly get accurate results tailored to your specific content. With AI Builder, you don't need a lot of manual intervention or data science expertise.

After you train and publish your model, you can use it in a flow in [Power Automate](#) or in a canvas app in [Power Apps](#).

ⓘ Note

Document processing name used to be form processing until June 2022.

Next step

[Create a form-processing model](#)

See also

- [Feature availability by region](#)
- [AI model types](#)
- [Training: Process custom documents with AI Builder \(module\)](#)

Requirements and limitations for a document processing model

Article • 05/23/2023

Languages supported

The following languages are supported when training a document processing model and selecting **Structured and semi-structured documents** as document type: Afrikaans, Albanian, Angika (Devanagari), Arabic, Asturian, Awadhi-Hindi (Devanagari), Azerbaijani (Latin), Bagheli, Basque, Belarusian (Cyrillic), Belarusian (Latin), Bhojpuri-Hindi (Devanagari), Bislama, Bodo (Devanagari), Bosnian (Latin), Brajbha, Breton, Bulgarian, Bundeli, Buryat (Cyrillic), Catalan, Cebuano, Chamling, Chamorro, Chhattisgarhi (Devanagari), Chinese Simplified, Chinese Traditional, Cornish, Corsican, Crimean Tatar (Latin), Croatian, Czech, Danish, Dari, Dhimal (Devanagari), Dogri (Devanagari), Dutch, English, Erzya (Cyrillic), Estonian, Faroese, Fijian, Filipino, Finnish, French, Friulian, Gagauz (Latin), Galician, German, Gilbertese, Gondi (Devanagari), Greenlandic, Gurung (Devanagari), Haitian Creole, Halbi (Devanagari), Hani, Haryanvi, Hawaiian, Hindi, Hmong Daw (Latin), Ho(Devanagiri), Hungarian, Icelandic, Inari Sami, Indonesian, Interlingua, Inuktitut (Latin), Irish, Italian, Japanese, Jaunsari (Devanagari), Javanese, Kabuverdianu, Kachin (Latin), Kangri (Devanagari), Karachay-Balkar, Kara-Kalpak (Cyrillic), Kara-Kalpak (Latin), Kashubian, Kazakh (Cyrillic), Kazakh (Latin), Khaling, Khasi, K'iche', Korean, Korku, Koryak, Kosraean, Kumyk (Cyrillic), Kurdish (Arabic), Kurdish (Latin), Kurukh (Devanagari), Kyrgyz (Cyrillic), Lakota, Latin, Lithuanian, Lower Sorbian, Lule Sami, Luxembourgish, Mahasu Pahari (Devanagari), Malay (Latin), Maltese, Malto (Devanagari), Manx, Maori, Marathi, Mongolian (Cyrillic), Montenegrin (Cyrillic), Montenegrin (Latin), Neapolitan, Nepali, Niuean, Nogay, Northern Sami (Latin), Norwegian, Occitan, Ossetic, Pashto, Persian, Polish, Portuguese, Punjabi (Arabic), Ripuarian, Romanian, Romansh, Russian, Sadri (Devanagari), Samoan (Latin), Sanskrit (Devanagari), Santali(Devanagiri), Scots, Scottish Gaelic, Serbian (Latin), Sherpa (Devanagari), Sirmauri (Devanagari), Skolt Sami, Slovak, Slovenian, Somali (Arabic), Southern Sami, Spanish, Swahili (Latin), Swedish, Tajik (Cyrillic), Tatar (Latin), Tetum, Thangmi, Tongan, Turkish, Turkmen (Latin), Tuvan, Upper Sorbian, Urdu, Uyghur (Arabic), Uzbek (Arabic), Uzbek (Cyrillic), Uzbek (Latin), Volapük, Walser, Welsh, Western Frisian, Yucatec Maya, Zhuang, Zulu

The following language is supported when training a document processing model and selecting **Unstructured and free-form documents** as document type: English

Requirements

Document processing works on input documents that meet the following requirements:

- JPG, PNG, or PDF format (text or scanned). Text-embedded PDFs are better, because there won't be any errors in character extraction and location.
- TIFF files can't be used for training. You'll need to use documents in PDF, JPG or PNG format to train a model. Once the model has been trained, it can extract data from TIFF files when the model is used in a Power Automate cloud flow.
- If your PDFs are password-locked, you must remove the lock before submitting them.
- Maximum document size to process must not exceed 20 MB.
- For images, dimensions must be between 50×50 and $10,000 \times 10,000$ pixels.
- If scanned from paper documents, scans should be high-quality images.
- You can create up to 200 collections per model.
- In a cloud flow, the limit of fields that can be tagged for document processing is 300.

 **Note**

- Extracting signatures from documents isn't currently supported.
- Fields that split across page boundaries aren't currently supported.

Optimization tip

Learn how to [improve the performance of document processing models](#).

Next step

[Create a form-processing model](#)

Create a document processing custom model

Article • 04/03/2023

After you review the [requirements](#), you can get started creating your document processing model.

Sign in to AI Builder

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. On the left pane, select **AI Builder > Explore**.
3. Select **Extract custom information from documents**.
4. Select **Get Started**.
5. A step-by-step wizard will walk you through the process by asking you to list all data you want to extract from your document. If you want to create your model by using your own documents, make sure you have at least five examples that use the same layout. Otherwise, you can [use sample data](#) to create the model.
6. Select **Train**.
7. Test the model by selecting **Quick test**.

Select the type of document

On the **Choose document type** step, select the type of document you want to build an AI model to automate data extraction. There are two options:

- **Structured and semi-structured documents.** Structured and semi-structured documents are those where for a given layout, the fields, tables, checkboxes, and other items can be found in similar places. Examples of structured and semi-structured documents are invoices, purchase orders, delivery orders, tax documents, and more.
- **Unstructured and free-form documents.** Unstructured documents are those where there's no set structure, usually free documents with a varying number of paragraphs. Examples of unstructured documents are contracts, statement of work, letters, and more.

Choose document type

Choose information to extract

Add collections of documents

Tag documents

Select the type of documents your model will process

Structured documents

Unstructured documents

Define information to extract

On the **Choose information to extract** screen, define the fields, tables, and checkboxes you want to teach your model to extract. Select the **+Add** button to start defining these.

Add

Field

Checkbox

Table

Recipient's / Lender's
Contoso, Ltd.
4567 Main St.
Buffalo, NY 90852

Mortgage Statement

1.Mortgage interest \$ 13,345.34	2.Ostanding mortgage \$ 764,321.33	3.Mortgage origination date 10/08/2019
4.Refund of overpaid \$ 15.01	5.Mortgage insurance \$ 1,222.05	
6.Address 1025 Fifth St. Sunnyvale, CA 27673		

Recipient's / Lender's TIN
750-01-1829

Payer's/Borrower's TIN
306-14-5298

Payer's/Borrower's name
Helena Wilcox

Street address
123 Main St Apt. 10

City or Town, state or province, country, and ZIP or foreign postal code
Brooklyn, NY 90852

8.Number of properties securing the mortgage
2

9.Other

- For each **Field**, provide a name you'd like the field to have in the model.
- For each **Checkbox**, provide a name you'd like the checkbox to have in the model. Define separate checkboxes for each item that can be checked in a document.
- For each **Table**, provide the name you'd like the table to have. Also, define the different columns that the model should extract.

Group documents by collections

A collection is a group of documents that share the same layout. Create as many collections as document layouts that you want your model to process. For example, if you're building an AI model to process invoices from two different vendors, each having their own invoice template, create two collections.

The screenshot shows the 'Add collections of documents' step in the Power Automate interface. On the left, a sidebar lists options: 'Choose information to extract 4 fields, 1 table', 'Add collections of documents' (which is selected and highlighted in blue), 'Tag documents', and 'Model summary'. The main area is titled 'Add collections of documents' and contains the instruction 'Add sample documents for your model to study. Put similar documents into the same collection.' Below this is a button 'Create a collection for each layout' with a help icon. A 'New collection' button is also present. At the bottom, there are 'Back', 'Analyze', and 'Add at least five sample documents for each collection' buttons. The top right corner shows 'My multiple invoices model' and 'Save and close' buttons. A 'Quick tips' section on the right provides advice on improving model performance and what analyze does.

For each collection that you create, you need to upload at least five sample documents per collection. Files with formats JPG, PNG, and PDF files are currently accepted.

This screenshot shows the 'Add collections of documents' step after creating two collections. The sidebar and main interface are identical to the previous screenshot. The 'Add documents' section now displays two collections: 'Adatum' and 'Contoso', both showing '0 documents'. The 'Quick tips' section remains the same, providing guidance on document analysis and performance.

(!) Note

You can create up to 200 collections per model.

Next step

[Tag documents in a document processing model](#)

See also

[Training: Process custom documents with AI Builder \(module\)](#)

Tag documents

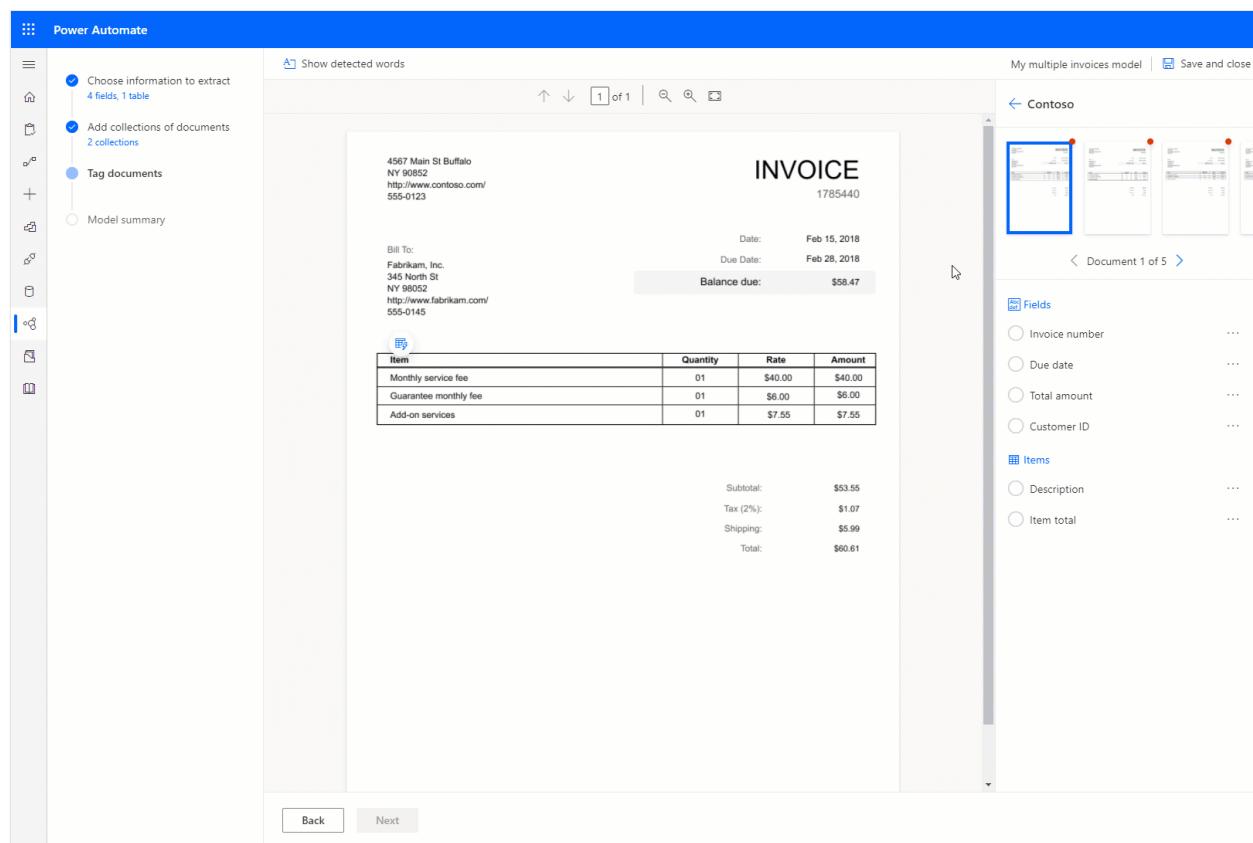
Article • 04/03/2023

By tagging the documents you've uploaded, you're teaching your AI Builder model to extract the fields and tables you've specified.

To start tagging, select one of the collections on the right panel.

Tag fields

To tag a field, draw a rectangle around the field you're interested in and select the field name that it corresponds to.



At any time, you can resize to adjust your selection.

If a field ends on one line and begins on another line, you can tag it using the '**Continue tagging**' option.

RENTAL LEASE AGREEMENT



This Rental Lease Agreement made this on 2nd of May, 2022 by and between Andre Lawson the “Landlord” and Preston Morales the “Tenant”. The Landlord and Tenant are collectively referred to in this Agreement as the “Parties”.

For the covenants contained herein, and other good and valuable consideration, the receipt and sufficiency of which is hereby acknowledged, the Parties agree as follows:

1. LEASE TERM. The term of this Agreement shall be a period of six (6) months, with a start date on 05-09-2022 and ending date on the day 11-07-2022 hereinafter known as the “Lease Term.”

2. PROPERTY. The leased premises shall be comprised of that certain personal residence (including both the house and the land) located at 4567 Main St Buffalo, NY 98052 (“Premises”). Landlord leases the Premises to Tenant and Tenant leases the Premises from Landlord on the terms and conditions set forth herein.

3. MONTHLY RENT. The rent to be paid by Tenant to Landlord throughout the term of

You can also use the **Control** key in Windows or **Command ⌘** in macOS to tag multiple words at once. You can also use the **Shift** key to select the first and last word to tag all words between the two.

When you hover over words in your documents, light blue boxes may appear. These indicate that you can draw a rectangle around those words to select a field.

4567 Main St Buffalo
NY 90852
<http://www.contoso.com/>
555-0123

Tag tables

1. Draw a rectangle around the table in the document you're interested in, and then select the table name that it corresponds to. The content of the panel on the right will change.
2. Draw *rows* by left-clicking between row separators.
3. Draw *columns* by pressing **Ctrl + left-click**.
4. Once the rows and columns have been set, assign the headers to extract by selecting the header column and mapping it to the desired one.
5. A preview of how the table will be extracted appears on the panel on the right.

6. If the header of the table has been tagged, select **Ignore first row** so the header of the table isn't extracted as the table content.

The following animation illustrates the process:

The screenshot shows the Power Automate interface for extracting data from a document. On the left, a sidebar lists actions: 'Choose information to extract' (4 fields, 1 table), 'Add collections of documents' (2 collections), 'Tag documents' (1 document tagged), and 'Model summary'. The main area displays an invoice document from 'Contoso'. The invoice header includes 'INVOICE' and the number '1785442'. Below the header, it says 'Date: Apr 15, 2020', 'Due Date: Apr 30, 2020', and 'Balance Due: \$56.28'. The 'Bill to:' section lists 'Fabrikam, Inc.' with address '345 North St NY 98052' and phone '555-0145'. A table details items: 'Monthly service fee' (Quantity 01, Rate \$40.00, Amount \$40.00), 'Guarantee monthly fee' (Quantity 01, Rate \$6.00, Amount \$6.00), and 'Add-on services' (Quantity 01, Rate \$3.30, Amount \$3.30). The subtotal is \$49.30. To the right, a sidebar titled 'Fields' shows checked options for 'Invoice number', 'Due date', and 'Total amount', along with unselected options for 'Customer ID' and 'Items'. Below the fields is a 'Tables' section with an 'Items' option. At the bottom of the main area are 'Back' and 'Next' buttons.

An alternative way to define the rows and columns for a table is by selecting **Delimit rows and columns** at the top of the screen.

The screenshot shows a document editor interface with a red box highlighting the 'Delimit rows and columns' button. A cursor is hovering over a grid icon. The document content includes an 'INVOICE' header with ID '1785442'. It lists 'Bill to: Fabrikam, Inc.' and provides dates: 'Date: Apr 15, 2020', 'Due Date: Apr 30, 2020', and 'Balance Due: \$56.28'. Below this is a table with four rows and four columns labeled A, B, C, and D. The table has row numbers 1 through 4 on the left. The first row contains column headers: 'Item', 'Quantity', 'Rate', and 'Amount'. The subsequent rows contain data: 'Monthly service fee' with quantity 01, rate \$40.00, and amount \$40.00; 'Guarantee monthly fee' with quantity 01, rate \$6.00, and amount \$6.00; and 'Add-on services' with quantity 01, rate \$3.30, and amount \$3.30. At the bottom, there are calculations: Subtotal: \$49.30, Tax (2%): \$0.99, Shipping: \$5.99, and Total: \$56.28. A tooltip at the bottom says 'Click to draw rows or Ctrl + Click to draw columns'.

	A	B	C	D
1	Item	Quantity	Rate	Amount
2	Monthly service fee	01	\$40.00	\$40.00
3	Guarantee monthly fee	01	\$6.00	\$6.00
4	Add-on services	01	\$3.30	\$3.30

Use the advanced tagging mode

Advanced tagging mode allows you to tag tables at the cell level. Use this mode for complex tables like:

- Tables that are skewed, where tagging with a grid isn't possible.
- When you need to extract nested items, like an item within a cell.

Given the table from the following example, to extract the unit price, we'll define it as a separate column on the [Choose information to extract step](#). We define **Description**, **Unit price**, **Quantity**, and **Amount** each as a column of the table and tag them accordingly using advanced tagging mode. See the animation below.

The screenshot shows the Power Automate interface with a document analysis model named "My multiple invoices model". On the left, there's a sidebar with options like "Choose information to extract", "Add collections of documents", "Tag documents", and "Model summary". The main area displays an invoice from "Contoso, Ltd" with the number 1785514. The invoice details include the date Jun 01, 2021, due date Jun 20, 2021, and balance due \$96.99. Below this is a table of items:

Item	Qty	Amount
Sideline towel Rate: \$12.24	03	\$36.72
Golf umbrella Rate: \$17.49	03	\$52.47
Ball markers 8-count Rate: \$2.97	01	\$2.97

On the right, there's a preview of five more invoices from "Contoso" and a sidebar for "Fields" and "Tables" with checkboxes for "Invoice number", "Due date", "Total amount", "Customer ID", and "Items".

You can start tagging in the default mode to quickly capture all rows and columns. Then switch to advanced mode to adjust each cell and tag nested items.

Nested items in tables

You can tag items that are nested within a row by defining these as columns. Given the table from the example below, to extract the unit price, define it as a separate column on the [Choose information to extract step](#) earlier in this topic. Define **Description**, **Unit price**, **Quantity**, and **Amount** each as a column of the table and then tag them accordingly.

The screenshot shows the Microsoft Power Automate interface with a document open for processing. On the left, there's a sidebar with options like 'Choose information to extract' (4 fields, 1 table), 'Add collections of documents' (1 collection), 'Tag documents', and 'Model summary'. The main area shows a bill from 'Fabrikam, Inc.' with a date of 'Apr 15, 2021' and a due date of 'Apr 30, 2021'. A table of items is listed with columns for Description, Quantity, and Amount. The table is being tagged, with several cells highlighted in green. The right panel shows a preview of the tagged table and a message: 'Tag the content of the table. Headers don't have to be tagged on the document. To start, click on a cell and tag it in the document.' A 'Done' button is at the bottom right.

Multipage tables

If a table spans across more than one page, make sure you tag it across all the pages. Once you've finished tagging the table on one page, select **This table continues on next page** and continue tagging the table on the following page. By doing this, you'll teach the model that the table can span across multiple pages.

The screenshot shows the Microsoft Power Automate interface with a document containing a table that spans multiple pages. The table is being tagged, with several cells highlighted in green. The right panel shows a preview of the tagged table and a message: 'This table continues on next page'. A red box highlights this message. A 'Done' button is at the bottom right.

Once you have tagged all the pages where the table is present. Select **Done** to mark the table as fully tagged.

You can tag complex tables that have merged cells, nested items, borders or no borders, or even unexpected elements like a coffee stain, as shown in the following example.

The screenshot shows the 'Document Processing' interface with the following details:

- Left sidebar:** Includes options like 'Choose document type' (Structured and semi-structured documents), 'Choose information to extract' (4 fields, 1 table), 'Add collections of documents' (1 collection), 'Tag documents', and 'Model summary'.
- Top bar:** Shows 'Show detected words', 'Delimit rows and columns', 'Document Processing 10/26/2', and 'Items'.
- Table area:** A table with columns: A, Description, Quantity, Rate, and Total. Rows 1 through 8 are listed. Row 8 contains a large brown coffee stain. A tooltip at the bottom says: 'Click to draw rows or Ctrl + Click to draw columns'. Below the table, it says: 'Subtotal: € 689,63', '(A 21%): € 144,82', and 'Total: € 843,45'.
- Right panel:** Titled 'Items' with columns: Description, Quantity, Rate. Rows 1 through 7 are listed. Row 7 has a red box around its 'Description' cell. A note says 'This table continues on next page'.
- Bottom buttons:** 'Back', 'Next', 'Tag all documents to continue (5 remaining)', and a 'Done' button which is highlighted with a red box.

💡 Tip

As a best practice, ensure that you upload documents for training with expected variations. For example, include samples where the entire table is on a single page and where tables span two or more pages.

Tag checkboxes

To tag a checkbox, draw a rectangle around the checkbox you're interested in extracting and select the checkbox name that it corresponds to.

The screenshot shows the Power Automate AI Builder interface. On the left, there's a sidebar with options like 'Choose information to extract' (checked), 'Add collections of documents' (checked), 'Tag documents' (unchecked), and 'Model summary' (unchecked). The main area displays a document titled 'DELIVERY ORDER' from 'WIDE WORLD IMPORTERS'. The document contains fields for Name (Audrey Dumoulin), Delivery Order Number (6011925), Address (12 Bahnhofstrasse, 87117 Zurich), Date (09/28/2021), Phone (555-0199), and Priority shipping (checkbox checked). Below these are two tables: one for items and one for products. The right panel shows the extracted data in a structured format with three red dots next to the 'Priority shipping' checkbox, indicating it was not available in the training documents.

If the quality of the document is low, AI Builder might not be able to detect the checkbox. If you're unable to tag a checkbox, do the following:

1. On the panel on the right, select the three dots next to the checkbox you want to extract.
2. Select **Not available in document**.

AI Builder supports detection and extraction of selection marks such as checkboxes and radio buttons, with different markers to indicate whether the selection is marked or not.

Field, checkbox, or table not in document

If a field, checkbox, or table isn't present in one of the documents you've uploaded for training, select **Not available in document** on the panel to the right, next to a field, checkbox, or table.

The screenshot shows the Power Automate interface with a document processing workflow. On the left, a sidebar lists steps: 'Choose information to extract' (4 fields, 1 table), 'Add collections of documents' (2 collections), and 'Tag documents'. The main area displays a scanned invoice from 'Contoso' with the following details:

INVOICE
1785440

Bill To:
Fabrikam, Inc.
345 North St
NY 98052
http://www.fabrikam.com/
555-0145

Date: Feb 15, 2018
Due Date: Feb 28, 2018
Balance due: \$58.47

Items

Item	Quantity	Rate	Amount
Monthly service fee	01	\$40.00	\$40.00
Guarantee monthly fee	01	\$6.00	\$6.00
Add-on services	01	\$7.55	\$7.55

Subtotal: \$53.55
Tax (2%): \$1.07
Shipping: \$5.99
Total: \$60.61

At the bottom, there are 'Back' and 'Next' buttons.

On the right, a panel titled 'My multiple invoices model' shows a list of fields with checkboxes indicating they have been mapped:

- Invoice number (checked)
- Due date (checked)
- Total amount (checked)
- Customer ID (unchecked)
- Items (checked)
- Description (checked)
- Item total (checked)

Tag all documents across all collections

All the documents that you've uploaded are presented for you to tag. If you've created multiple collections, tag all documents across all the collections.

If fields can't be tagged in your document, go to [Fields couldn't be loaded for this document](#) for possible resolutions.

Next step

[Extract dates and numbers](#)

See also

[Training: Process custom documents with AI Builder \(module\)](#)

Extract dates and numbers from documents (preview)

Article • 04/03/2023

[This topic is pre-release documentation and is subject to change.]

While many fields to be extracted are simple texts, there are cases where the information to extract is a date or a number including amounts.

Important

- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- This feature is in process of rolling out, and might not be available in your region yet.

Importing this data to a target system can be cumbersome, requiring significant custom conversion logic. Most of the import connectors and APIs only accept normalized dates in ISO 8601 format like `YYYY-MM-DD`. To learn more about date format, go to <https://www.iso.org/iso-8601-date-and-time-format.html>. They also only accept numbers using dot as a decimal separator without a thousands separator like `NNN.DD`.

We've added the ability to declare this type during the field creation step of the wizard and to choose a date or number convention (equivalent to a locale).

The screenshot shows the 'Choose information to extract' step of an AI Model setup. On the left, a sidebar lists options: 'Choose document type' (selected), 'Structured and semi-structured documents'; 'Choose information to extract' (selected); 'Add collections of documents'; 'Tag documents'; and 'Model summary'. The main area has a title 'Choose information to extract' with a sub-instruction: 'List all pieces of information that you want the AI model to extract from your documents. For example: Name, Address, Total amount. You'll tag them in the documents.' Below this is a button labeled '+ Add' with a dropdown arrow. At the bottom of the main area are 'Back' and 'Next' buttons. In the top right corner, there's a 'Document Processing 3/14/2023, 1:53:29 PM | Save and close' bar, a 'Quick tips' section, and two other sections: 'What information do I need to enter?' and 'What are number and date fields?'. The 'Get help or send feedback' section at the bottom right includes a 'Get help' button.

ⓘ Note

For each field, only one convention is allowed for a given field for all the collections of this model. For instance, if you extract a field amount by selecting **Use comma (,) as decimal separator**, the following text 1234,56 or 1 234,56 will be converted to 1234.56. Amounts with format 12,34,576.78 or 1,234.56 won't be converted.

During the extraction, the text will be automatically converted according to the convention provided. This converted value can be retrieved using the "YOURFIELDNAME value" result. This value will be empty If the conversion isn't possible. The original text can be retrieved using the "YOURFIELDNAME text" result.

Supported date formats

When defining the field, choose among **Year, Month, Day, Month, Day, Year, or Day, Month, Year**.

The following characters can be used as date delimiters: , - / . \. Whitespace can't be used as a delimiter. For example:

- 01,01,2020
- 01-01-2020
- 01/01/2020

The day and month can each be written as one or two digits, and the year can be two or four digits:

- 1-1-2020
- 1-01-20

If a date string has eight digits, the delimiter is optional:

- 01012020
- 01 01 2020

The month can also be written as its full or short name. If the name is used, delimiter characters are optional. However, this format may be recognized less accurately than others.

- 01/Jan/2020
- 01Jan2020
- 01 Jan 2020

Supported number formats

When defining the field, choose either **Use dot (.) as decimal separator** or **Use comma (,) as decimal separator**.

When the decimal separator is a dot (.), thousand separators can be omitted, and comma (,) or whitespace can be used. For example:

- 1234.56
- 1,234.56
- 1 234.56

When the decimal separator is a comma (,), thousand separators or whitespace can be omitted. For example:

- 1234,56
- 1 234.56

Next step

[Train and publish your document processing model](#)

See also

[Training: Process custom documents with AI Builder \(module\)](#)

Train and publish your document processing model

Article • 09/29/2022

After you create your document processing model, you can train, test, and publish it to make it available.

Train and validate your model

1. Select **Next** to check your selected form fields. If everything looks good, select **Train** to train your model.
2. When training is completed, select **Go to Details page** on the **Training complete** screen.

Quick-test your model

1. On the details page, select **Quick test**.
2. You can either drag a document or select **Upload from my device** to upload your test file. The quick-test should only take a few seconds before displaying the results.
3. Select **Start over** to run another test, or **Close** if you're finished.

Troubleshooting tips

If you have trouble training your model, try these suggestions:

- If you don't see your training document, go to [Training document isn't displayed on the document processing model details page](#) for a possible resolution.
- Make sure your data meets the guidelines listed in [document processing model requirements and limitations](#).
- Learn how you can [improve the performance of your document processing model](#).
- Download [sample material](#) and use it for testing.

Publish your model

If you're happy with your model, you can select **Publish** to publish it. When publishing is complete, your model is promoted as **Published** and is ready to be used. More information: [Publish your model in AI Builder](#)

After you've published your form-processing model, you can use it in a [Power Apps canvas app](#) or in [Power Automate](#).

Limits

The following applies to calls made per environment across document processing models including prebuilt models: receipt processing and invoice processing.

Action	Limit	Renewal period
Calls (per environment)	360	60 seconds

See also

[Use a document processing model in Power Automate](#)

[Use the form processor component in Power Apps](#)

Improve the performance of your document processing model

Article • 07/27/2022

If your model performance isn't where you want it to be, for example you're getting bad results or low confidence scores, there are some things you can try.

Interpret your model accuracy score

Interpret your accuracy score to identify what your model is struggling to extract. Model evaluations include recommendations for raising the score.

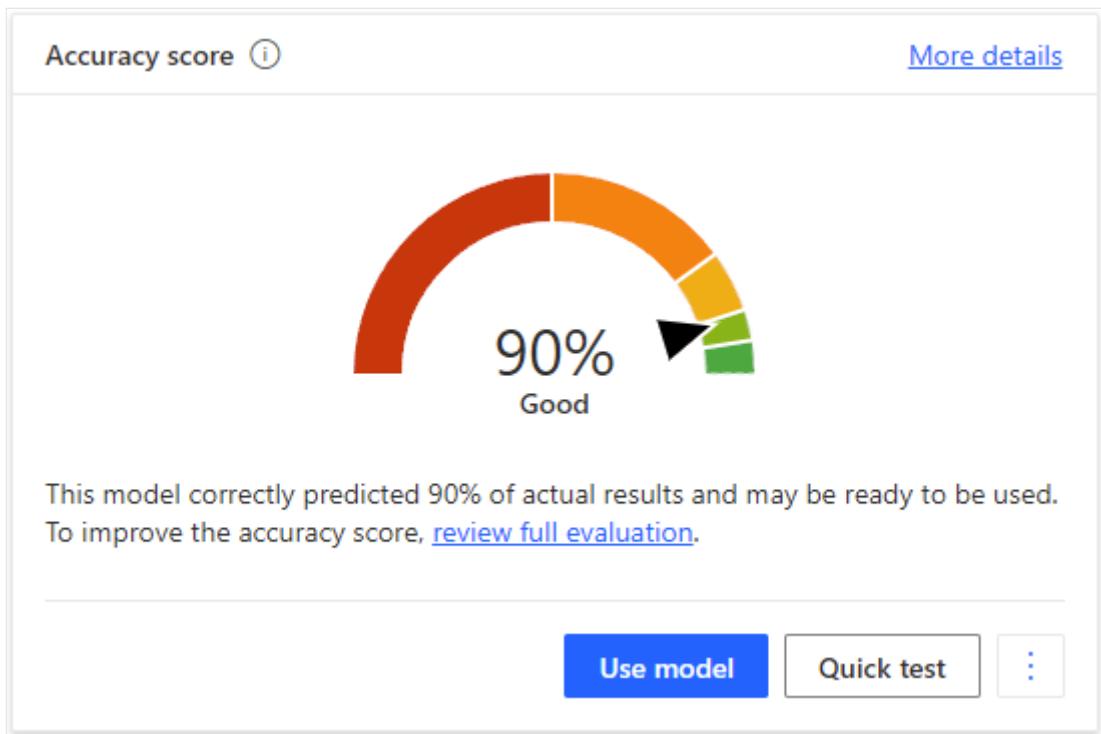
1. Sign in to [Power Apps](#) or [Power Automate](#).
2. In the left pane, select **AI Builder > Models**.
3. Open the document processing model you want to investigate. You should see the accuracy score.

Note

In the following cases you won't see accuracy scores for document processing models:

- If the model was trained selecting 'Unstructured and free-form documents' as document type. Currently, accuracy scores are only returned for models of type 'Structured and semi-structured documents'.
- Your model has been imported from another environment.
- If your model was trained before January 1, 2022. In this case, you can retrain it.

4. On the model details page, you should see the general accuracy score.



5. To get details, select **review full evaluation**.

Model evaluation
Published version

Overview **Collection** **Field** **Table** **Checkbox**

Training date
4/1/2022, 6:43:24 PM

Accuracy score
90%

Your model correctly predicted 90% of actual results. AI Builder calculates the accuracy score for your model based on prediction result of the test dataset. Before training, AI Builder separates your dataset into separate training data and testing data sets. And after training, AI Builder applies your AI model to the testing dataset to calculate the accuracy score.

Information to extract
Sales order number
Contact person
Vendor
<input checked="" type="checkbox"/> Priority shipping
<input type="checkbox"/> Line items

In this panel, you can navigate among different tabs to identify what your model is struggling to extract. You can browse the **Collection**, **Field**, **Table**, and **Checkbox** tabs to find what isn't processed properly.

Here's an example of the information in the **Field** tab.

The screenshot shows the Microsoft Power BI AI Builder interface. On the left, there's a sidebar with navigation options like Home, Action items, My flows, Create, Templates, Connectors, Data, Monitor, AI Builder, Explore, Models, Document automation, Process advisor, Solutions, and Learn. The main area displays the 'My Sales Order model' with an accuracy score of 90% (Good). Below the score, it says the model correctly predicted 90% of actual results and may be ready to be used. There are buttons for 'Use model' and 'Quick test'. To the right, there's a 'Model evaluation' section with a table showing information to extract: Sales order number, Contact person, Vendor, Priority shipping, and Line items. Another 'Model evaluation' window is open, showing a breakdown of accuracy scores for fields: Overall, Adatum, Contoso, and Wide. The 'Vendor' field has a low score of 66.

In this example, you'll want to improve the accuracy of the **Vendor** information.

The screenshot shows the 'Model evaluation' window with the 'Field' tab selected. It displays a message about improving the model by providing more examples in training data. Below is a table with columns: Name, Overall, Adatum, Contoso, and Wide. The 'Vendor' row is highlighted with a red box. A callout box over the 'Vendor' row contains the following text:

Very poor accuracy score

- Check that **Vendor** has been tagged correctly in all the documents.
- Provide more sample documents for training where **Vendor** is present.

Name	Overall	Adatum	Contoso	Wide ...
Sales order number	99	99	99	99
Contact person	99	99	99	99
Vendor	66	99	0	99

See suggestions on what you can do to improve your model by hovering over items with a poor accuracy score. For example, you might see a recommendation to [provide more sample documents](#) for training.

Common questions

[What can I do if I have a low accuracy score for a field, table, or checkbox?](#)

1. Check that the field, table, or checkbox has been tagged correctly in all the documents.
2. Provide more sample documents for training where the field, table, or checkbox is present.

What can I do if I have a low accuracy score for a collection?

Check that the documents within the collection all have the same layout. To learn more about collections, go to [Group documents by collections](#).

Add more documents to the training data

The more documents you tag, the more AI Builder will learn how to better recognize the fields. To add more documents, edit your document processing model and upload more documents. You'll find the option to edit the model on the details page of the model.

The screenshot shows the 'Contoso Invoices' model details page. At the top, there are buttons for 'Edit model', 'Share', 'Settings', and 'Delete'. Below that, it says 'Models > Contoso Invoices' and 'Form Processing • Published • Angie Andrews'. The main area is divided into three sections: 'Training document' (with a preview of an invoice form), 'Selected fields' (listing Invoice number, Due date, Total amount, and Customer ID), and 'How your model is used' (listing Power Automate and Power Apps).

More tips

- For filled-in forms, use examples that have all of their fields filled in.
- Use forms with different values in each field.
- If your form images are of lower quality, use a larger data set (10-15 images, for example).
- If possible, use text-based PDF documents instead of image-based documents. Scanned PDFs are handled as images.
- When you create a document processing model, upload documents with the same layout where each document is a separate instance. For example, invoices from different months should be in separate documents and not all in the same one.

- Documents that have different layouts should go [into different collections](#) when you upload samples for training.
- If the document processing model is extracting values from neighboring fields from the one you want the model to extract, edit the model and tag adjacent values that are being picked up incorrectly as being different fields. By doing this, the model will better learn the boundaries for each field.

See also

- [Invoice processing prebuilt model](#)
- [Receipt processing prebuilt model](#)

Use sample data to do document processing

Article • 09/19/2022

To explore the possibilities of document processing, you can get started by building and training a document processing model that uses sample invoices.

Get the sample data

Download sample documents for AI Builder document processing: [English version](#) or [Japanese version](#)

Guided learning experience

The [Get started with Document processing in AI Builder](#) Learn module walks you through each step of the model creation process using the sample data.

See also

[Use a document processing model in Power Automate](#)

[Use the form processor component in Power Apps](#)

FAQ for document processing

Article • 05/31/2022

This topic consists of frequently asked questions about the document processing model in AI Builder. If you don't find your question here, review the [overview of the document processing AI model](#) or submit your question to the [Power Automate Community for AI Builder](#).

Functionality

What can I do with document processing?

With document processing, you can build a custom AI model to extract information from documents. Document processing supports extracting data from structured, semi-structured documents like invoices, purchase orders, delivery orders, tax forms as well as unstructured documents such as contracts, statement of work, letters and others. Learn more: [Overview of the document processing model](#)

What types of documents can document processing handle?

Supported file types are PDF, JPG, and PNG.

What data can I extract from documents by using document processing?

document processing can extract fields, tables, and checkboxes from documents. Learn more: [Define information to extract](#)

Can I extract handwritten text from documents?

Yes. Document processing can extract printed and handwritten text from your documents.

How many samples do I need to train a document processing model?

For high-quality documents that use the same layout, five sample documents should be sufficient. For low-quality documents (for example, scans of poor quality), more sample documents might be necessary. To improve results, use 15 to 20 sample documents.

Can a single form-processing model extract information from documents that have different layouts or templates?

Yes. By using the collections feature, you train a single form-processing model to handle documents that have different layouts. Learn more: [Group documents by collections](#)

Can document processing handle *multiple* forms in a *single* document?

Each form needs to be in a separate file. For example, if you have a PDF document with multiple invoices in it, create a separate file for each invoice before you send it to the document processing model.

You can also specify pages for the document processing model to handle. This way you can take advantage of the model's functionality to loop page by page, and process one form at a time. Learn more: [Page range](#)

I've trained a document processing model but I'm not getting good results when it comes to extracted data. How can I improve the model?

If your model is returning poor results after you've trained it, edit the model and provide more samples for training. The more samples you provide, the more the AI model will learn how to extract data from your documents. Learn more: [Improve the performance of your document processing model](#)

Limitations

What is the maximum number of documents I can process?

You can process up to 360 documents per environment, every 60 seconds.

Can I use contracts and letters in my documents processing model?

Yes, unstructured documents like contracts and letters are supported by document processing.

Comparisons

What are the differences among document processing, invoice processing, receipt processing, identity document reader, business card reader, and text recognition?

Depending on your situation, you might need to use a particular model or a combination of them.

Use [text recognition](#) when you want to extract all the text present in an image or a PDF. You can then, for example, search for a keyword in the text that's extracted or build some fixed rules to extract certain items.

If you want to extract information from invoices, receipts, passports, driver's licenses, or business cards, start with the corresponding prebuilt model:

- [Invoice processing](#)
- [Receipt processing](#)
- [Identity document reader](#) (passports and driver's licenses)
- [Business card reader](#)

You can use these prebuilt models immediately, without having to create a new model. These models can extract common information found in their corresponding document type.

For any other document type, you can create a custom document processing model to extract the fields and tables you need. This also applies if you need additional information not provided by the prebuilt model. Learn more: [Custom document processing model](#)

What is the difference between AI Builder document processing and Azure Form Recognizer?

AI Builder document processing is built on top of Azure Form Recognizer. This provides both products with the latest advancements in Microsoft AI.

- AI Builder is part of [Microsoft Power Platform](#). This enables anyone to add AI into apps and automation with an easy-to-use interface. *You don't need to be a developer or data scientist.*
- [Azure Form Recognizer](#) is *targeted to professional developers*. They can use simple REST APIs to add AI capabilities to their custom code solutions.

Cost options

How much does AI Builder document processing cost?

You can start trying out document processing for free by starting a [trial](#). After you've evaluated it, you'll need to purchase AI Builder credits to use document processing. Every page you process with document processing will consume AI Builder credits. AI Builder credits can be purchased through AI Builder add-ons. Learn more: [AI Builder licensing](#)

Transition to the newest document processing model version

Article • 06/23/2022

AI Builder is improving its document processing model with a new model version. Your document processing preview models created before March 5, 2020 are deprecated. For more information, go to [Document processing and object detection preview models in AI Builder are deprecated](#)

Starting March 5, 2020, you have to re-create your existing document processing models if you want to continue to use them. Models created after this date will automatically use the new model version. It's a good idea to transition to the new model version as soon as possible.

What do I need to do?

You must re-create document processing models created before March 5, 2020

1. Identify any document processing models created before March 5, 2020, that you want to continue using.
2. For each model, [create a new document processing model](#) with the same training documents you used in the original model.
3. After the new model is trained and published, make sure you update any apps or flows to use the new model.
4. After you're done with your deprecated document processing model, you can delete it.

We understand you might have spent significant time building your document processing models. [Contact us](#) if you need help with migrating larger complex models.

Why this change?

Document processing models are being upgraded for general availability. Your existing preview models won't be compatible with the new model version.

Until March 5, 2020, document processing models stored documents as attachments in the Note table. With this update, these models now use the new File and Image data types in Microsoft Dataverse, which enables a better and more optimized usage of capacity.

See also

[Use a document processing model in Power Automate](#)

[Use the document processor component in Power Apps](#)

Bring your own AI model into AI Builder

Article • 12/13/2022

You can bring your own model into AI Builder so that it can function like any AI Builder custom model. You can use your model in Microsoft Power Platform by using Power Automate, or you can build apps with Power Apps.

When you use your own model, it's sometimes referred to as a *model endpoint*, which enables communication. When you use your own model, limitations apply. These [limitations](#) are described later in this article.

Create your own model

Outside of AI Builder, you can create your own model by using the Azure Machine Learning platform. To use the model in AI Builder, it must meet certain requirements:

- Your model contains an API definition that adheres to the OpenAPI specification (also known as Swagger).
- You've registered your model in AI Builder by using a Python package.

Register your own model

The first step in bringing your own model into AI Builder is to register it. Follow the procedure in [Bring your own model tutorial](#) (on GitHub).

After you register the model, you'll see it in the list of AI Builder models. On the model details page, the **Model source** will be **Imported** to show that the external model is registered to AI Builder by using your imported model endpoint.

The screenshot shows the AI Builder interface with the 'Models' tab selected. On the left sidebar, 'AI Builder' is expanded, and 'Models' is selected. In the main content area, a model named 'pneumonia-detection-v1' is displayed. The 'Model source' field in the 'Details' section is highlighted with a red box. The 'Input fields' and 'How your model is used' sections are also visible.

Limitations

- The only supported authentication mechanism is [API keys](#) that use [Azure Machine Learning](#).
- Only Swagger 2.0 is supported.
- The maximum allowed batch size is 500 rows.
- The maximum allowed latency/timeout is 20 seconds.
- The supported OpenAPI data types are:
 - Integer
 - Number
 - Boolean
 - String
- If your model takes an image as an input in Base64, it can be used for real-time prediction only, for consumption in [Power Automate](#) or [Microsoft Power Fx](#). Batch prediction isn't supported.
 - The name of the field must end with **image** (not case-sensitive).
 - The data type must be **String**.

You're now ready to use your own model in AI Builder. You can perform application lifecycle management tasks such as export your own model by using a solution, import your model into the target environment, and upgrade your model in source or target environments.

See also

[Package your own model by using solutions](#)

Package your own model using solutions

Article • 12/13/2022

After you've [registered](#) your own model in AI Builder using a Python package, you can package your model using solutions to perform application lifecycle management (ALM) tasks, such as:

- Export your own model using a solution.
- Import your own model into the target environment.
- Upgrade your own model in source or target environments.
- Stage your own model for upgrade.

Solutions are used to transport apps and components from one environment to another. They're also used to apply a set of customizations to existing apps. For detailed information about solutions and how they're used to apply ALM concepts across Microsoft Power Platform, go to [Overview of application lifecycle management](#).

Export your own model using a solution

In this example, you'll export two solutions:

- The solution that contains only the custom connector
 - The solution that contains the model, connection reference, and any other components
1. Make sure you've registered your own model in the source environment using a Python package. If you haven't already registered it, follow the procedure in [Bring your own model tutorial](#) (on GitHub).
 2. Create a solution to store the custom connector that's associated with the model:
 - a. Sign in to [Power Apps](#) or [Power Automate](#).
 - b. On the left pane, select **Solutions > New solution**.
 - c. In the **New solution** dialog, complete the fields. For detailed information about each field, go to [Create a solution](#).
 - d. Select **Create**.

New solution X

Display name *

Name *

Publisher *

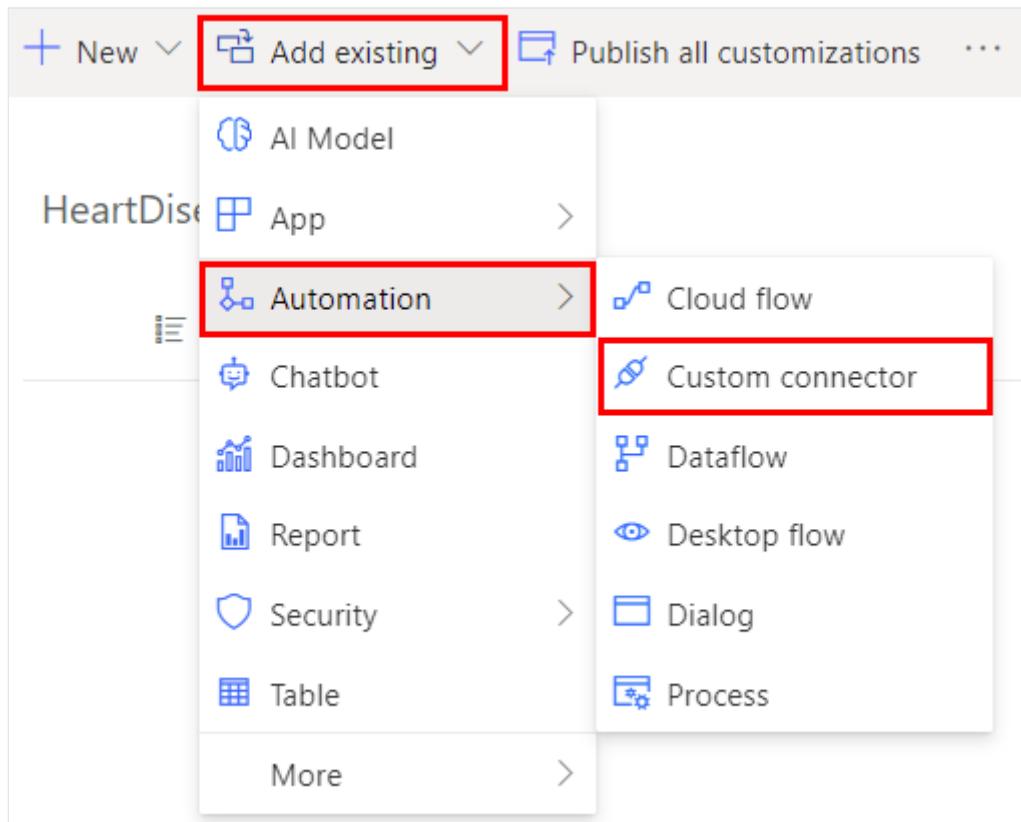
▼ edit

+ New publisher

Version *

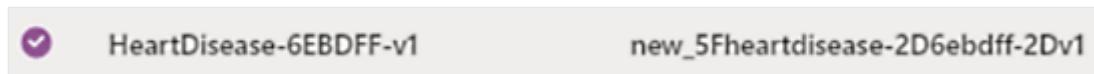
More options ▾

3. Add an existing custom connector by selecting **Add existing > Automation > Custom connector.**



4. Select the custom connector that's associated with the model.

The name will be in this format: <model_name>-<random number>-v<version>. The version helps distinguish which connector to add when you update the model.



5. Export the solution. For help on how to export solutions, go to [Export solutions for Power Apps](#) or [Export solutions for Power Automate](#).
6. Create a new solution to add the model and any other components:
 - a. Select **Solutions** > **New solutions**.
 - b. In the **New solution** dialog, complete the fields.
 - c. Select **Create**.

New solution X

Display name *

Name *

Publisher *

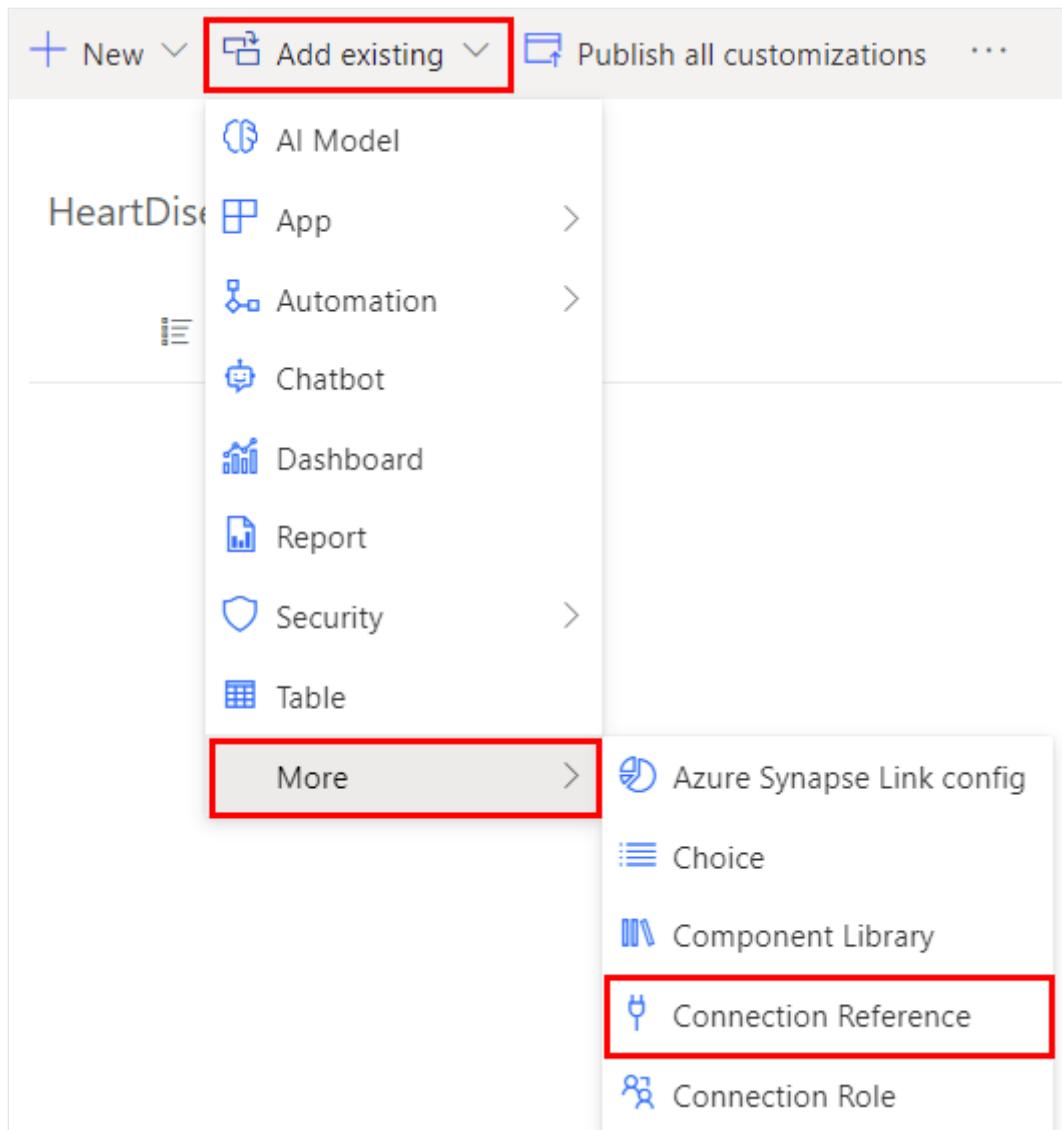
▼ edit

+ New publisher

Version *

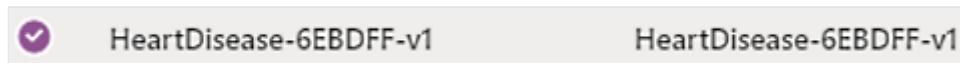
[More options](#) ▾

7. Add an existing connection reference by selecting **Add existing** > **More** > **Connection Reference (preview)**.

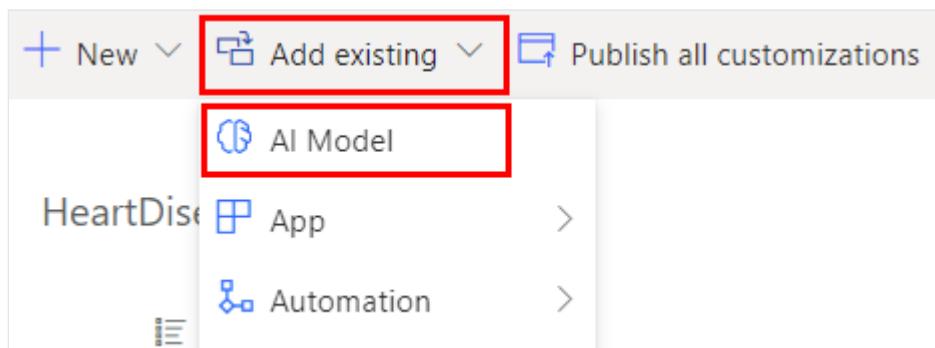


8. Add the existing connection reference associated with the model.

The naming format is the same as the custom connector in step 4:
<model_name>-<random number>-v<version>.



9. Add the AI model by selecting **Add existing > AI Model**.



Your solution should contain the components shown in the following image.

All	Display name ↑ ↓	Name ↓	Type ↓
HeartDisease	HeartDisease	HeartDisease	AI Model
HeartDisease-6EBDFF-v1	HeartDisease-6EBDFF-v1	HeartDisease-6EBDFF-v1	Connection Reference
HeartDisease-6EBDFF-v1	new_5Fheartdisease-2D6ebdff-2Dv1	new_5Fheartdisease-2D6ebdff-2Dv1	Custom Connector

10. (Optional) Add any components you want. In this example, we'll add a cloud flow that uses the model.

All	Display name ↑ ↓	Name ↓	Type ↓
HeartDisease	HeartDisease	HeartDisease	AI Model
HeartDisease-6EBDFF-v1	HeartDisease-6EBDFF-v1	HeartDisease-6EBDFF-v1	Connection Reference
HeartDisease-6EBDFF-v1	new_5Fheartdisease-2D6ebdff-2Dv1	new_5Fheartdisease-2D6ebdff-2Dv1	Custom Connector
Microsoft Dataverse	Microsoft Dataverse	Microsoft Dataverse	Connection Reference
Run Prediction	Run Prediction	Run Prediction	Cloud Flow

11. Export the solution. For help on how to export solutions, go to [Export solutions for Power Apps](#) or [Export solutions for Power Automate](#).

Import your own model into the target environment

In this example, you'll import two solutions that you created in the previous section:

- The solution that contains only the custom connector
- The solution that contains the model, connection reference, and any other components

For help on how to import solutions, go to [Import solutions for Power Apps](#), or [Import solutions for Power Automate](#).

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. Import the solution containing only the custom connector.
3. Import the solution containing the model, connection reference, and any other components.
4. During the import process, create the connection to your external endpoint:

- If your endpoint is not secured, select **Create**.
- If your endpoint is secured with an API key, enter it in the **API Key** field, and then select **Create**.

Connections

Re-establish connections to activate your solution.

2 updates needed

Sign in

These services use your credentials to sign into apps and create connections. A green check means you're ready to go.

*Connection name
HeartDisease-6EBDFF-v1

*API Key ⓘ
.....

Create **Cancel**

Your model is now ready to be consumed in your target environment.

Solution "HeartDisease_Connector" imported successfully.
Solution "HeartDisease_Model" imported successfully.

Solutions

Display name	Name	Created ↓	Version	Managed externally?
HeartDisease_Model	HeartDisease_Model	7/27/2021	1.0.0.1	✗
HeartDisease_Connector	HeartDisease_Conne...	7/27/2021	1.0.0.1	✗

If you experience issues with your import, go to [Troubleshooting solution import](#).

Upgrade your own model

Upgrade your solution to the latest version number so that it rolls up all previous patches in one step. Any components that were associated with the previous version of the solution and aren't in the newer version will be deleted. This ensures that outdated components won't be included in the solution.

In the examples, you'll upgrade two solutions:

- The solution that contains only the custom connector

- The solution that contains the model, connection reference, and any other components

For help on how to upgrade, go to [Upgrade or update a solution](#).

Upgrade your own model in the source environment

1. Make sure you've registered your upgraded model in the source environment by using a Python package. If you haven't already registered it, follow the procedure in [Bring your own model tutorial](#) (on GitHub).
2. Sign in to [Power Apps](#) or [Power Automate](#).
3. In the solution that contains only the custom connector, add a new existing custom connector by selecting **Add**.
4. Complete the fields and enter the new version. In this example, it's v2 because this is the second version.
5. Remove the older version of the custom connector from the solution:
 - a. Select the vertical ellipsis (:) next to the older version.
 - b. Select **Remove > Remove from this solution**.

The screenshot shows the Microsoft Power Platform Admin Center interface. At the top, a green banner displays the message: "Solution 'HeartDisease_Connector' imported successfully." Below this, there are two sections: "Solutions" and "All".

Solutions:

Display name	Name	Created	Version
HeartDisease_Connector	HeartDisease_Conne...	7/27/2021	1.0.0.1

All:

Display name	Name	Type
HeartDisease-569F27-v2	new_5Fheartdisease-2D569f27-2Dv2	Cus...
HeartDisease-6EBDFF-v1	new_5Fheartdisease-2D6ebdff-2Dv1	Cus...

In the "All" table, the row for "HeartDisease-6EBDFF-v1" is selected and highlighted with a red box. A context menu is open over this row, also enclosed in a red box. The menu items are: "Edit", "See solution layers", "Remove" (with a sub-menu "Remove from this solution"), and "Delete from this environment".

6. Export the solution by selecting **Export**. By default, the version will be incremented during export.

7. In the solution that contains the model, connection reference, and any other components, update the connection reference to point to the latest version:
 - a. The custom connection will have the same name as the custom connector you added in step 4. In this example, v2 is the latest version.
 - b. Remove the older version of the connection reference from the solution by selecting the name to remove (in this example v1), and then selecting **Remove > Remove from this solution**.
8. Export the solution by selecting **Export**. The AI model and any component referencing the model are automatically updated to point to the latest version.

Upgrade your own model in the target environment

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. Import the solution that contains only the connector by selecting **Import > Upgrade**.
3. Import the solution that contains the model, connection reference, and any other component by selecting **Import > Upgrade**.
4. Create new connections to the new endpoint:
 - a. If your endpoint is not secured, select **Create**.
 - b. If your endpoint is secured with an API key, enter it in the **API Key** field, and then select **Create**.

Connections
Re-establish connections to activate your solution.

1 updates needed

Sign in
These services use your credentials to sign into apps and create connections. A green check means you're ready to go.

HeartDisease-569F27-v2

*Connection name

*API Key ⓘ

Create **Cancel**

The model now will reference the new endpoint in the target environment.

Upgraded solution successfully.
Solution "HeartDisease_Model" imported successfully.
Solutions

Display name	Name	Created ↓	Version	Managed externally?
HeartDisease_Model	HeartDisease_Model	7/27/2021	1.0.0.2	⋮
HeartDisease_Connector	HeartDisease_Connector	7/27/2021	1.0.0.2	⋮

Stage your own model for upgrade

Staging your model for upgrade only imports it into the organization. It doesn't make it active. In this scenario, components that reference the model in the environment will reference the old version until you complete this procedure by applying the upgrade.

For help on how to import solutions, go to [Import solutions for Power Apps](#) or [Import solutions for Power Automate](#).

1. Sign in to [Power Apps](#) or [Power Automate](#).
2. Import the solution that contains only the connector by selecting **Import > Upgrade**.
3. Import the solution that contains the model, connection reference, and any other components by selecting **Import > Stage for Upgrade**.
4. Create new connections to the new endpoint:
 - a. If your endpoint is not secured, select **Create**.
 - b. If your endpoint is secured with an API key, enter it in the **API Key** field, and then select **Create**.

Connections

Re-establish connections to activate your solution.

1 updates needed

Sign in

These services use your credentials to sign into apps and create connections. A green check means you're ready to go.

The screenshot shows a configuration dialog for a connection named "HeartDisease-569F27-v2". The "Connection name" field contains the value "HeartDisease-569F27-v2". The "API Key" field is redacted with several dots. At the bottom are two buttons: "Create" and "Cancel".

You'll see two versions of the solution: the *base* and the *upgrade*. Any component in the environment outside of these solutions will still reference the old version of the model. Components inside of these solutions will reference the version of the model contained in the corresponding solution.

In the following example, the cloud flow in the base solution references the old version and the cloud flow in the new solution references the new version.

Solutions				
Display name	Name	Created ↓	Version	Managed externally?
HeartDisease_Model	HeartDisease_Model_Upgrade	7/27/2021	1.0.0.2	✗
HeartDisease_Model	HeartDisease_Model	7/27/2021	1.0.0.1	✗
HeartDisease_Connector	HeartDisease_Connector	7/27/2021	1.0.0.2	✗

5. To upgrade the old version, select **Apply upgrade**.

Solutions				
Display name	Name	Created ↓	Version	Managed externally?
HeartDisease_Model	HeartDisease_Model	7/27/2021	1.0.0.2	✗
HeartDisease_Connector	HeartDisease_Connector	7/27/2021	1.0.0.2	✗

All components in the environment now reference the new version of the model.

See also

[Overview of application lifecycle management](#)

[ALM basics](#)

[Import solutions](#)

[Upgrade or update a solution](#)

AI Builder in Power Automate overview

Article • 01/05/2023

Power Automate is a service that helps you create automated workflows between your favorite apps and services to synchronize files, get notifications, collect data, and more.

More information: [Power Automate docs](#)

ⓘ Note

For any AI Builder action in a cloud flow, ensure that you leave the default **Asynchronous Pattern** setting to **On**. This will ensure that results from the AI models are returned properly.

Prebuilt AI models you can use in Power Automate right away

- [Business card reader model](#)
- [Category classification model](#)
- [Entity extraction model](#)
- [ID reader model \(preview\)](#)
- [Key phrase extraction model](#)
- [Language detection model](#)
- [Receipt processing model \(preview\)](#)
- [Sentiment analysis model](#)
- [Text recognition model](#)
- [Text translation model](#)

Custom AI models that you build and train

- [Category classification model](#)
- [Entity extraction model](#)
- [Document processing model](#)
- [Object detection model](#)
- [Prediction model](#)

Predict action

You can use the predict action in Power Automate with many different model types. For more information, see [Use the predict action in Power Automate](#).

Next step

[Prerequisites](#)

See also

- [Feature availability by region](#)
- [AI Builder in Power Apps overview](#)
- [Training: Use AI Builder in Power Automate \(module\)](#)
- [Training: Improve business performance with AI Builder \(learning path\)](#)

Prerequisites for using AI Builder in Power Automate

Article • 07/08/2022

Before you can use AI Builder in Power Automate, you need:

- An account with access to [Power Automate](#)
- A trained [AI Builder model](#) or an AI Builder [prebuilt model](#).

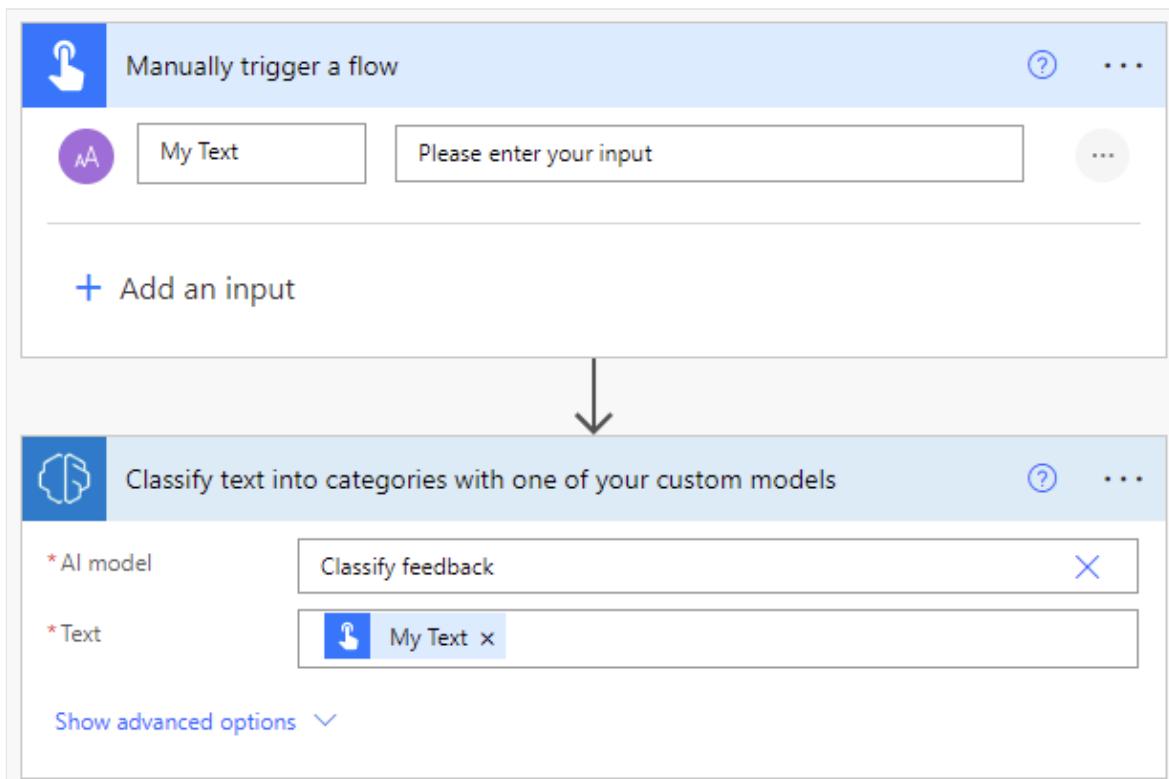
See also

[AI Builder prerequisites](#)

Use a category classification custom model in Power Automate

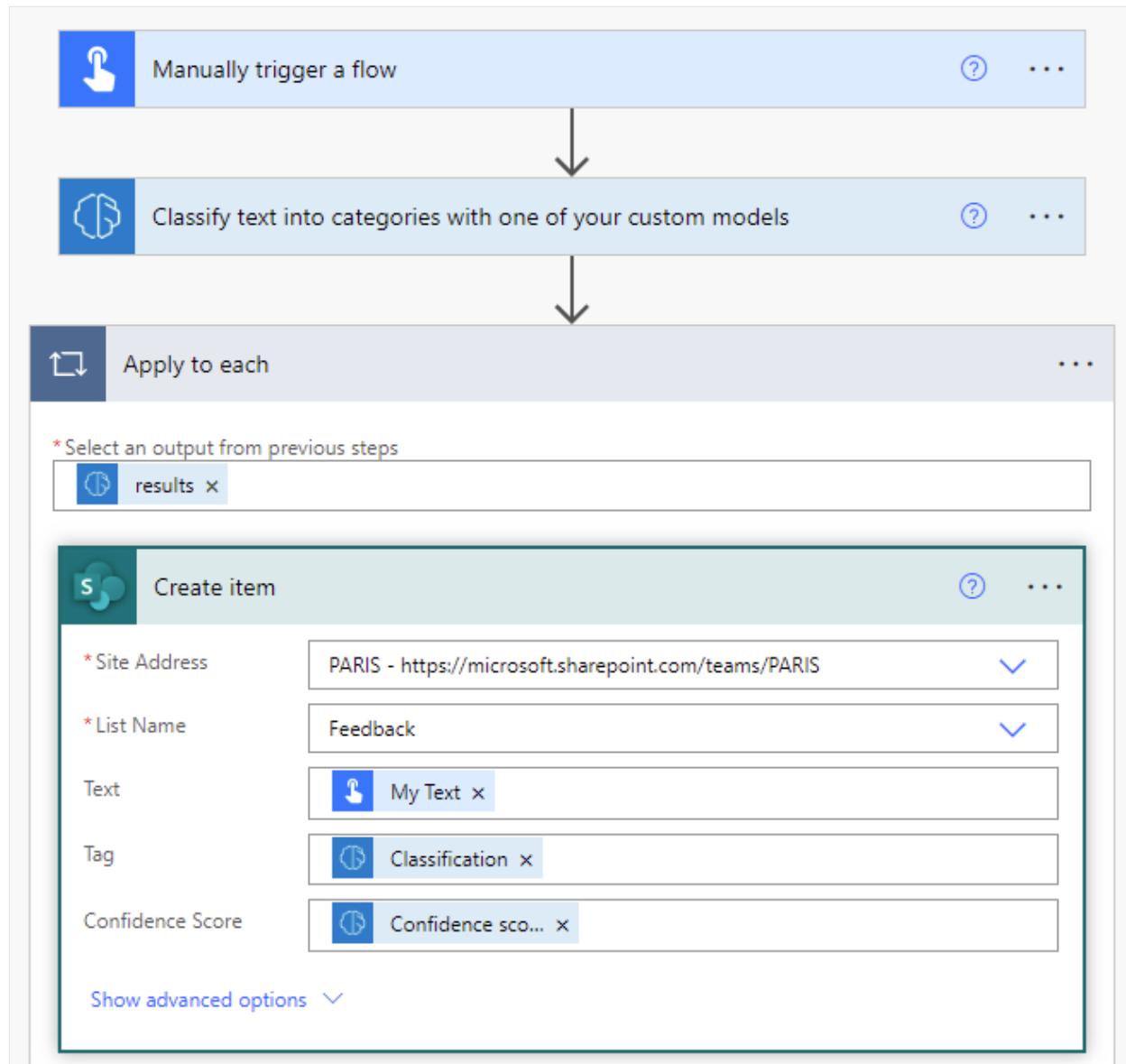
Article • 01/05/2023

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
5. Replace the word **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Classify text into categories with one of your custom models** in the list of actions.
7. Select the category classification model you want to use, and in the **Text** column add **My Text** from the trigger.



8. In the successive actions, use any columns and tables extracted by the AI Builder model.

The following example saves each inferred Classification and Confidence score into a list created with Microsoft Lists in SharePoint.



Congratulations! You've created a flow that uses an AI Builder category classification model. Select **Save** on the top right, and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description	Values
AI model	Yes	model	Category classification model to use for analysis	Trained and published category classification model
Text	Yes	string	Text to analyze	Text sentences

Name	Required	Type	Description	Values
Language	Yes	string	Language of the text to analyze	"Detect automatically" or language code (ex.: "en", "fr", "zh_chs", "ru")

Output

Name	Type	Description	Values
Classification	string	Table identified	Issues, compliment, customer service, documentation, price & billing, staff
Confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted value is accurate

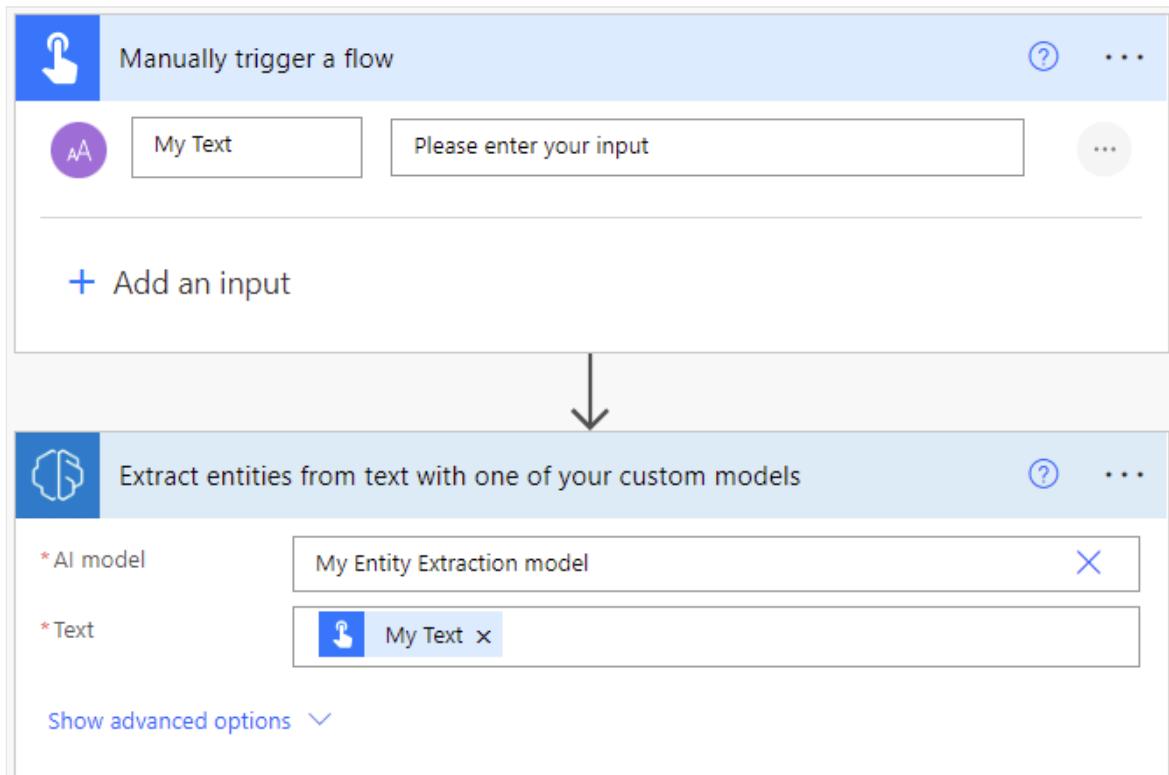
See also

- [Overview of the category classification custom model](#)
- [Training: Get started with AI Builder category classification \(module\)](#)

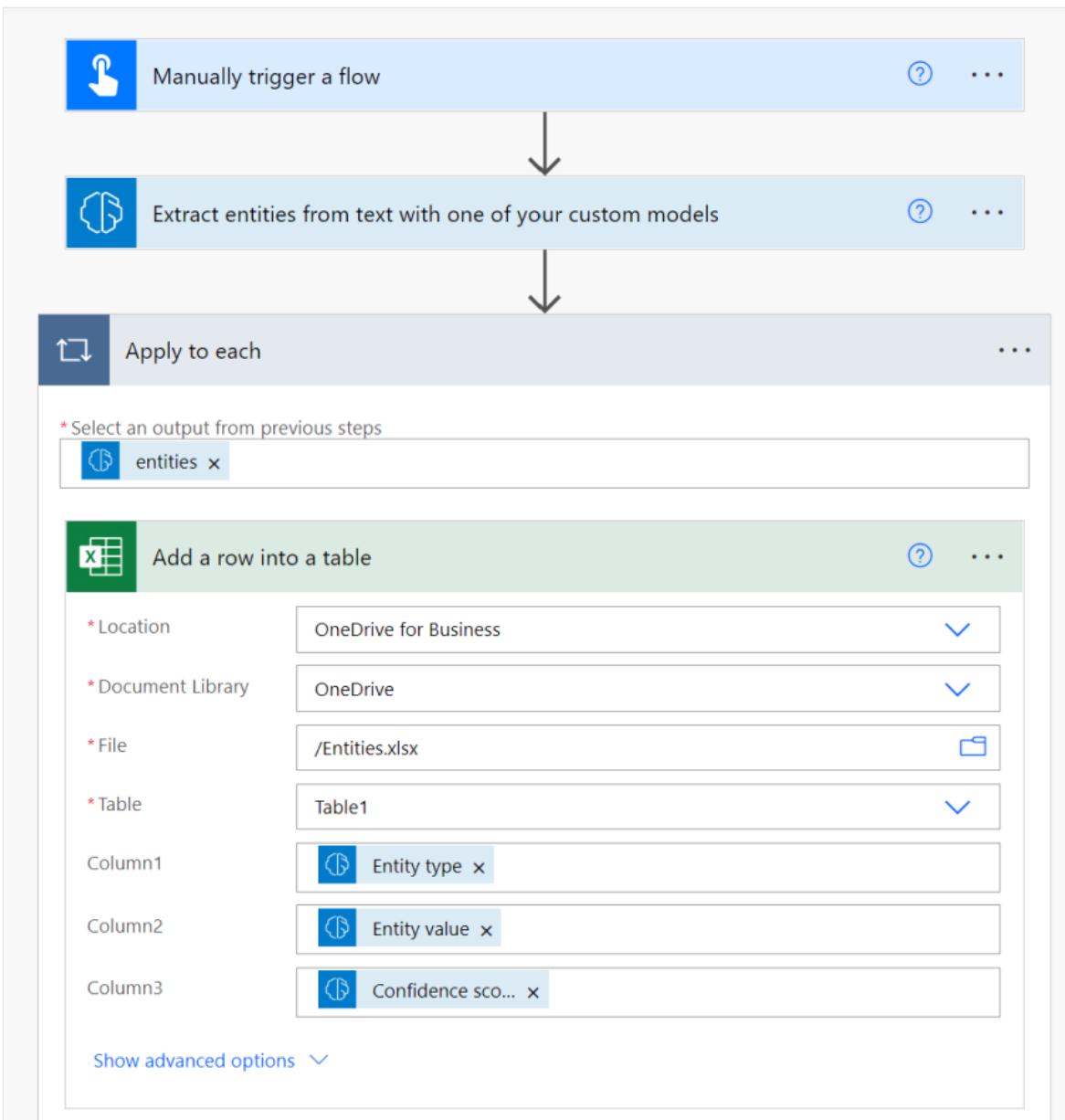
Use an AI Builder custom entity extraction AI model in Power Automate

Article • 12/13/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
5. Replace the word **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Extract entities from text with one of your custom models** in the list of actions.
7. Select the entity extraction model you want to use, and in the **Text** column add **My Text** from the trigger.



8. In the successive actions, you can use any columns and tables extracted by the AI Builder model. The following example saves each inferred **Entity type**, **Entity value** and **Confidence score** into an Excel table.



Congratulations! You've created a flow that uses an AI Builder entity extraction model. Select **Save** on the top right, and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description	Values
AI model	Yes	model	Entity extraction model to use for analysis	Trained and published entity extraction model
Text	Yes	string	Text to analyze	Text sentences

Name	Required	Type	Description	Values
Language	Yes	string	Language of the text to analyze	"Detect automatically" or language code (ex.: "en", "fr", "zh_chs", "ru")

Output

Name	Type	Description	Values
Entity type	string	Type of the entity	Example: DateTime or Organization
Entity value	string	Content of the entity	Example: June 1 or Contoso
Confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted value is accurate
Starting location	integer	Where the entity's first character appear in the line	
Character count	integer	How long the entity is	

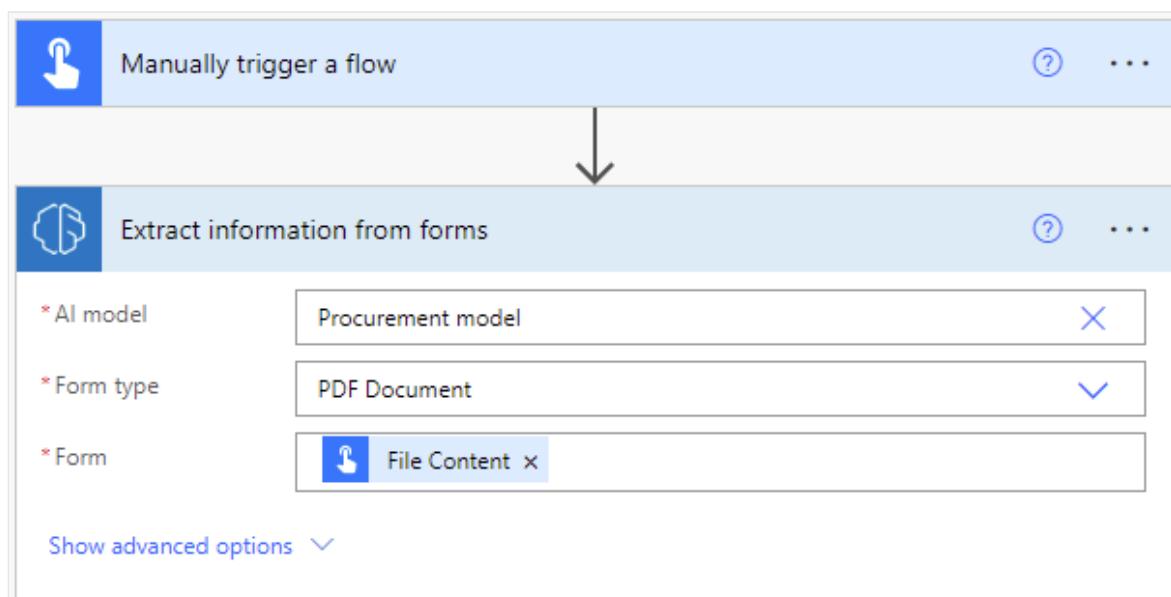
See also

[Entity extraction model overview](#)

Use a document processing model in Power Automate

Article • 01/05/2023

1. Sign in to [Power Automate](#).
2. Select **My flows** > **New flow** > **Instant cloud flow**.
3. Enter a name for your flow.
4. Under **Choose how to trigger this flow**, select **Manually trigger a flow**, and then select **Create**.
5. Expand **Manually trigger a flow**, and then select **+Add an input** > **File** as the input type.
6. Select **+New step** > **AI Builder**, and then select **Extract information from forms** in the list of actions.
7. Select the document processing model you want to use, and then select the document type.
8. In the **Form** field, add **File Content** from the trigger.

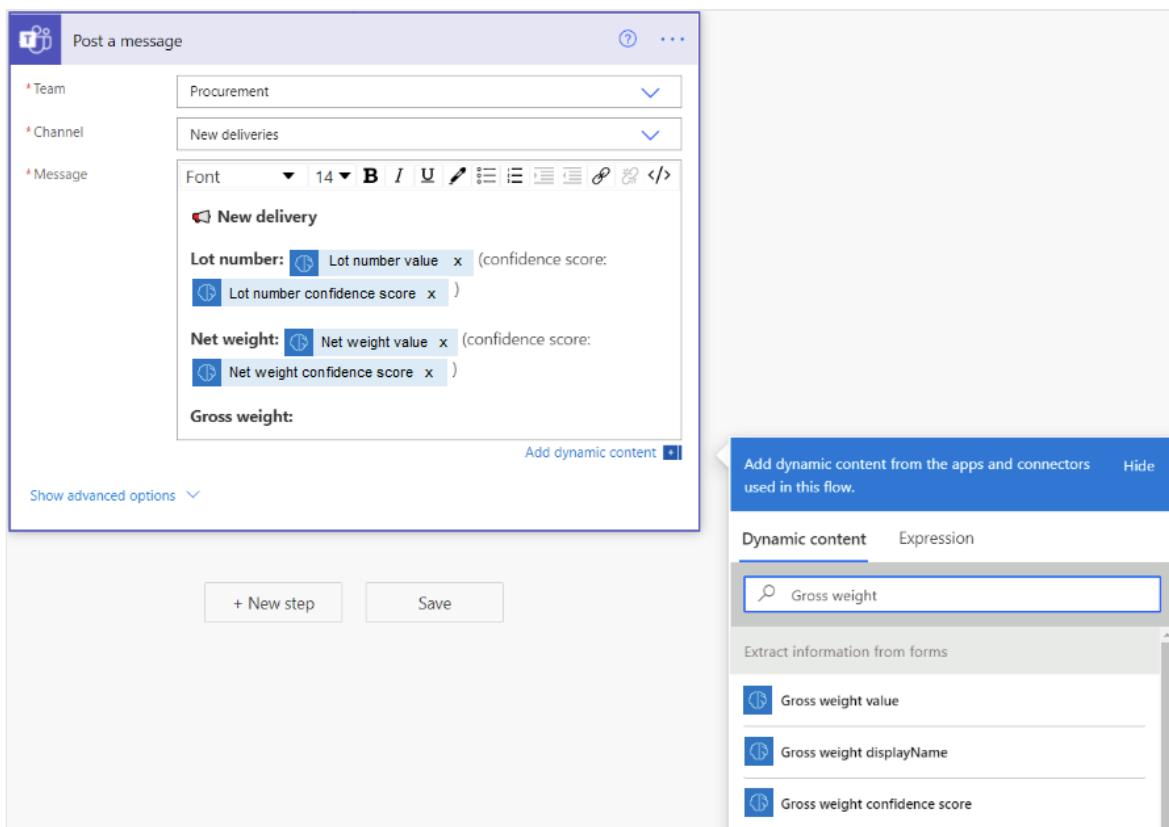


9. In the successive actions, you can use any of the fields and tables extracted by the AI Builder model. For example, let's say that our model is trained to extract the *Lot number*, the *Net weight*, and the *Gross weight* values. We also want to post these to a Microsoft Teams channel after AI Builder has extracted them from the document.

Just add the **Post a message** action from the Microsoft Teams connector, and then select your fields from the list of tokens.

ⓘ Note

- To retrieve the value for a field, select **<field_name> value**. For example, for the *Lot number* field, select **Lot number value**.
- To retrieve the value for a checkbox, select **<checkbox_name> value**. For example, for a checkbox named *Priority shipping*, select **Priority shipping value**. The return value is of type Boolean: **true** if the checkbox is marked as selected in the document, **false** if it's not.
- To retrieve the confidence score for an extracted item, select **<field_name> confidence score**. For example, for the *Lot number* field, select **Lot number confidence score**.



Congratulations! You've created a flow that uses an AI Builder document processing model. Select **Save** on the top right, and then select **Test** to try out your flow.

Page range

For documents that have multiple pages, it's possible to specify the page range to process.

The screenshot shows the configuration interface for the 'Extract information from forms' action. At the top, there's a blue header bar with a hand icon and the text 'Manually trigger a flow'. Below it is another blue header bar with a document icon and the text 'Extract information from forms'. A large downward arrow is positioned between these two bars. The main area contains several input fields:

- *AI model: Procurement model
- *Form type: PDF Document
- *Form: File Content
- Pages: Enter page range (Ex: 1 or 3-5). Only returns results of a single form.

At the bottom left, there's a link 'Hide advanced options' with a collapse/expand arrow.

You can enter a page value or page range in the **Pages** parameter. Example: 1 or 3-5.

ⓘ Note

If you have a large document with only one form, we strongly recommend you use the **Pages** parameter. Doing this can reduce the cost of model prediction, which can increase performance. However, the page range should contain a unique form for the action to return correct data.

Example: A document contains a first form in page 2 and a second form that spans over pages 3 and 4:

- If you enter page range 2, it will return the data of the first form.
- If you enter page range 3-4, it will only return the data of the second form.
- If you enter page range 2-4, it will return partial data of first and second form (should be avoided).

Parameters

Input

Name	Required	Type	Description	Values
AI Model	Yes	model	Document processing model to use for analysis	Trained and published document processing models

Name	Required	Type	Description	Values
Document type	Yes	list	The file type of the form to analyze	PDF Document (.pdf), JPEG Image (.jpeg), PNG Image (.png)
Form	Yes	file	Form to process	
Pages	No	string	Page range to process	

Output

Name	Type	Description	Values
{field} value	string	The value extracted by the AI model	
{field} confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted value is accurate
{table} {column} value	string	The value extracted by the AI model for a cell in a table	
{table} {column} confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted cell value is accurate

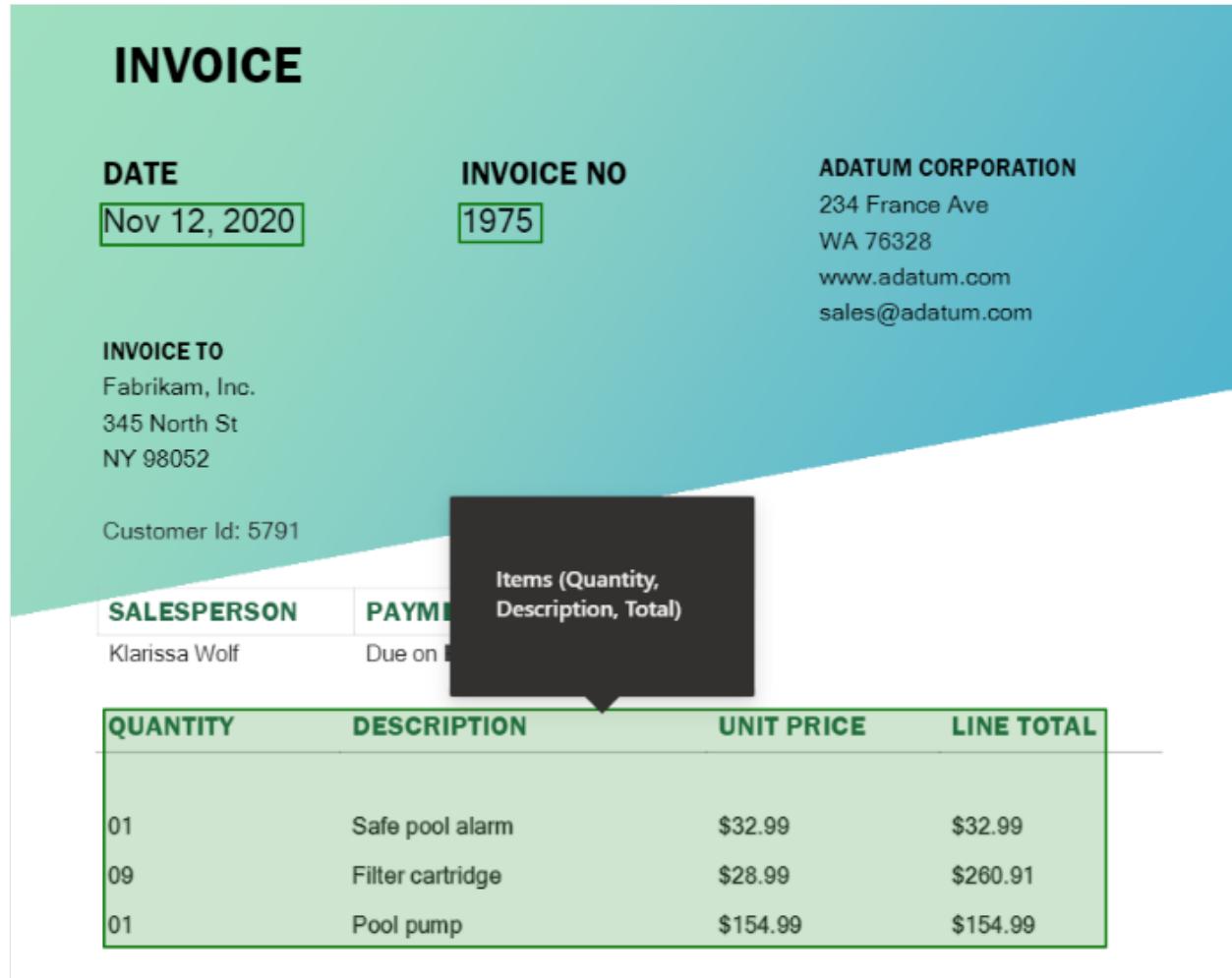
ⓘ Note

- More output parameters may be proposed such as field coordinates, polygons, bounding boxes and page numbers. These aren't listed on purpose as they're mainly intended for advanced use.
- Coordinates are represented as percentages of the document's height and width, originating from the top-left corner. For instance, if coordinates X = 0.10 and Y = 0.20 are given, this signifies a location at 10% of the document's width along the X-axis and 20% of its height along the Y-axis, both measured from the top-left corner.

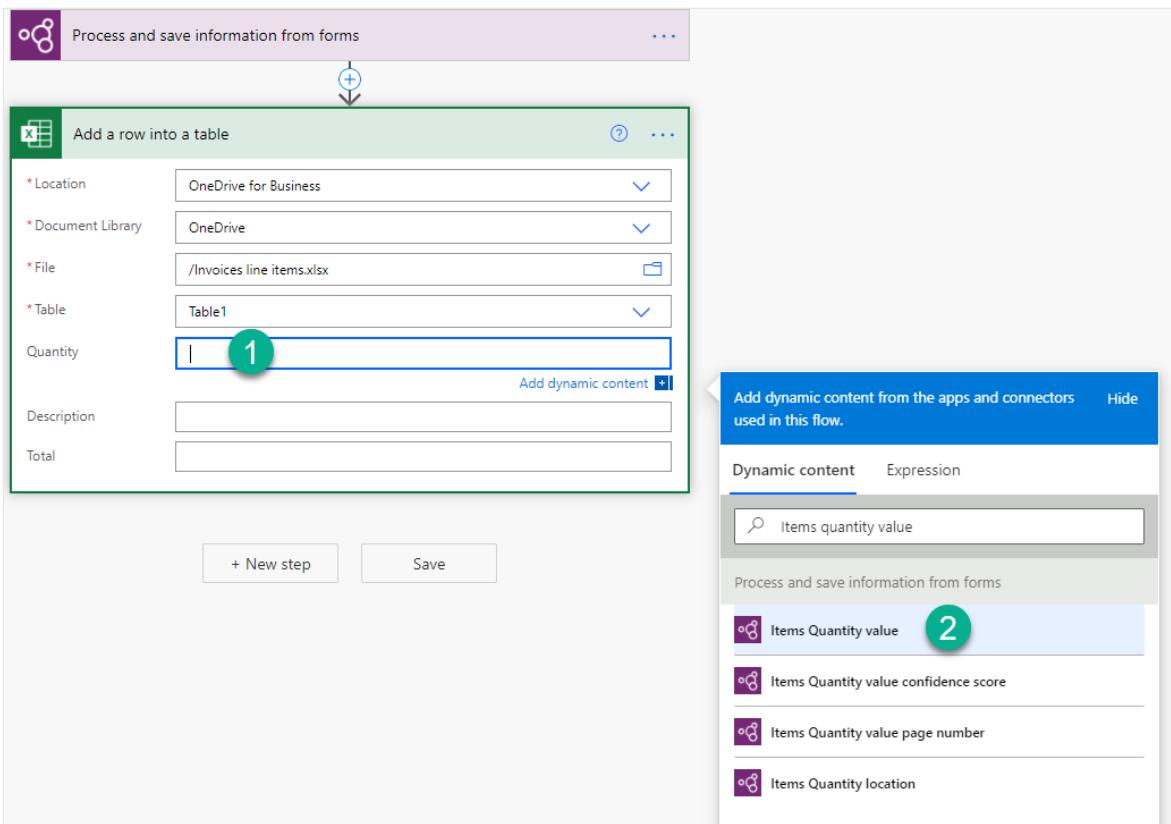
Common use cases

Iterate a document processing table output in Power Automate

To illustrate this procedure, we use the following example where we've trained a document processing model to extract a table that we've named **Items** with three columns: **Quantity**, **Description** and **Total**. We wish to store each line item from the table into an Excel file.



1. Select the field you wish to write the cell for a table. The dynamic content panel will open showing everything that the document processing model knows how to extract. Search for {your table name} {your column name} value. Our example uses *Items Quantity value*.



2. Once you add this value, the action where you added it is automatically inserted into an **Apply to each** control. This way, every row in the table will be processed when the flow is run.
3. Keep adding columns you want to iterate.

The screenshot shows a Power Automate flow. It begins with a step titled "Process and save information from forms". An arrow points down to an "Apply to each" loop. Inside the loop, there is a step titled "Add a row into a table". The "Add a row into a table" step has the following configuration:

Setting	Value
* Location	OneDrive for Business
* Document Library	OneDrive
* File	/Invoices line items.xlsx
* Table	Table1
Quantity	Items Quantity value
Description	Items Description value
Total	Items Total value

Below the "Add a row into a table" step is a button labeled "Add an action".

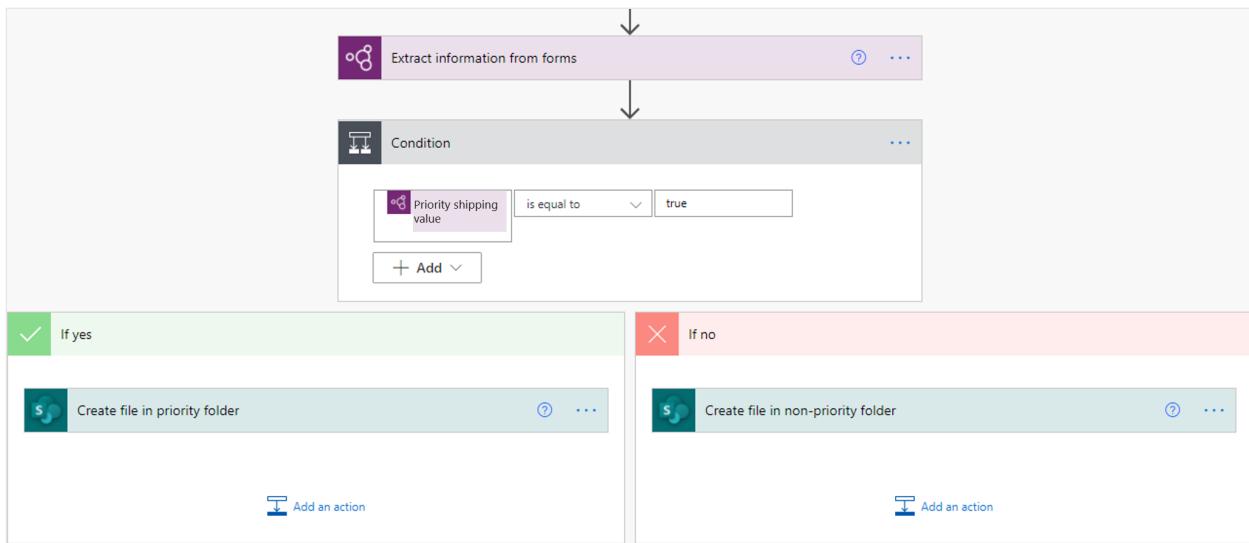
ⓘ Note

Tables extracted by document processing currently don't return a confidence score.

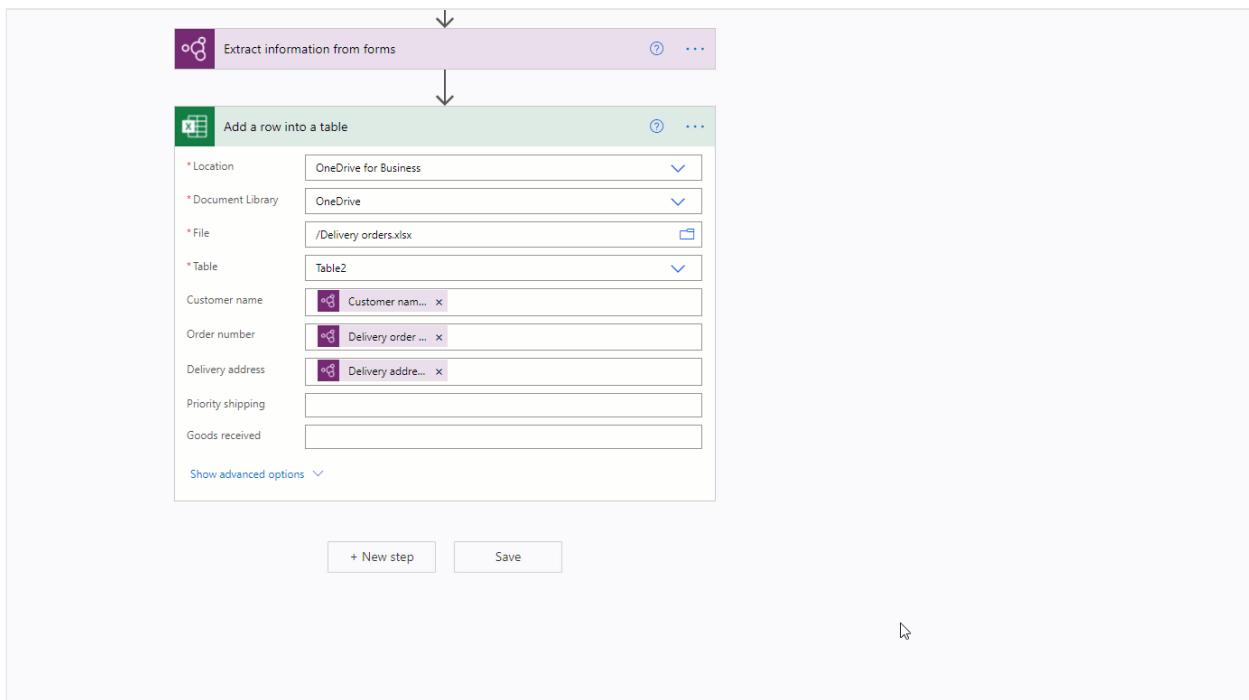
Process outputs of checkboxes in Power Automate

Checkbox values are of type Boolean: `true` means the checkbox is marked as selected in the document, and `false` means it's not.

One way you can check its value is with a **Condition** action. If the checkbox value is equal to `true`, then execute one action. If the value is `false`, execute a different action. The following illustration shows an example.



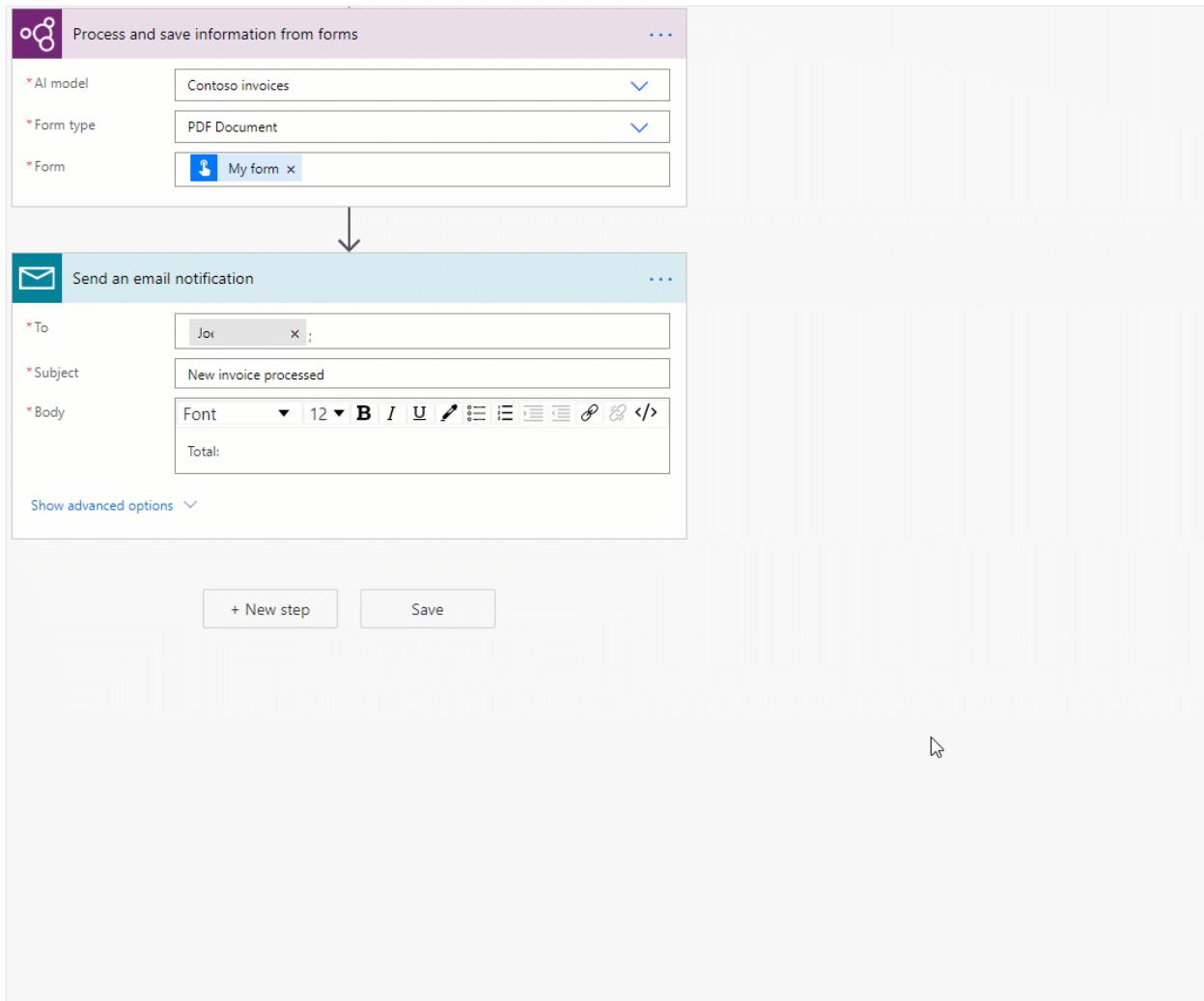
Another option is to map the `true/false` output of the checkbox to other values of your choice by using the `if` expression. For example, you might have a column in an Excel file where you want to write 'Priority' if one of the checkboxes in the document is selected, or 'Non-priority' if not selected. To do this, you can use the following expression: `if(<document processing output>, 'Priority', 'Non-priority')`. The following animation shows an example.



Remove currency symbols (€, \$,...) in a document processing output in Power Automate

To illustrate, the *Total* value extracted by the document processing model might have a currency symbol, for example, \$54. To remove the \$ sign, or any other symbols you want to omit, use the `replace` expression to remove it. Here's how:

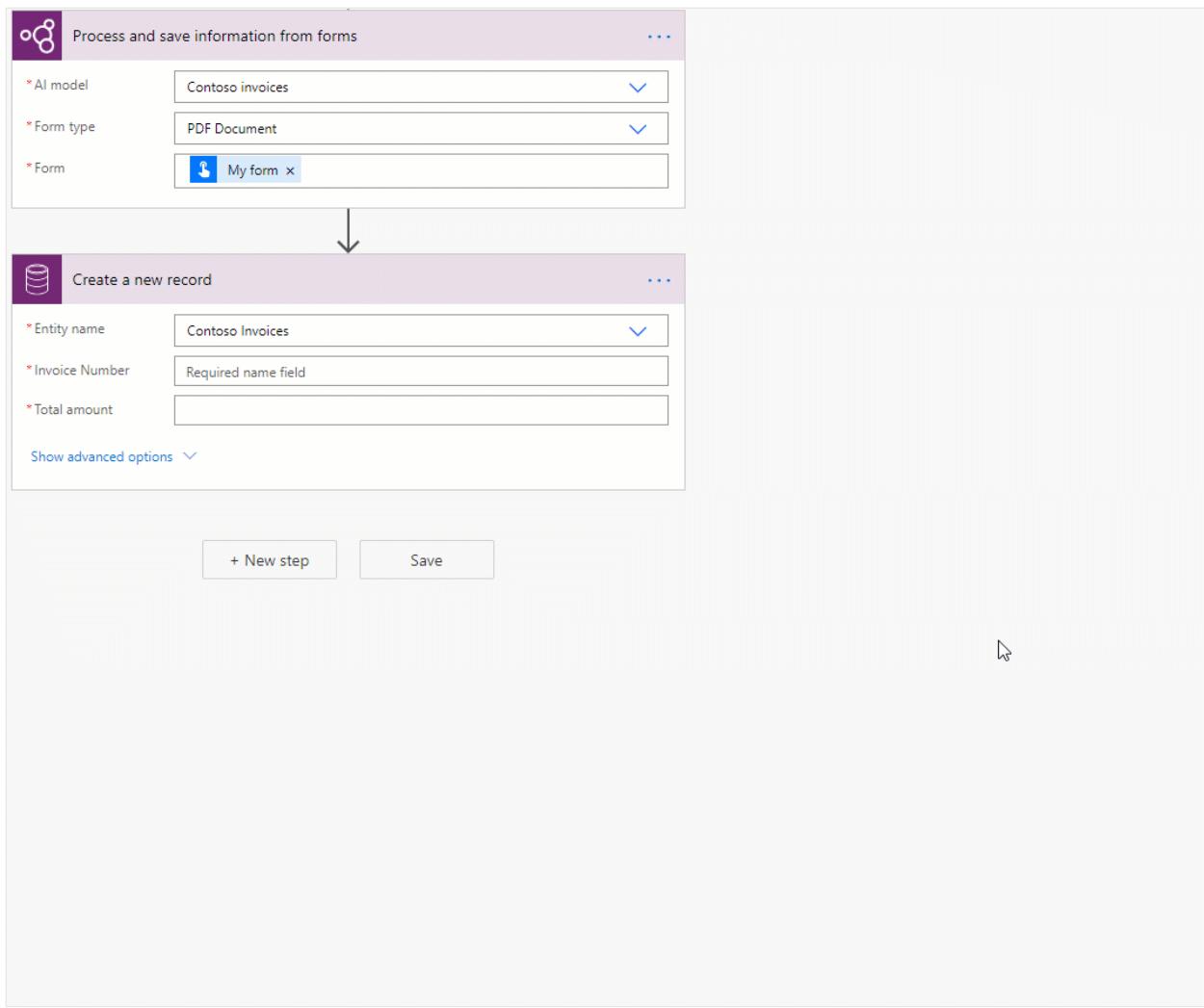
```
replace(<document processing output>, '$', '')
```



Convert a document processing output string to a number in Power Automate

AI Builder document processing returns all extracted values as strings. If the destination where you want to save a value extracted by AI Builder document processing requires a number, you can convert a value to number using the [int](#) or [float](#) expression. Use int if the number has no decimals. Use float if the number does have decimals. Here's how to do it:

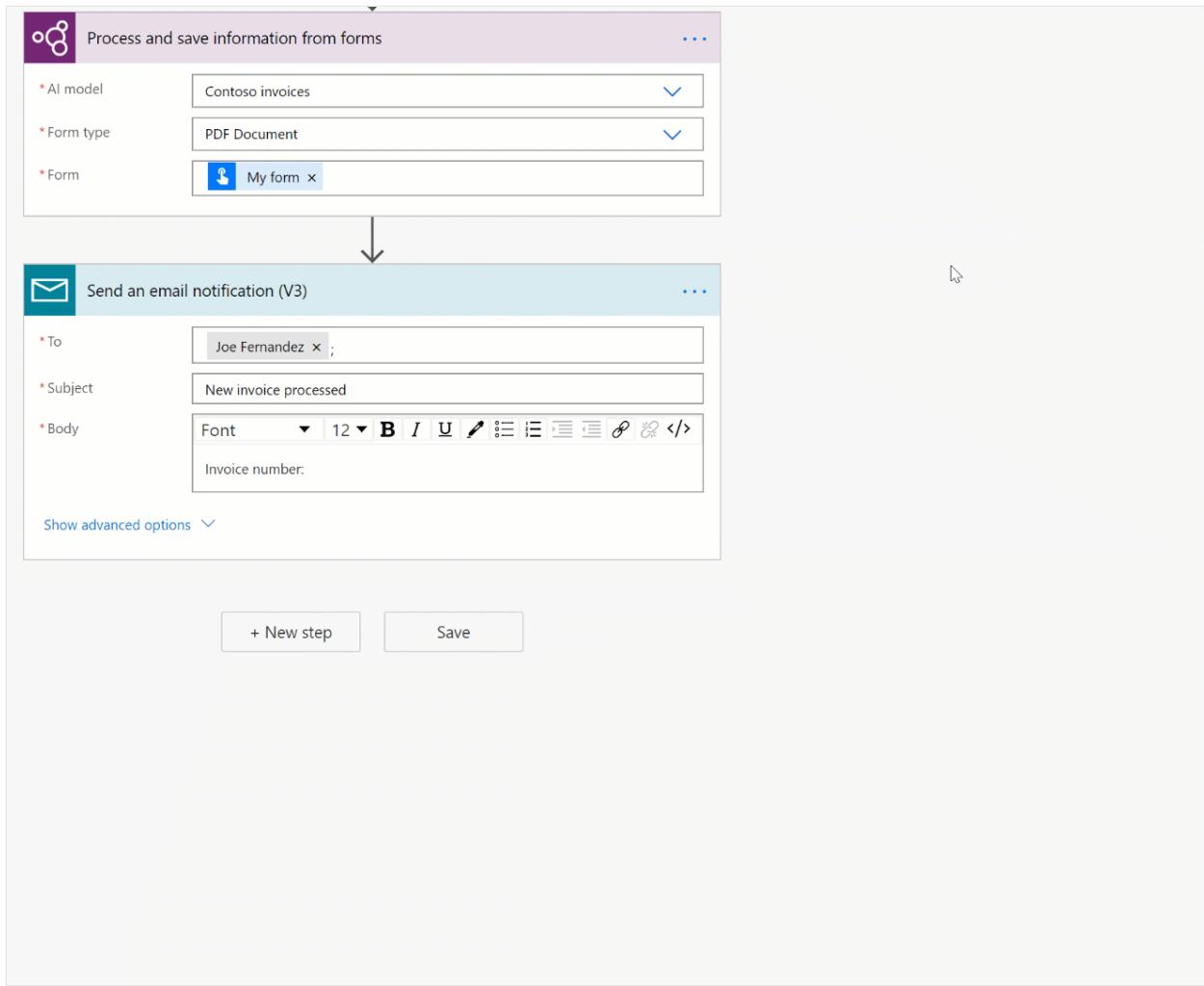
```
float('<document processing output>')
```



Remove blank spaces in a document processing output in Power Automate

To remove blank spaces from output values, use the [replace](#) function:

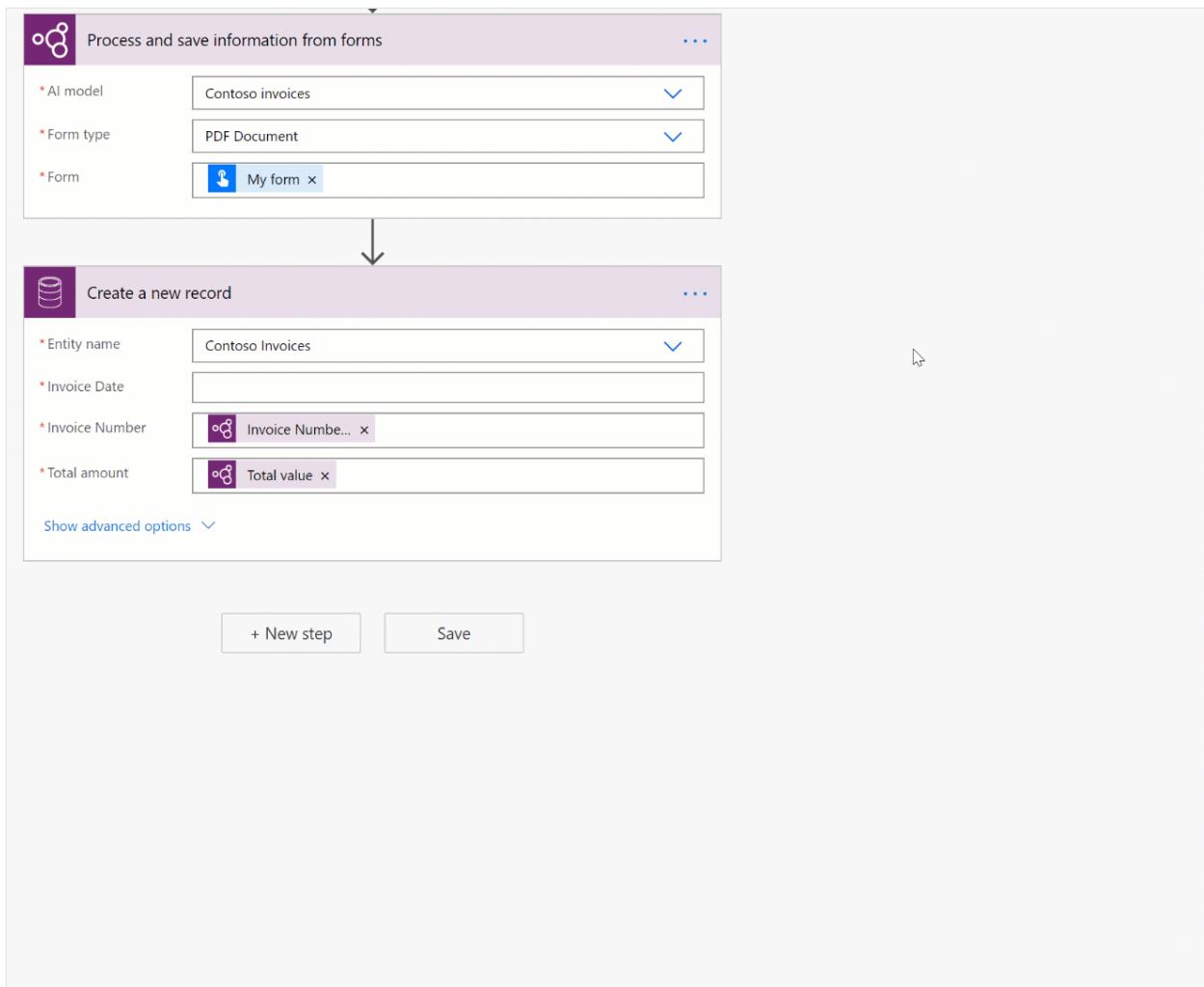
```
replace(<document processing output>, ' ', '')
```



Convert a document processing output string to a date in Power Automate

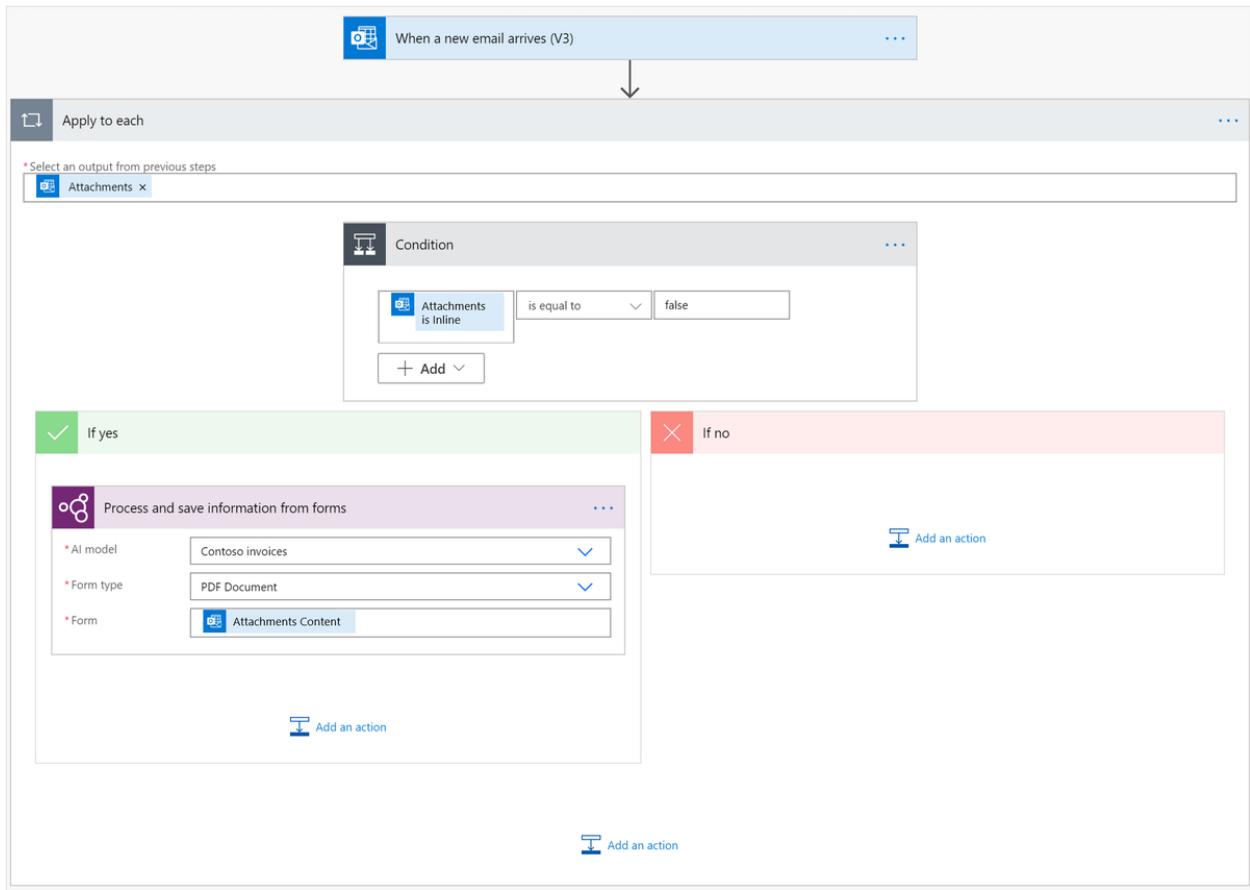
AI Builder document processing returns all outputs as strings. If the destination where you want to save a value extracted by document processing is required to be in date format, you can convert a value that contains a date into date format. Do this by using the [formatDateTime](#) expression. Here's how to do it:

```
formatDateTime(<document processing output>)
```



Filter email signature from a flow so that it's not processed by the document processing model (Microsoft 365 Outlook)

For incoming emails from the Microsoft 365 Outlook connector, email signatures are picked up by Power Automate as attachments. To keep these from being processed by the document processing model, add a condition to your flow that checks if the output from the Microsoft 365 Outlook connector named **Attachments is Inline** is equal to false. In the **If yes** branch of the condition, add the document processing action. With this, only email attachments that aren't inline signatures will be processed.



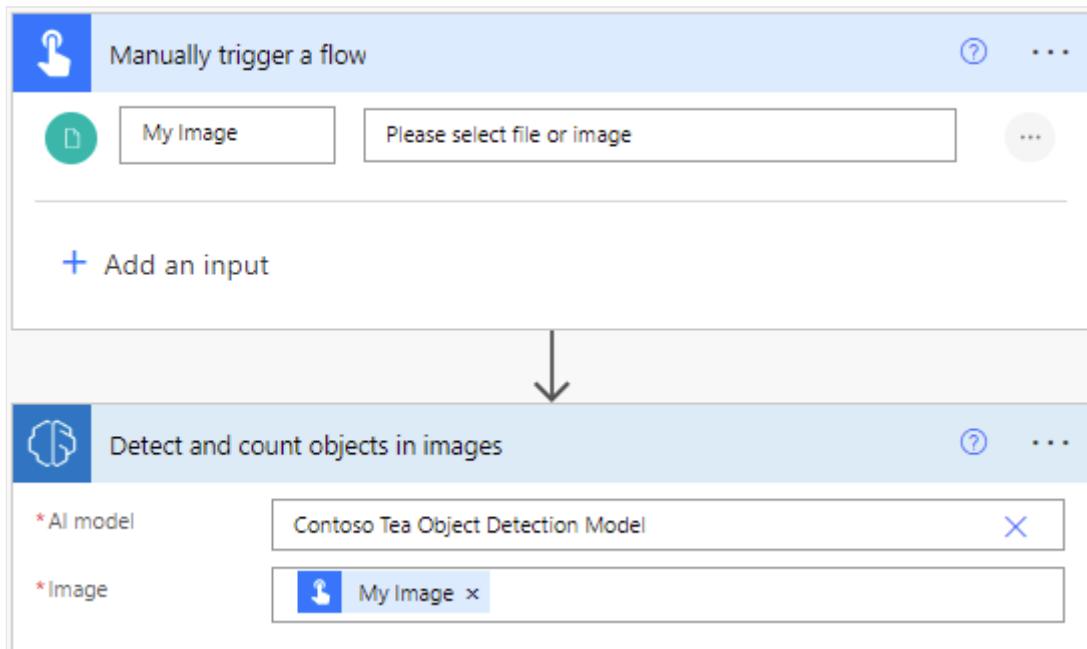
See also

- [Overview of the document processing model](#)
- [Training: Process custom documents with AI Builder \(module\)](#)

Use the object detection model in Power Automate

Article • 01/05/2023

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > File** as the input type.
5. Replace **File Content** with **My image** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Detect and count objects in images** in the list of actions.
7. Select the object detection model you want to use.
8. In the **Image** input, select **My Image** from the **Dynamic content** list:



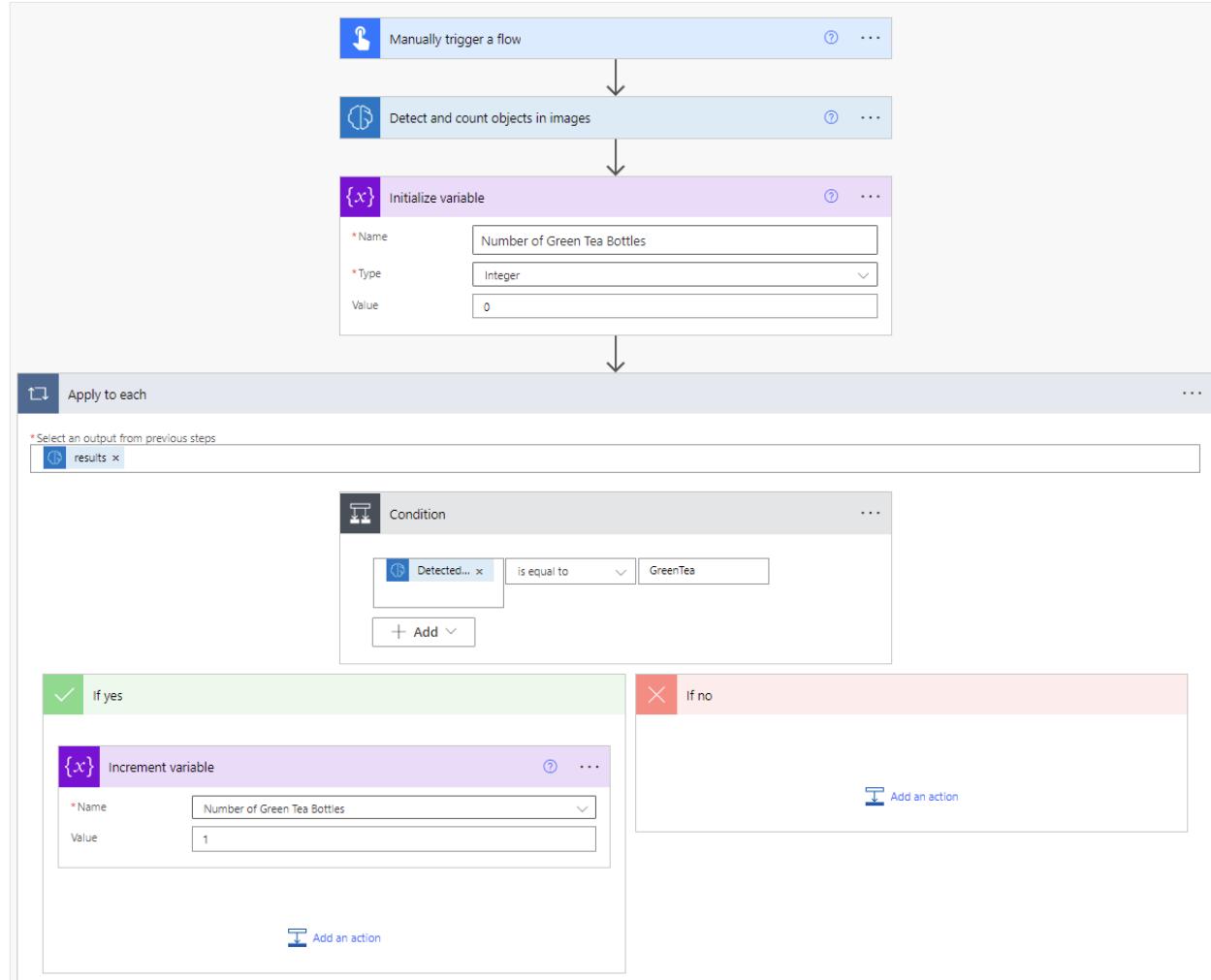
9. To retrieve the name of the detected object or objects on the image:
 - a. Select **New step**.
 - b. Search for the successive action you want your flow to perform, for example add a row into an Excel table or send an email.

- c. Select any of the successive actions' inputs, and then select **Detected object name** in the **Dynamic content** list.

Congratulations! You've created a flow that uses an object detection AI Builder model. Select **Save** on the top right, and then select **Test** to try out your flow.

Example object detection flow

The following example shows the creation of a flow that is triggered by an image. This flow counts the number of green tea bottles in the image.



To learn more about the triggers and actions you can use, see the [Power Automate documentation](#).

Parameters

Input

Name	Required	Type	Description	Values

Name	Required	Type	Description	Values
AI Model	Yes	model	Object detection model to use for analysis	Trained and published object detection models
Image	Yes	file	Image to process	

Output

Name	Type	Description	Values
Detected object name	string	The detected object name	Among the tags defined at model creation
Detected object ID	string	The detected object ID	
Confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted value is accurate
Coordinates height	float	Coordinates left of the object	
Coordinates left	float	Coordinates left of the object	
Coordinates top	float	Coordinates top of the object	
Coordinates width	float	Coordinates width of the object	

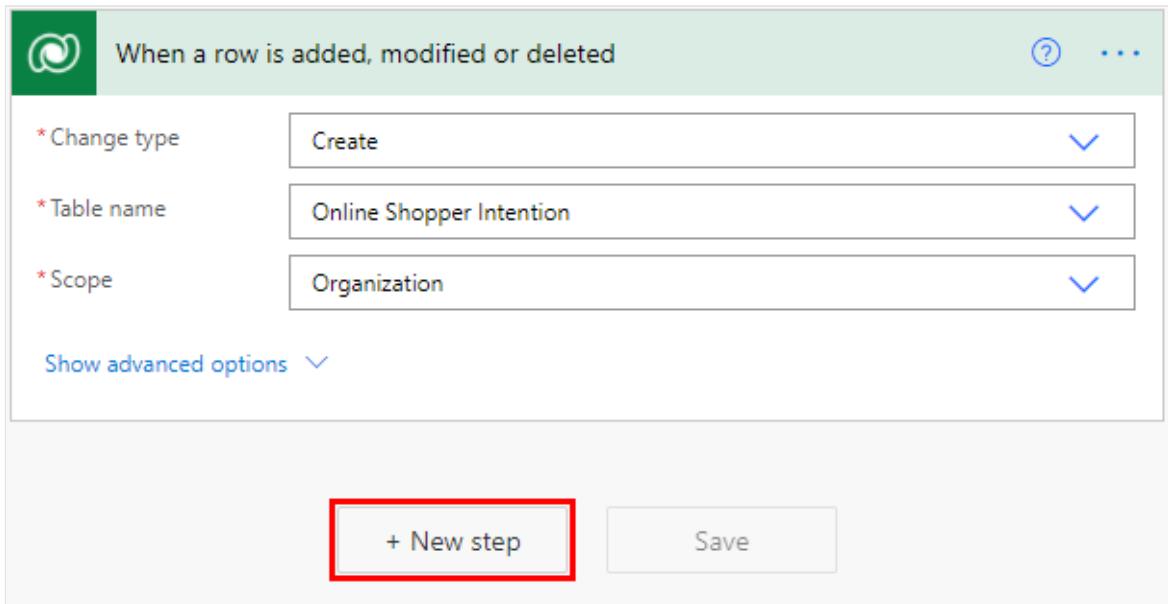
See also

- [Overview of the object detection model](#)
- [Training: Detect objects with AI Builder \(module\)](#)

Use a prediction model in Power Automate

Article • 03/05/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane.
3. Select **New flow** from the menu at the top, and then select a flow in the **Build your own from blank** section. (The example uses **Automated cloud flow**.)
4. Name your flow, and then select **Skip**. Instead of using this screen, you'll select your flow's trigger in the next step by filtering a list.
5. Find your trigger by typing it in the search bar and then selecting it in the **Triggers** list. (The example uses the **When a row is added, modified, or deleted** trigger.)
6. Configure the trigger.
7. If you want to add any steps to prepare your data, select **Next step**.



8. Select **Next step**, enter **predict** in the search bar, and select **Predict** in the **Actions** list.
9. Select your model in the input, and then complete the fields using your data or data from previous steps in the flow.
10. Update the row by using prediction output.

Congratulations! You've created a flow that uses the real-time prediction feature in AI Builder. Select **Save** on the top right, and then select **Test** to try out your flow.

See also

[Use your prediction model](#)

[Train your model in AI Builder](#)

[Publish your model in AI Builder](#)

Use your Azure OpenAI Service model in Power Automate (preview)

Article • 06/12/2023

[This topic is pre-release documentation and is subject to change.]

GPT (Generative Pre-trained Transformer) models are a type of natural language processing model. GPT models are trained on a large body of content to generate human-like text from a prompt. When you combine them with workflow automation, you can use AI models like GPT to automate a variety of tasks. For example, you can build workflows to automatically generate drafts of emails, customer service responses, and product descriptions. You can also use them to generate scripts that allow customer service agents to respond quickly to customer inquiries.

Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- [View our preview terms ↗](#).
- This capability might not be available in your region yet.
- This capability may be subject to usage limits or capacity throttling.

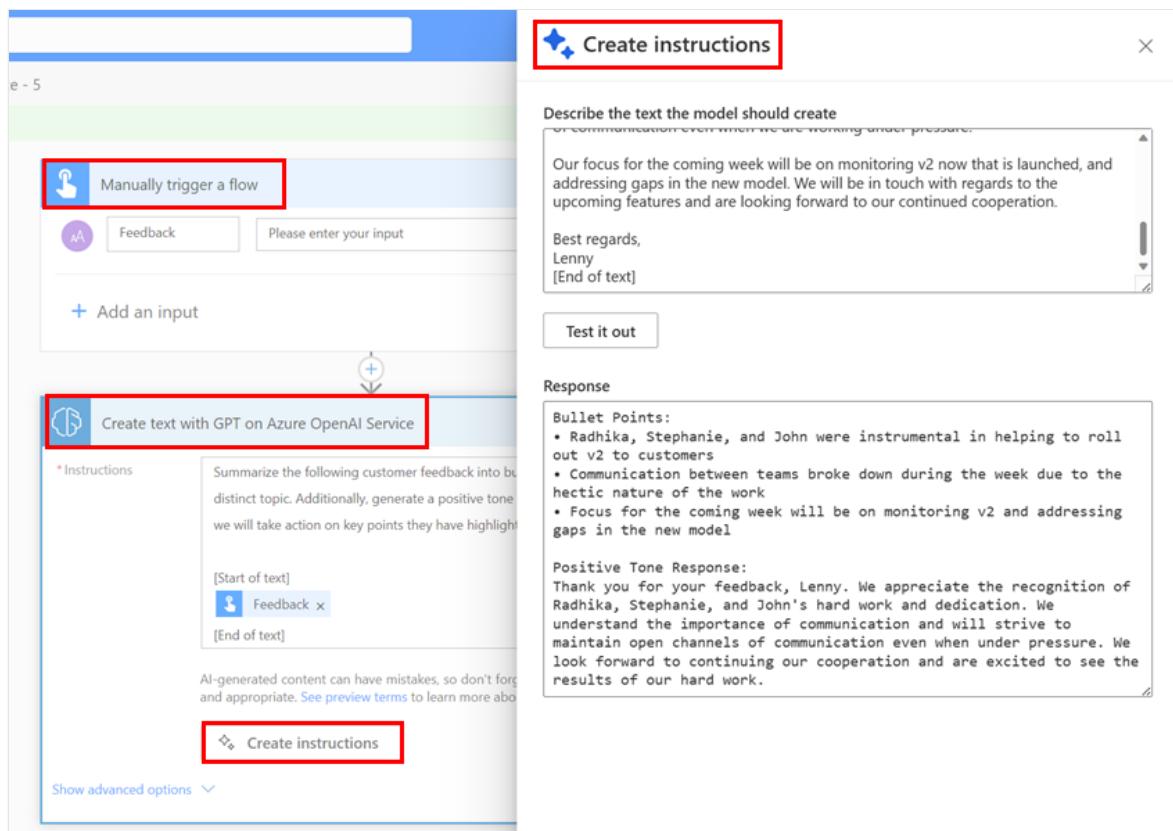
Create instructions

GPT model prompts have two parts, the *instruction* and the *context*. The instruction tells the model what it should do. The context is the information the model needs to follow the instruction. In an automation task, the instruction is constant and the context is provided by dynamic content.

The following example uses an instant cloud flow, but you can include a GPT model in an automated cloud flow, too.

1. Sign in to [Power Automate ↗](#).

2. On the left pane, select **My flows**.
3. Select **New flow > Instant cloud flow**, and then name your flow.
4. Under **Choose how to trigger this flow**, select **Manually trigger a flow**, and then select **Create**.
5. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
6. Select **+ New step > AI Builder**, and then select **Create text with GPT on Azure OpenAI Service** in the list of actions.
7. Select **Create instructions** and enter instructions and a sample context. Refine the prompt based on the responses until you're satisfied the model is working as intended.



8. Replace the sample context with dynamic content.

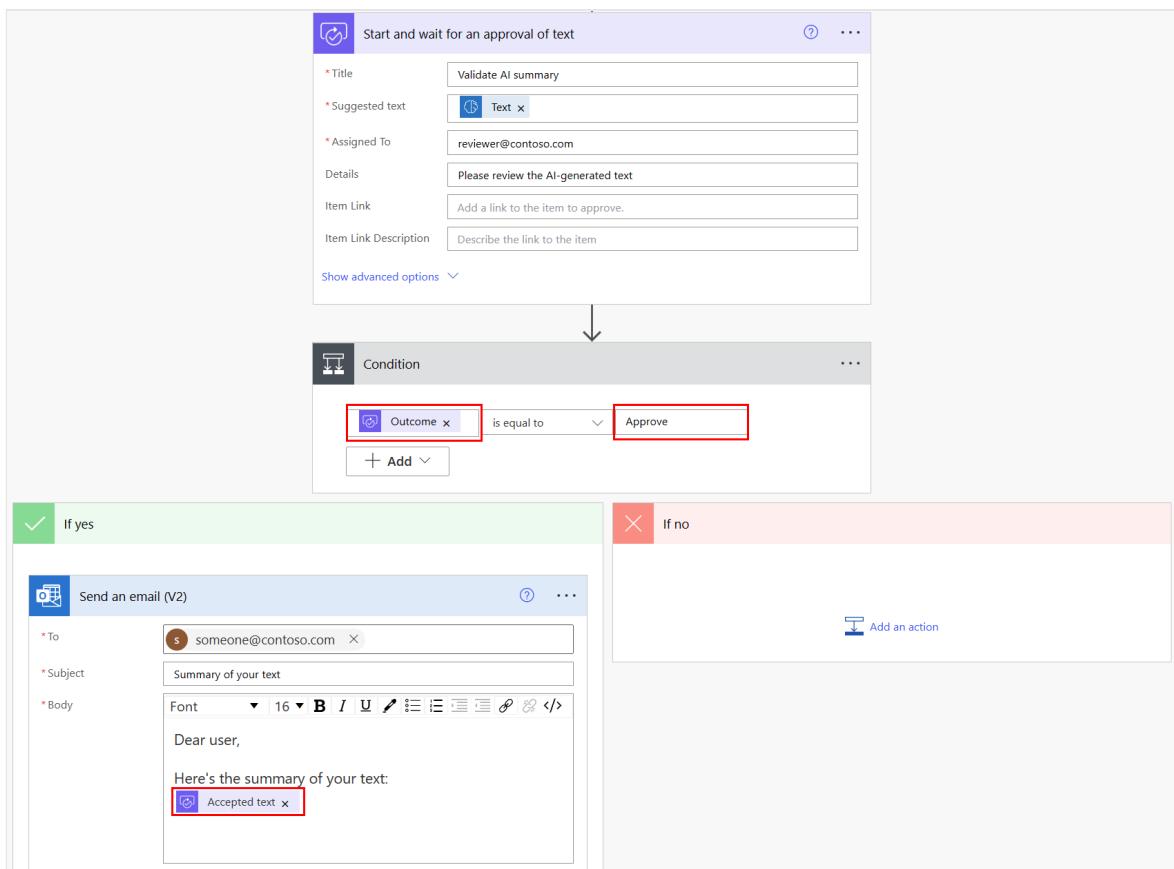
In this example, the dynamic content is the topic variable from the previous step. The dynamic content can be anything the model needs to generate a new response every time; for example, an email to generate a response or text from a document to summarize.

Insert human oversight

AI-generated content can be factually incorrect, inappropriate, or biased. We strongly recommend that you institute a practice of inserting [human oversight](#) in workflows that use AI-generated text before it's posted or used anywhere.

In the following example, you send the AI-generated summary of a text by email after a human reviews it.

1. Select + New step > Approvals.
2. In the list of actions, select **Start and wait for an approval of text**.
3. In the list of actions, select + New step > Condition.
4. In the **Condition** box, set **Outcome** from the approval step as the condition to check and **Approve** as the positive response to validate.
5. If the condition is true, you can proceed with the email sending. In the body, make sure to select **Accepted text** from the approval step, which is the AI-generated text reviewed by a human.



6. Select **Save**, and then select **Test** to try out your flow.

The human receiving the AI-generated text to review has the possibility to accept, edit, or reject the text.



Approvals | Power Automate

Validate AI summary

Requested by [Antonio Rodriguez](#) <antonio@contoso.com>

Date Created Thursday, June 8, 2023 4:14 PM

Please review the AI-generated text

Text

You can modify the text below before accepting.

This is a summary of a long text.

[Approve](#)

[Reject](#)

Get the Power Automate app to receive push notifications and grant approvals from anywhere. [Learn more](#). This message was created by a flow in Power Automate. Do not reply. Microsoft Corporation 2020.

Input parameters

Name	Required	Type	Description	Values
Prompt/instructions	Yes	String	The instruction or the prompt for the model to act on	Natural language instruction for the model along with the dynamic content that the model can act on
Parameters	Optional	JSON	Model parameters to optimize the output	{ "temperature": 0, "max_tokens": 750, "top_p": 1, "frequency_penalty": 0, "presence_penalty": 0 }

Output parameters

Name	Type	Description	Values
Text	String	Generated text	The response that the model has generated based on the input instructions

Name	Type	Description	Values
Finish reason	String	Finish reason returned by AI model	TBD

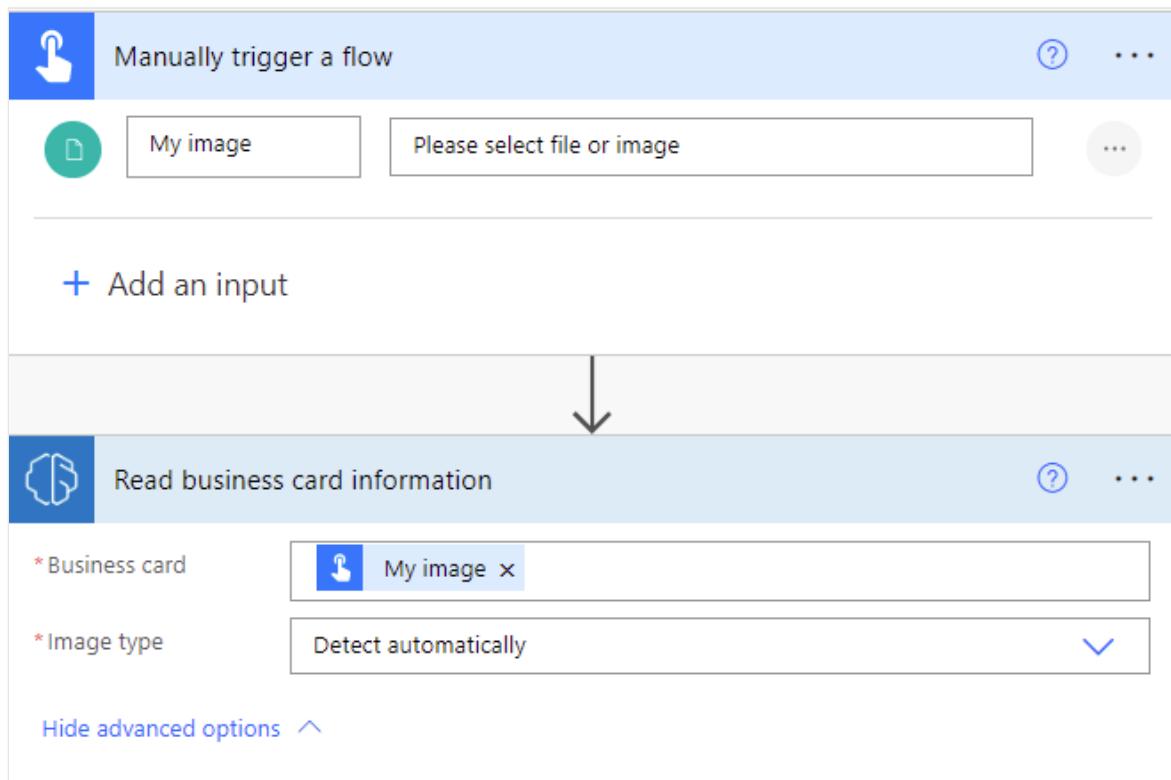
See also

- [Azure OpenAI Service model overview \(preview\)](#)
- [How text generation in Azure OpenAI Service works \(preview\)](#)
- [Use your Azure OpenAI Service model in Power Apps \(preview\)](#)
- [Video: How to automate extracting text from emails in just 1 minute ↗](#)

Use the business card reader prebuilt model in Power Automate

Article • 02/11/2023

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > File** as the input type.
5. Replace **File Content** with **My image** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Read business card information** in the list of actions.
7. Specify **My Image** from the trigger in the **Business card** input for your flow.
8. Select **Show advanced options** and verify that **Detect automatically** is in the **Image type** input.

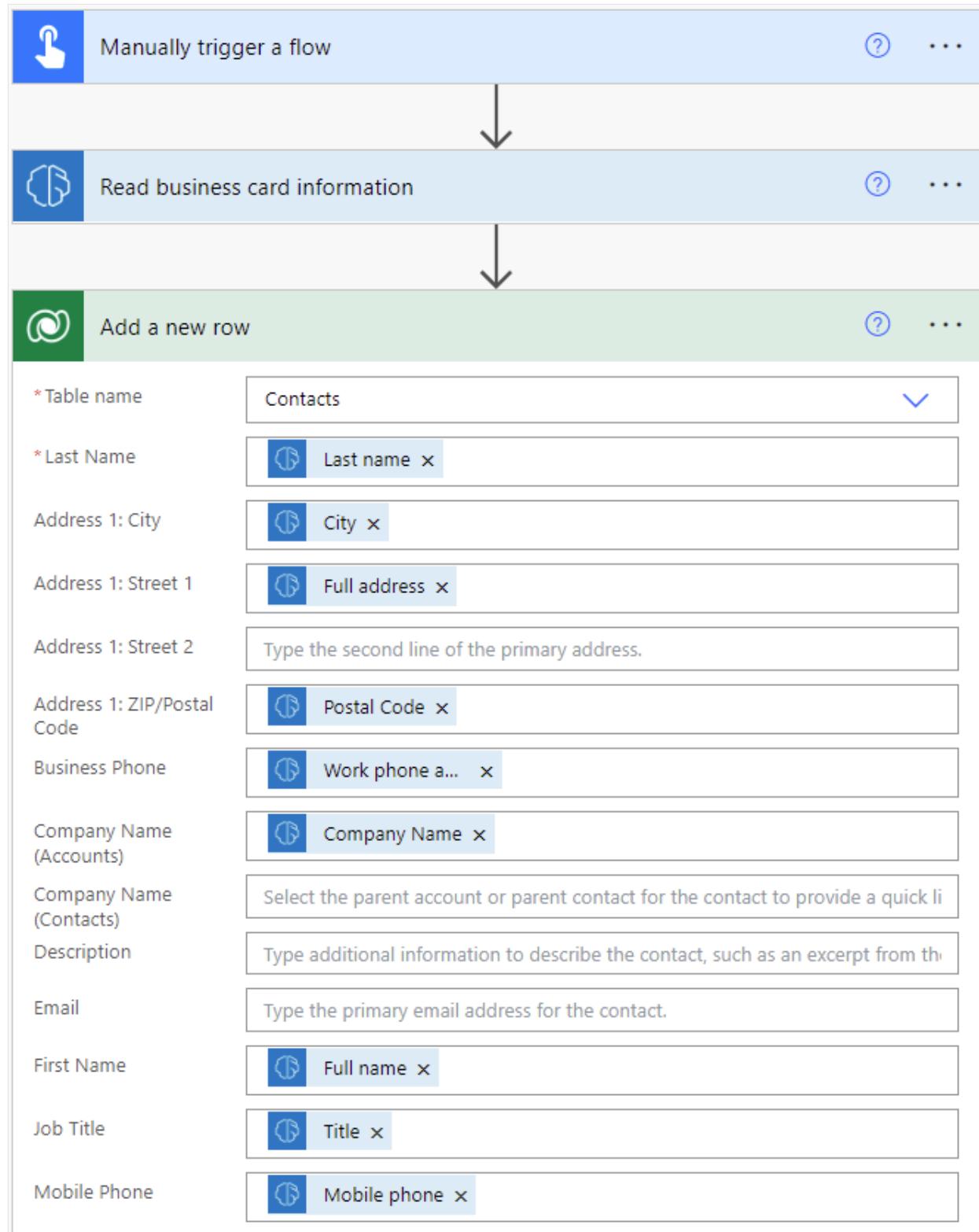


Congratulations! You've created a flow that uses the business card reader AI model. Select **Save**, and then select **Test** in the upper-right corner to try out your flow.

Example business card reader flow

The following example shows a new contact being created in Microsoft Dataverse using the business card data.

To add the **Add a new row** step, select **+ New step > Microsoft Dataverse > Add a new row**.



Parameters

Input

Name	Required	Type	Description	Values
Image type	Yes	string	Mime type of the image	"auto" as default value. This column being obsolete, any value will be accepted.
Image	Yes	file	Image file to analyze	

Output

Name	Type	Description
City	string	The city address
Country	string	The country address
Postal Code	string	The postal code address
PO Box	string	The post office box address
State	string	The state address
Street	string	The street address
Work phone or other phone	string	The first phone or fax number
Company name	string	The company name
Department	string	The organization department found
Email	string	The contact email found in the business card, if any
Fax	string	The third phone or fax number
First name	string	The contact first name
Full address	string	The contact full address
Full name	string	The contact full name
Title	string	The contact job title
Last name	string	The contact last name
Mobile phone	string	The second phone or fax number

Name	Type	Description
Website	string	The website

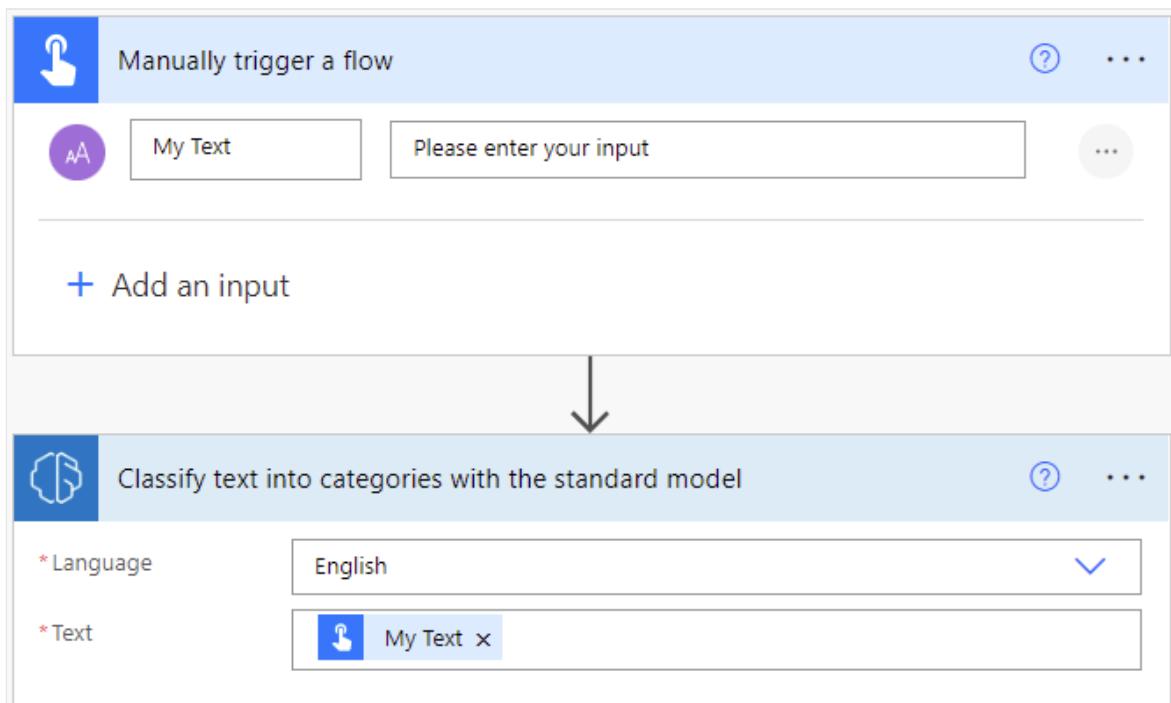
See also

[Business card reader overview](#)

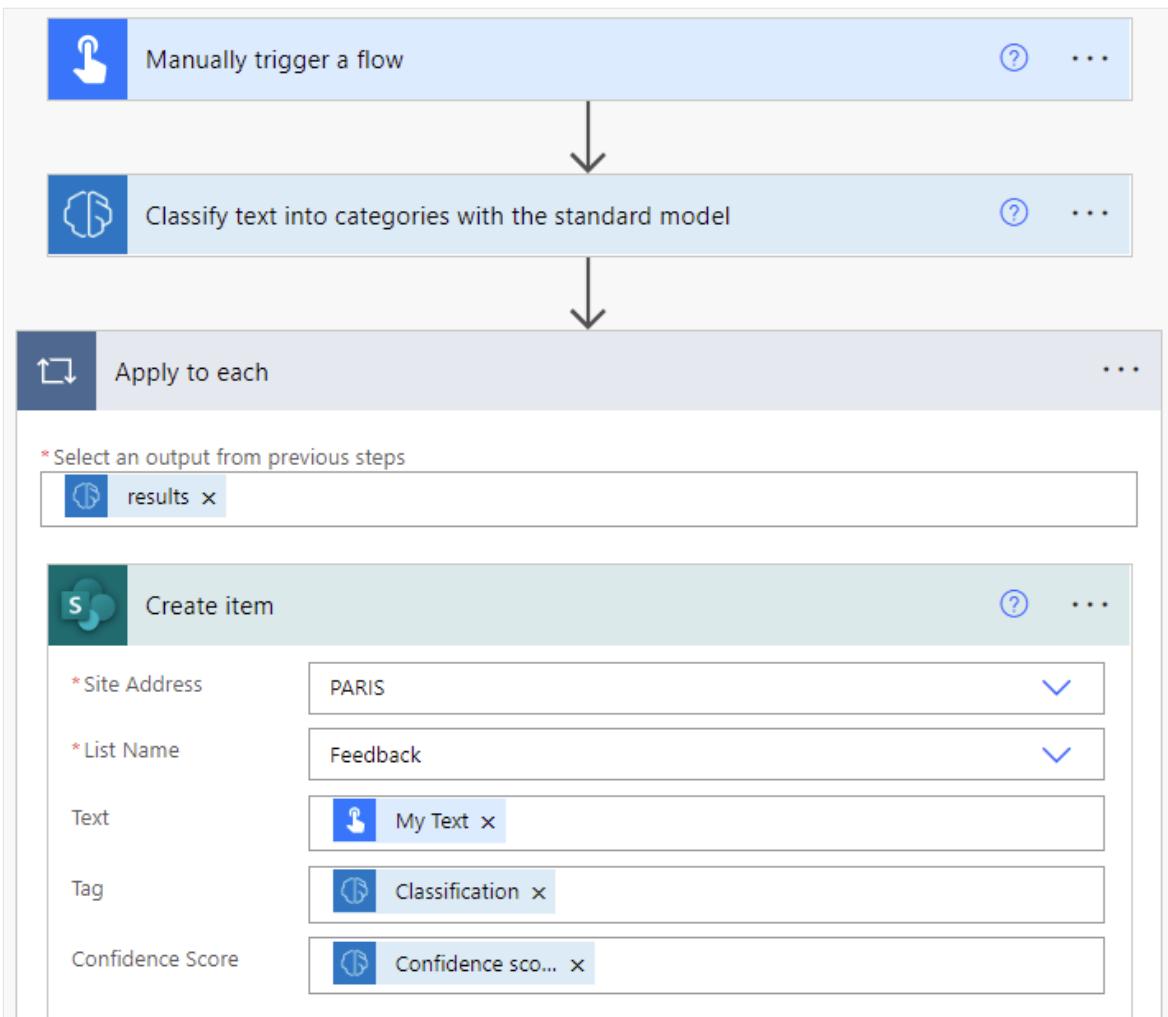
Use the category classification prebuilt model in Power Automate

Article • 01/05/2023

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
5. Replace **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Classify text into categories with the standard model** in the list of actions.
7. Select the language in the **Language** input and specify the **My Text** column from the trigger in the **Text** input.



8. In the successive actions, you can use any columns and tables extracted by the AI Builder model. The following example, saves each inferred **Classification** and **Confidence score** into a SharePoint list.



Congratulations! You've created a flow that uses an AI Builder prebuilt category classification model. Select **Save** on the top right and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description	Values
Text	Yes	string	Text to analyze	Text sentences
Language	Yes	string	Language of the text to analyze	List of predefined languages or language code (ex.: "en", "fr", "zh_chs", "ru")

Congratulations! You've created a flow that uses a prebuilt category classification AI Builder model. Select **Save** on the top right, and then select **Test** to try out your flow.

Output

Name	Type	Description	Values
------	------	-------------	--------

Name	Type	Description	Values
Classification	string	Entity identified	Issues, compliment, customer service, documentation, price & billing, staff
Confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted value is accurate

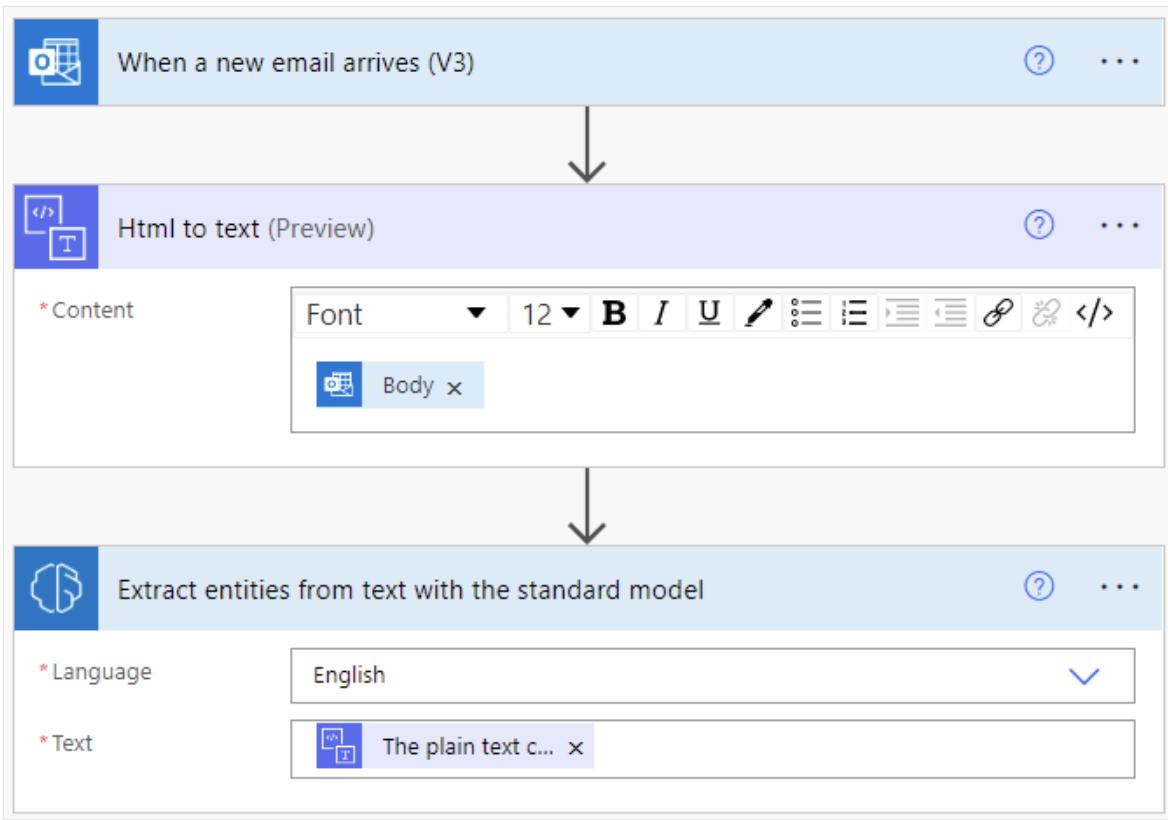
See also

- [Training: Get started with AI Builder category classification \(module\)](#)
- [Category classification prebuilt model](#)

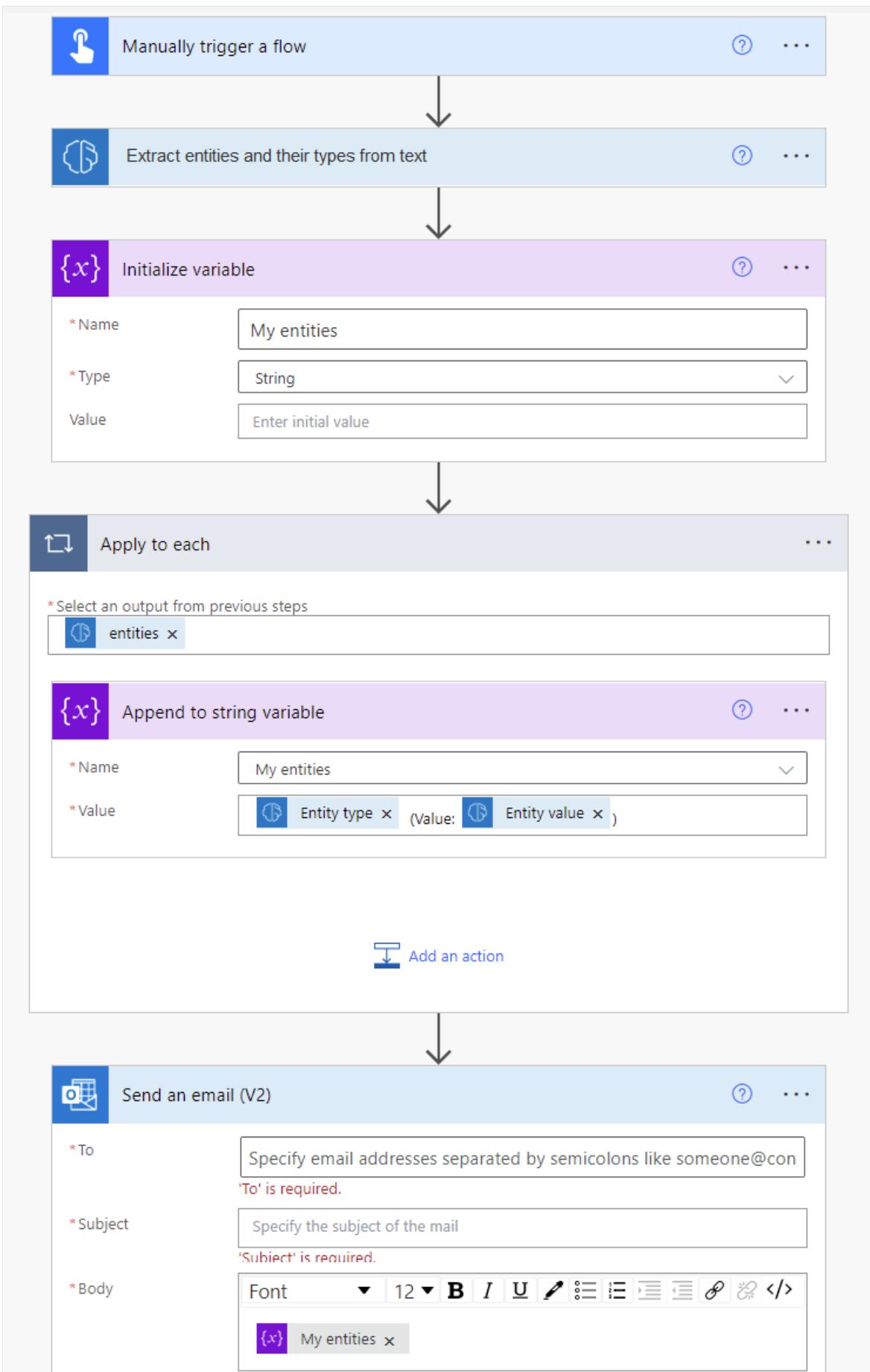
Use the entity extraction prebuilt model in Power Automate

Article • 03/05/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Automated cloud flow**.
3. Name your flow, select **When a new email arrives V3 (Office 365 Outlook)** under **Choose your flow's trigger**, and then select **Create**.
4. Select **+ New step**.
5. In the **Search connectors and actions** input, enter *html to text* to search for and select **Html to text (preview)** in the list of actions.
6. Select the text below the ribbon and select **Body** from the **Dynamic Content** list. This will convert the body of your document to plain text.
7. Select **+ New step > AI Builder > Extract entities from text with the standard model** in the **Actions** list. (If you want to use your own model instead, select **Extract entities from text with one of your custom models**.)
8. In the **Language** input, select or enter your language.
9. In the **Text** input, select **The plain text content** from the **Dynamic content** list.



10. In the successive actions, you can use any columns extracted by the AI Builder model. For example, you can send an email using the **Entity type** and **Entity value** columns.



Congratulations! You've created a flow that uses an entity extraction model. Select **Save** on the top right, and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description	Values
Text	Yes	string	Text to analyze	Text sentences
Language	Yes	string	Language of the text to analyze	List of predefined languages or language code (ex.: "en", "fr", "zh_chs", "ru")

Output

Name	Type	Description	Values
Entity type	string	Type of the entity	Example: DateTime or Organization
Entity value	string	Content of the entity	Example: June 1 or Contoso
Confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the extracted value is accurate
Starting location	integer	Where the entity's first character appear in the line	
Character count	integer	How long the entity is	

See also

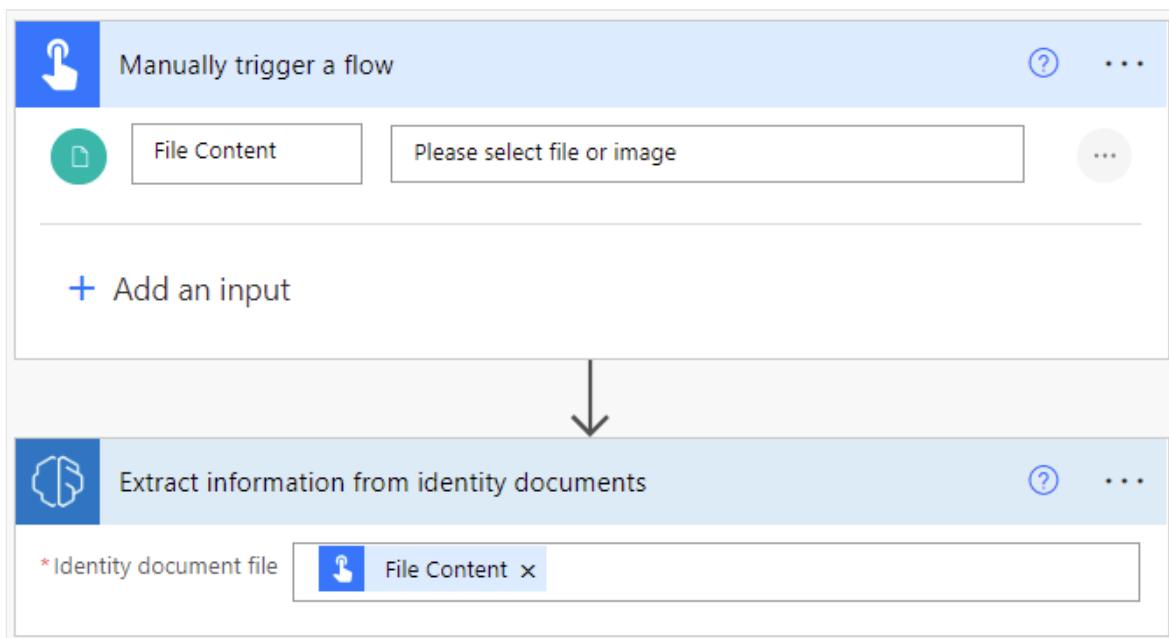
[Entity extraction prebuilt model](#)

[AI Builder in Power Automate overview](#)

Use the ID reader prebuilt model in Power Automate

Article • 09/14/2022

1. Sign in to [Power Automate](#).
2. In the left pane, select **My flows**, and then select **New flow > Instant cloud flow** in the menu at the top.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > File** as the input type.
5. Select **+New step > AI Builder > Extract information from identity documents**.
6. Specify **File Content** as the identity document file you want to process in your flow.



Congratulations! You've created a flow that uses the ID reader model. Select **Save**, and then select **Test** in the upper-right corner to try out your flow.

Example flow that adds extracted information to an Excel worksheet

In the following example, you'll add steps to your flow to enter the extracted information in an Excel worksheet. First, you'll prepare a table to use in your flow. The

table must match the information you want to extract. Then you'll add an Excel connector to your flow.

Create an Excel table

1. Create an Excel workbook in a Microsoft OneDrive or SharePoint folder.
2. In the first row of the worksheet, enter the following values, one to a column: **First name**, **Last name**, **Identity document number**, and **Country**. These values are the column headers for your table.
3. Select the cells and format them as a table, with the first row as the header.

	A	B	C	D
1	First name ▾	Last name ▾	Identity document number ▾	Country ▾
2				
3				

4. Save and close the workbook.

Enter the extracted data in the table

1. Use the ID reader flow you created, or create another one for this example.
2. Select **+New step > Excel Online (Business) > Add a row into a table**.
3. Select a **Location**, **Document Library**, and **File** to specify where to find your Excel workbook.
4. Select the **Table** that you created in the previous step.
5. In **First name**, **Last name**, and **Identity document number**, select the matching value in the dynamic content list.
6. In **Country**, select **Country/Region** in the dynamic content list.

Add a row into a table

* Location: OneDrive for Business

* Document Library: OneDrive

* File: /ID Documents/ID Documents.xlsx

* Table: Table1

First name: First name x

Last name: Last name x

Identity document number: Identity docum... x

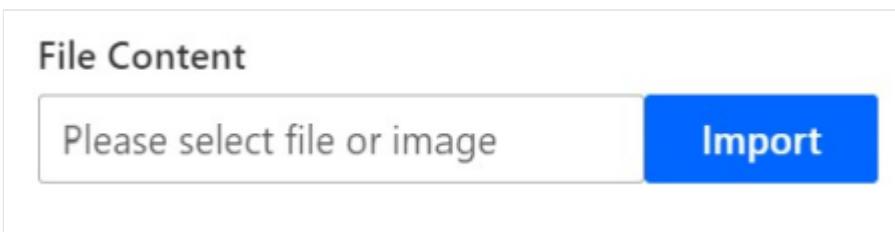
Country: Country/Region x

Show advanced options ▾

7. Select Save.

Test the flow

1. Select **Test**, select **Manually**, and then select **Test** to trigger the action.
2. In **File Content**, select an identity document file or image, and then select **Import**.



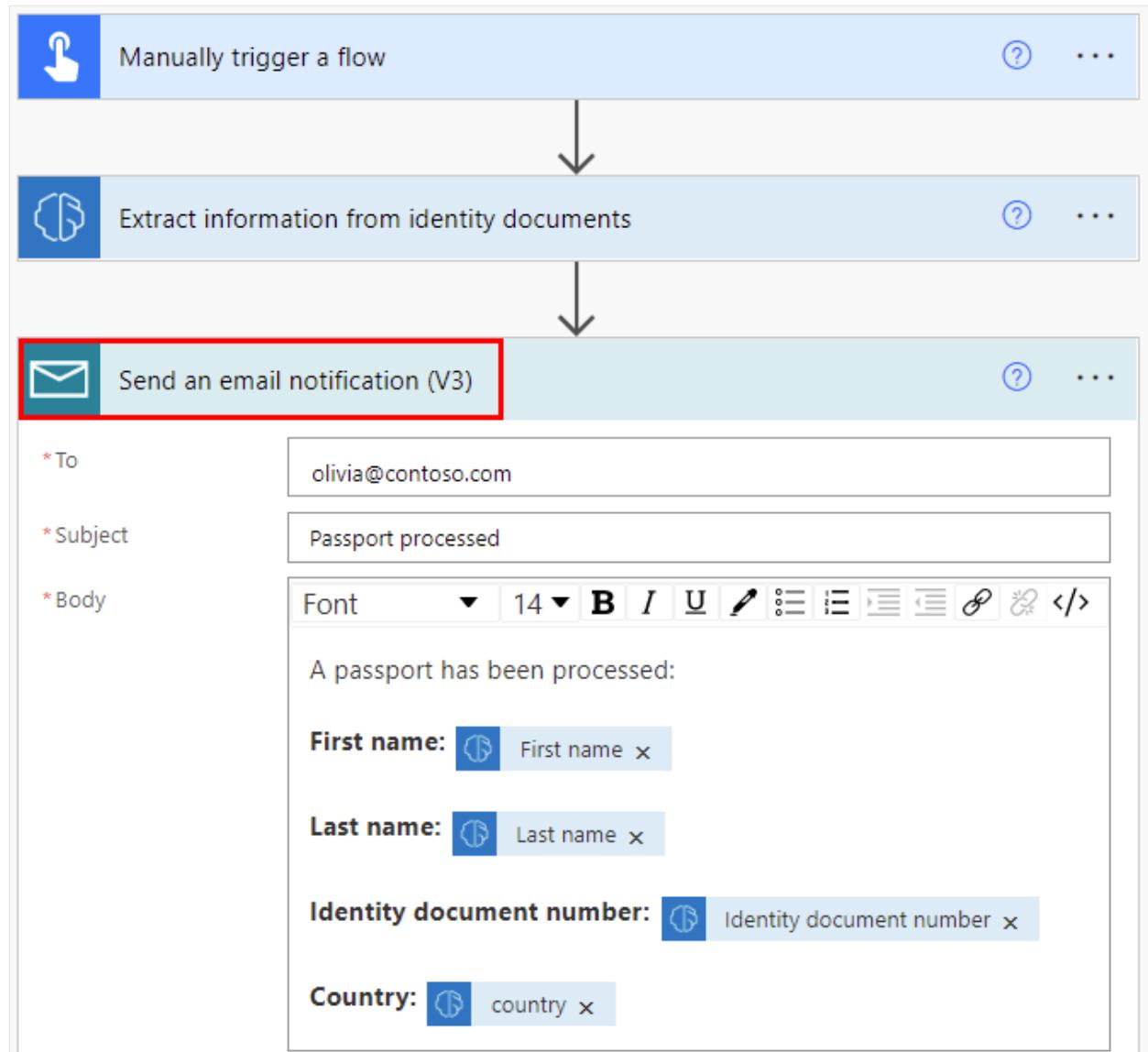
3. Select **Run Flow**.

The flow may take a few seconds to execute while AI Builder extracts the data and adds a new entry to the table in Excel. Open your Excel workbook to confirm the extracted information has been entered.

	A	B	C	D
1	First name	Last name	Identity document number	Country
2	JENNIFER	BROOKS	340020013	USA
3				

Example flow that sends extracted information in an email

The following example shows how to set up a flow to send the extracted information in an email. You can add the **Send an email notification** connector to the flow you created earlier or create an ID reader flow for this example.



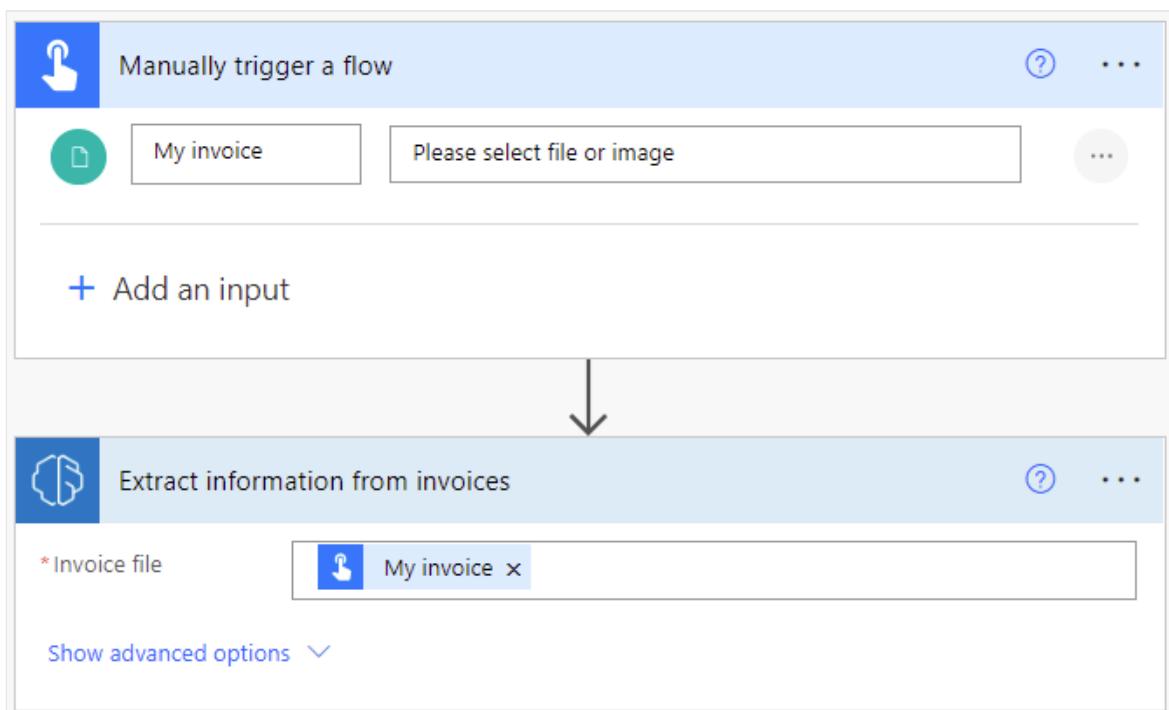
See also

[ID reader prebuilt model overview](#)

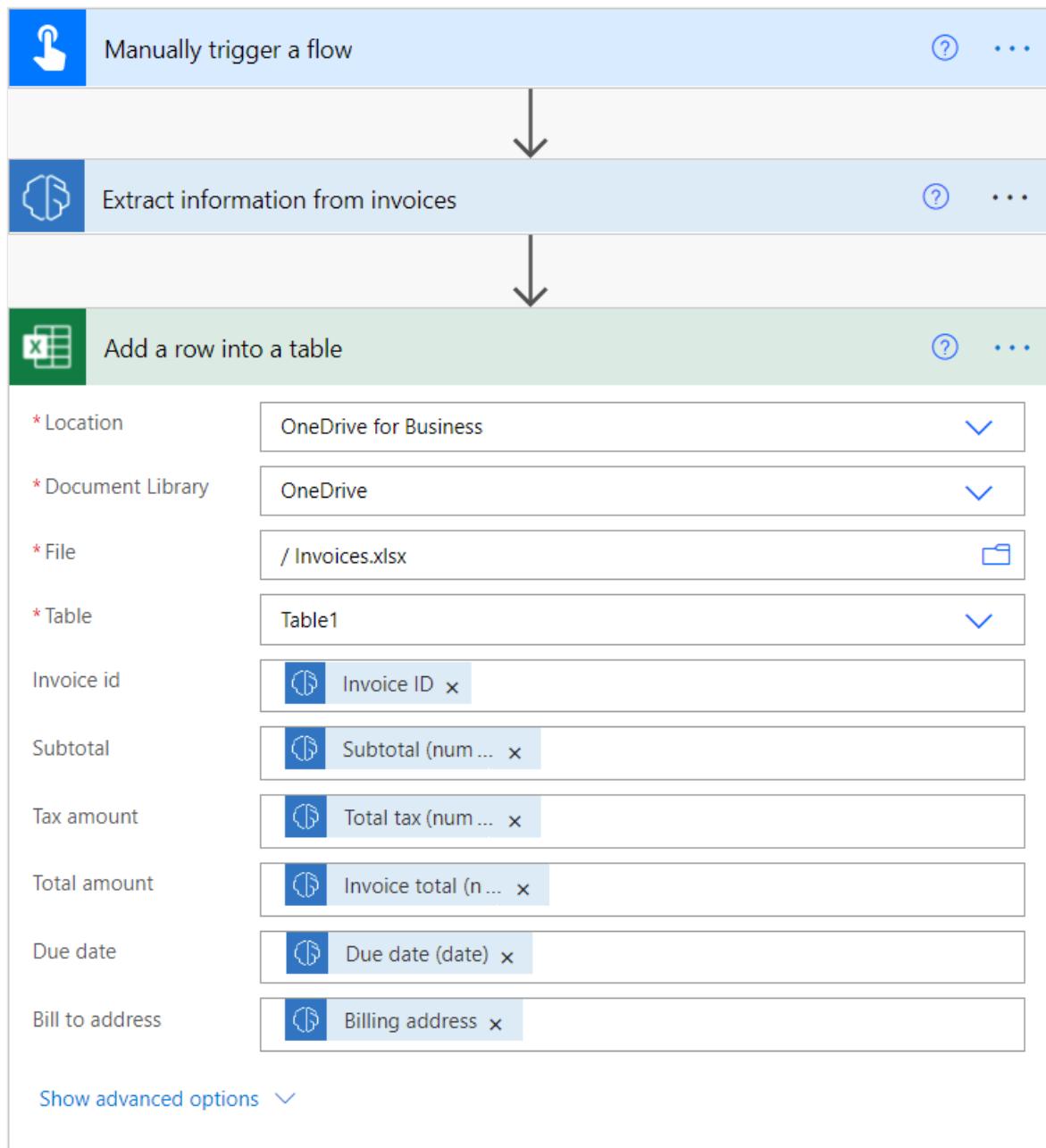
Use the invoice processing prebuilt model in Power Automate

Article • 08/01/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > File** as the input type.
5. Replace **File Content** with **My invoice** (also known as the title).
6. Select **+New step > AI Builder**, and then select **Extract information from invoices** in the list of actions.
7. Specify **My invoice** from the trigger in the **Invoice file** input.



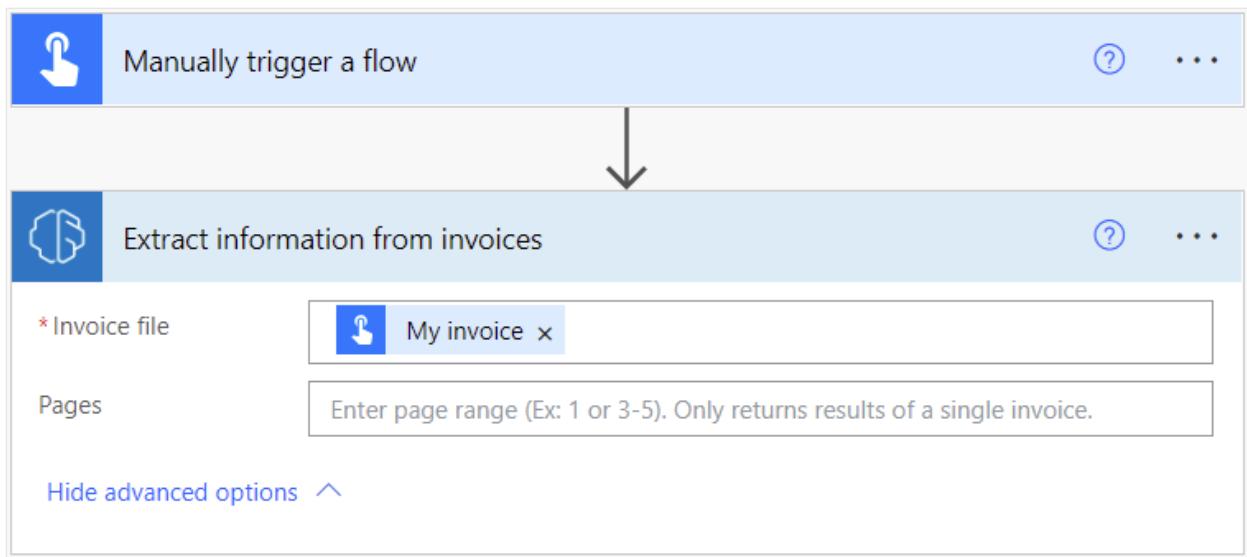
8. In the successive actions, you can use any of the invoice values from the [model output](#).



Congratulations! You've created a flow that uses the AI Builder invoice processing model. Select **Save** on the top right, and then select **Test** to try out your flow.

Page range

For large documents, it's possible to specify the page range to process.



You can enter a page value or page range in the **Pages** parameter. Example: 1 or 3-5.

! Note

If you have a large document with only one invoice, we strongly recommend to **use the Pages parameter to aim at your invoice, and therefore reduce the cost of model prediction and increase performance**. However, the page range should contain a **unique invoice** for the action to return correct data.

Example: A document contains a first invoice in page 2 and a second invoice that spans over pages 3 and 4:

- If you enter page range 2, it will return the data of the first invoice.
- If you enter page range 3-4, it will only return the data of the second invoice.
- If you enter page range 2-4, it will return partial data of the first and second invoices (should be avoided).

Parameters

Input

Name	Required	Type	Description
Receipt file	Yes	file	The invoice file to process
Pages	No	string	Page range to process

Output

Name	Type	Definition
Amount due (text)	string	Amount due as it's written on the invoice
Amount due (number)	float	Amount due in standardized number format. Example: 1234.98
Confidence of amount due	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Billing address	string	Billing address
Confidence of billing address	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Billing address recipient	string	Billing address recipient
Confidence of billing address recipient	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer address	string	Customer address
Confidence of customer address	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer address recipient	string	Customer address recipient
Confidence of customer address recipient	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer ID	string	Customer ID
Confidence of customer ID	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Customer name	string	Customer name
Confidence of customer name	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Due date (text)	string	Due date as it's written on the invoice
Due date (date)		Due date in standardized date format. Example: 2019-05-31T00:00:00Z
Confidence of due date	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Invoice date (text)	string	Invoice date as it's written on the invoice

Name	Type	Definition
Invoice date (date)	date	Invoice date in standardized date format. Example: 2019-05-31T00:00:00Z
Confidence of invoice date	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Invoice ID	string	Invoice ID
Confidence of invoice ID	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Invoice total (text)	string	Invoice total as it's written on the invoice
Invoice total (number)	float	Invoice total in standardized date format. Example: 2019-05-31T00:00:00Z
Confidence of invoice total	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Purchase order	string	Purchase order
Confidence of purchase order	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Remittance address	string	Remittance address
Confidence of remittance address	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Remittance address recipient	string	Remittance address recipient
Confidence of remittance address recipient	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Service address	string	Service address
Confidence of service address	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Service address recipient	string	Service address recipient
Confidence of service address recipient	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Shipping address	string	Shipping address
Confidence of shipping address	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).

Name	Type	Definition
Shipping address recipient	string	Shipping address recipient
Confidence of shipping address recipient	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Subtotal (text)	string	Subtotal as it's written on the invoice
Subtotal (number)	float	Subtotal in standardized number format. Example: 1234.98
Confidence of subtotal	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Total tax (text)	string	Total tax as it's written on the invoice
Total tax (number)	float	Total tax in standardized number format. Example: 1234.98
Confidence of total tax	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Vendor address	string	Vendor address
Confidence of vendor address	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Vendor address recipient	string	Vendor address recipient
Confidence of vendor address recipient	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Vendor name	string	Vendor name
Confidence of vendor name	float	How confident the model is in its prediction. Score between 0 (low confidence) and 1 (high confidence).
Detected text	string	Line of recognized text from running OCR on an invoice. Returned as a part of a list of text.
Page number of detected text	integer	Which page the line of recognized text is found on. Returned as a part of a list of text.

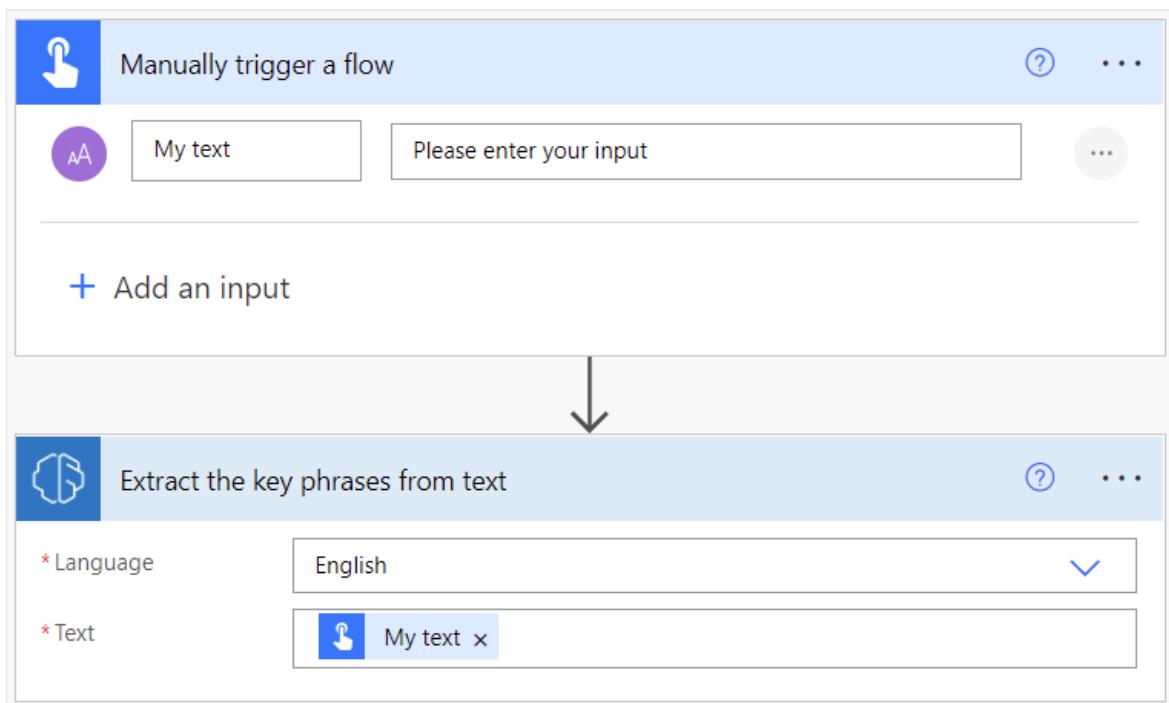
Related topics

[Invoice processing overview](#)

Use the key phrase extraction prebuilt model in Power Automate

Article • 08/01/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
5. Replace **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Extract the key phrases from text** in the list of actions.
7. In the **Language** input, select or enter your language.
8. In the **Text** input, select **My Text** from the **Dynamic content** list:



9. In the successive actions, you can use any columns extracted by the AI Builder model. For example, you can create a Microsoft Dataverse row for each **Key phrase**.

* Select an output from previous steps
results x

Add a new row

* Table name: Phrases

* Key Phrase: Key phrase x

* Text: My text x

Show advanced options ▾

Parameters

Input

Name	Required	Type	Description	Values
Text	Yes	string	Text to analyze	Text sentences
Language	Yes	string	Language of the text to analyze	Item in a list of predefined languages or a language code (ex.: "en", "fr", "zh_chs", "ru")

Output

Name	Type	Description
Key phrase	string	String denoting a key talking points in the analyzed text. As there could be multiple key phrases, selecting this parameter will create an apply to each loop

Congratulations! You have created a flow that uses your key phrase extraction AI model. Select **Save** on the top right and then select **Test** to try out your flow.

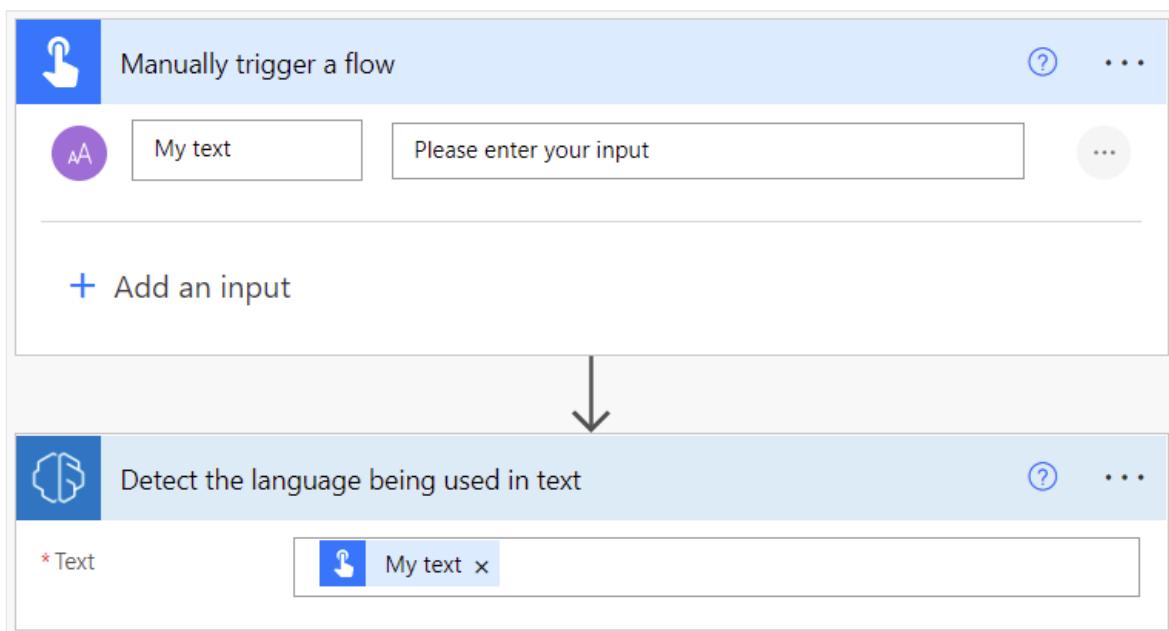
See also

[Key phrase extraction overview](#)

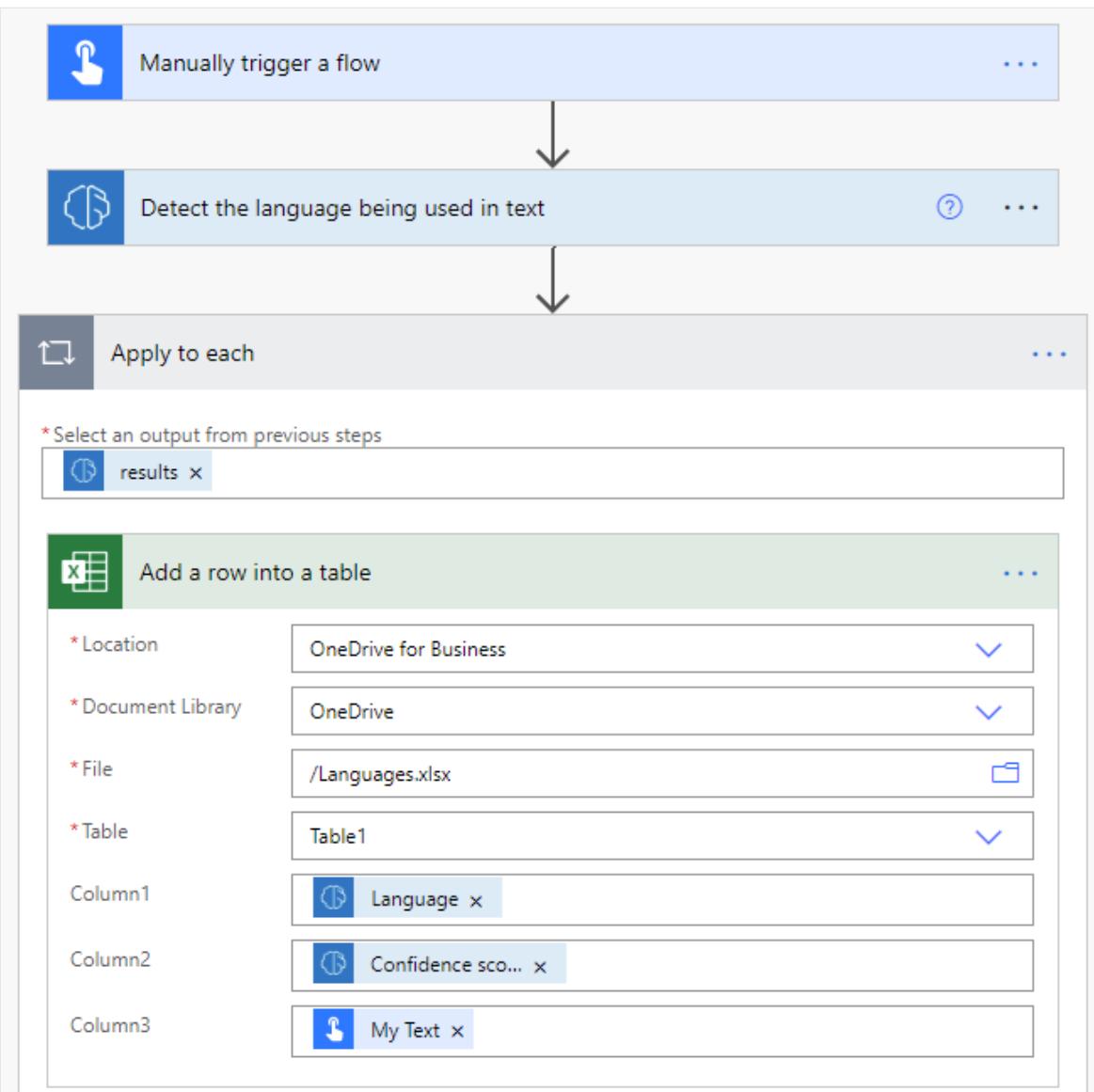
Use the language detection prebuilt model in Power Automate

Article • 03/05/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
5. Replace **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Detect the language being used in text** in the list of actions.
7. Select the **Text** input, and then select **My Text** from the **Dynamic content** list:



8. In the successive actions, you can use any columns extracted by the AI Builder model. For example, you can add lines to an Excel file using **Language** and **Confidence score**:



Congratulations! You've created a flow that uses a language detection model. Select **Save** on the top right and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description	Values
Text	Yes	string	Text to analyze	Text sentences

Output

Name	Type	Description	Values

Name	Type	Description	Values
results	list	A list of languages detected in the input text	List of score and languages
Confidence score	float	How confident the model is in its prediction	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
Language	string	Language inferred from the text	Language code (ex.: "en", "fr", "zh_chs", "ru")

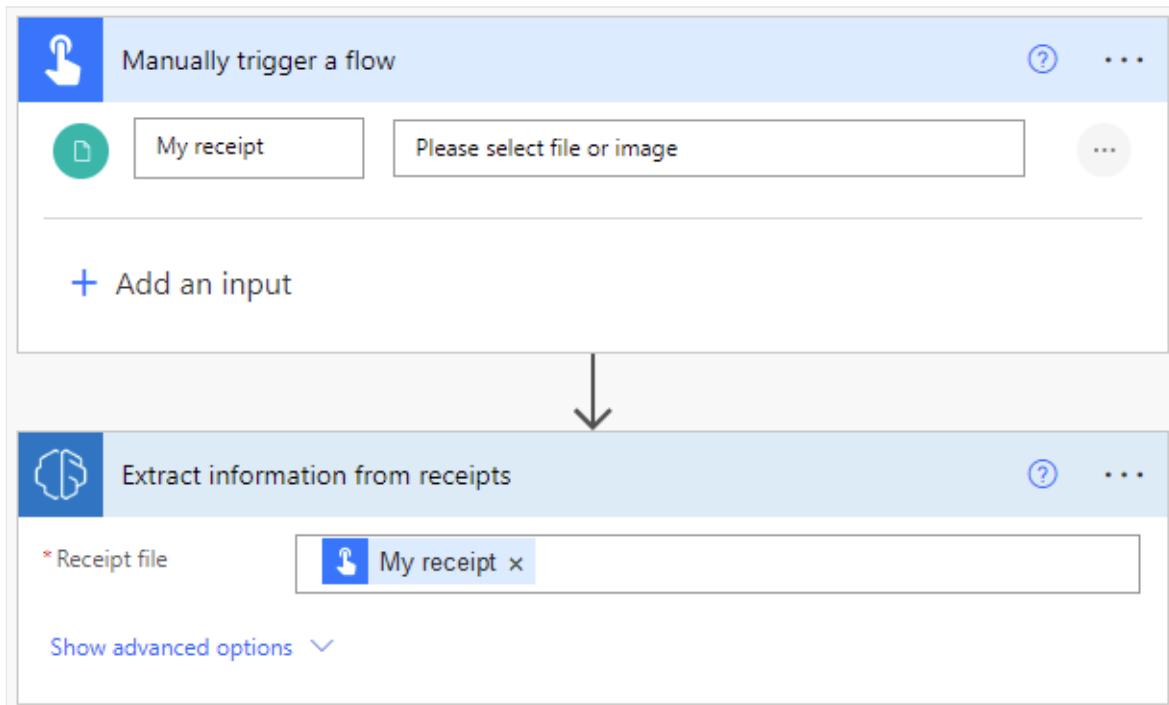
See also

[Language detection overview](#)

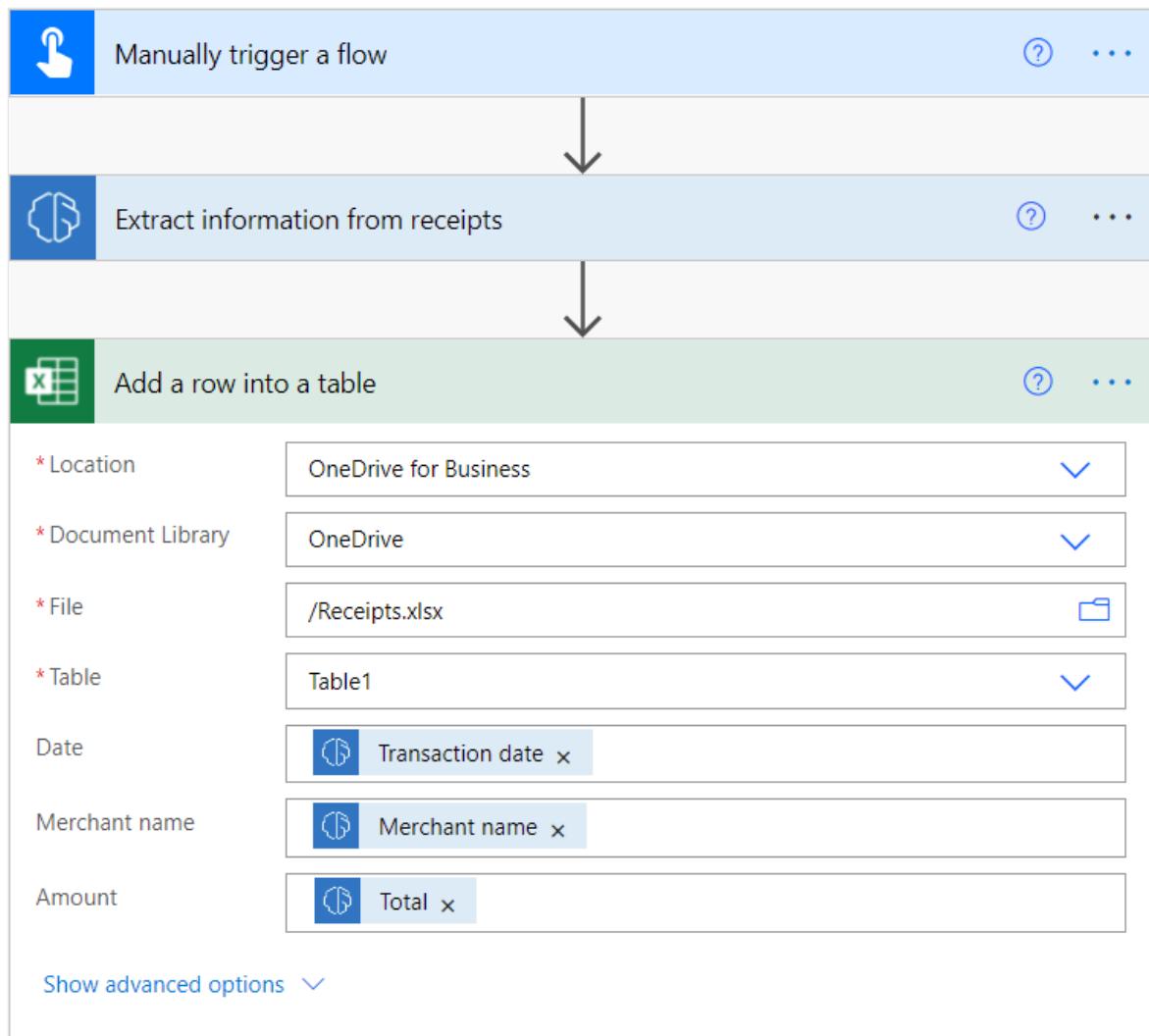
Use the receipt processing prebuilt model in Power Automate

Article • 01/05/2023

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > File** as the input type.
5. Replace **File Content** with **My receipt** (also known as the title).
6. Select **+New step > AI Builder**, and then select **Extract information from receipts** in the list of actions.
7. Select the **Receipt file** input, and then select **My receipt** from the **Dynamic content** list:



8. In the successive actions, you can use any of the receipt values from the [model output](#) section below.



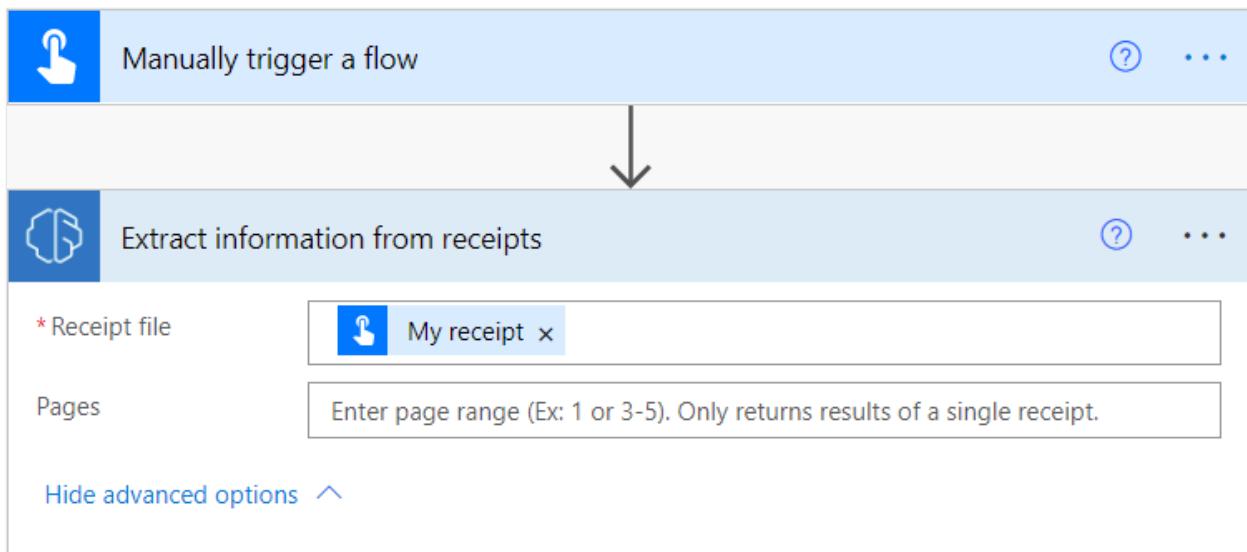
! Note

Receipt values are returned as strings. To manipulate them as numbers, you can use the **float** or **int** conversion functions.

Congratulations! You've created a flow that uses the AI Builder receipt processing model. Select **Save** on the top right, and then select **Test** to try out your flow.

Page range

For large documents, it's possible to specify the page range to process.



You can enter a page value or page range in the **Pages** parameter. Example: 1 or 3-5.

Note

If you have a large document with only one receipt, we strongly recommend to **use the Pages parameter to aim at your receipt, and therefore reduce the cost of model prediction and increase performance**. However, note that only the data of the **first receipt** within the page range will be returned and that multi-page receipts are not supported.

Example: A document contains a first receipt in page 2 and a second receipt that spans overs page 3 and 4:

- If you enter value 2, it will return the data of the first receipt
- If you enter value 3-4, it will only return the data of the first page of the second receipt
- If you enter value 2-4, it will only return data of the first receipt, not the data of the second receipt

Parameters

Input

Name	Required	Type	Description
Receipt file	Yes	string	The receipt file to process
Pages	No	string	Page range to process

Output

Name	Type	Description
Merchant name	string	Merchant name
Merchant address	string	Merchant address
Merchant phone number	string	Merchant phone number
Transaction date	string	Transaction date
Transaction time	string	Transaction time
Purchased item name	string	Purchased item name. Returned as a part of a list of items.
Purchased item quantity	string	Purchased item quantity. Returned as a part of a list of items.
Purchased item price	string	Purchased item price. Returned as a part of a list of items.
Purchased item total price	string	Purchased item total price. Returned as a part of a list of items.
Subtotal	string	Subtotal
Tax	string	Tax
Tip	string	Tip
Total	string	Total
Confidence of merchant name	float	How confident the model is in its detection
Confidence of merchant address	float	How confident the model is in its detection
Confidence of merchant phone number	float	How confident the model is in its detection
Confidence of transaction date	float	How confident the model is in its detection
Confidence of transaction time	float	How confident the model is in its detection
Confidence of purchased item name	float	How confident the model is in its detection. Returned as a part of a list of items.

Name	Type	Description
Confidence of purchased item quantity	float	How confident the model is in its detection. Returned as a part of a list of items.
Confidence of purchased item price	float	How confident the model is in its detection. Returned as a part of a list of items.
Confidence of purchased item total price	float	How confident the model is in its detection. Returned as a part of a list of items.
Confidence of subtotal	float	How confident the model is in its detection
Confidence of tax	float	How confident the model is in its detection
Confidence of tip	float	How confident the model is in its detection
Confidence of total	float	How confident the model is in its detection
Detected text	string	Line of recognized text. Returned as a part of a list of text.
Page number of detected text	integer	Which page the line of recognized text is found on. Returned as a part of a list of text.
Height of detected text	float	Height of the line of text. Returned as a part of a list of text.
Left position of detected text	float	Left position of the line of text. Returned as a part of a list of text.
Top position of detected text	float	Top position of the line of text. Returned as a part of a list of text.
Width of detected text	float	Width of the line of text. Returned as a part of a list of text.

See also

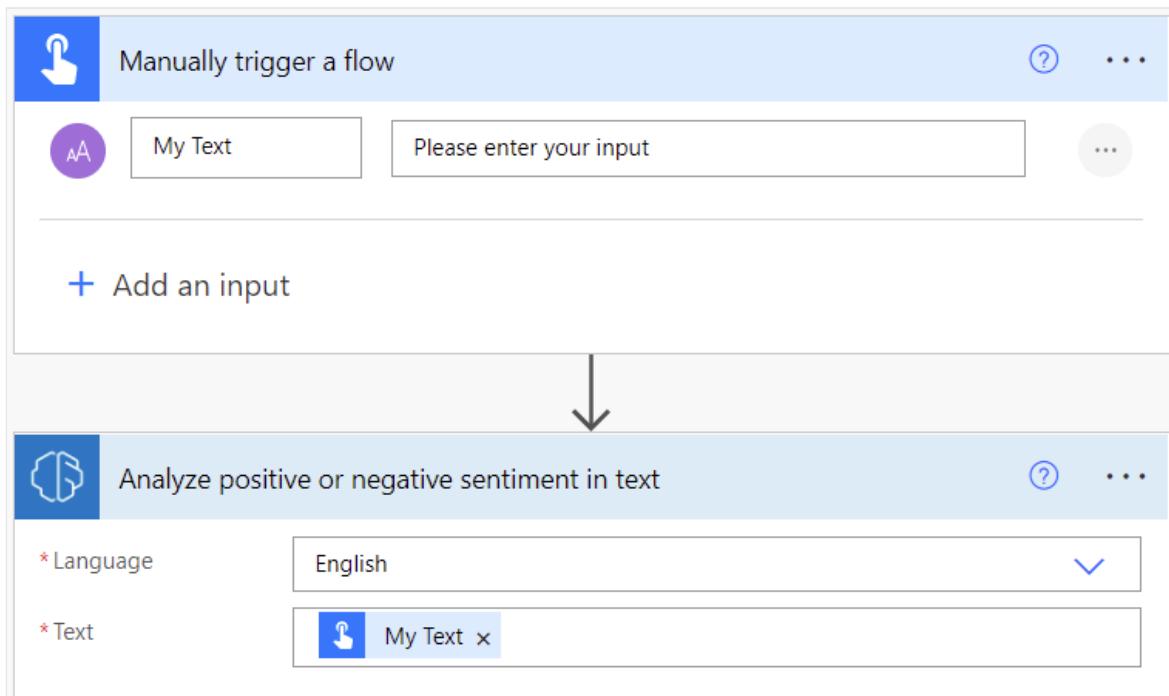
- [Receipt processing overview](#)
- [Training: Process receipts with AI Builder \(module\)](#)

Use the sentiment analysis prebuilt model in Power Automate

Article • 03/05/2022

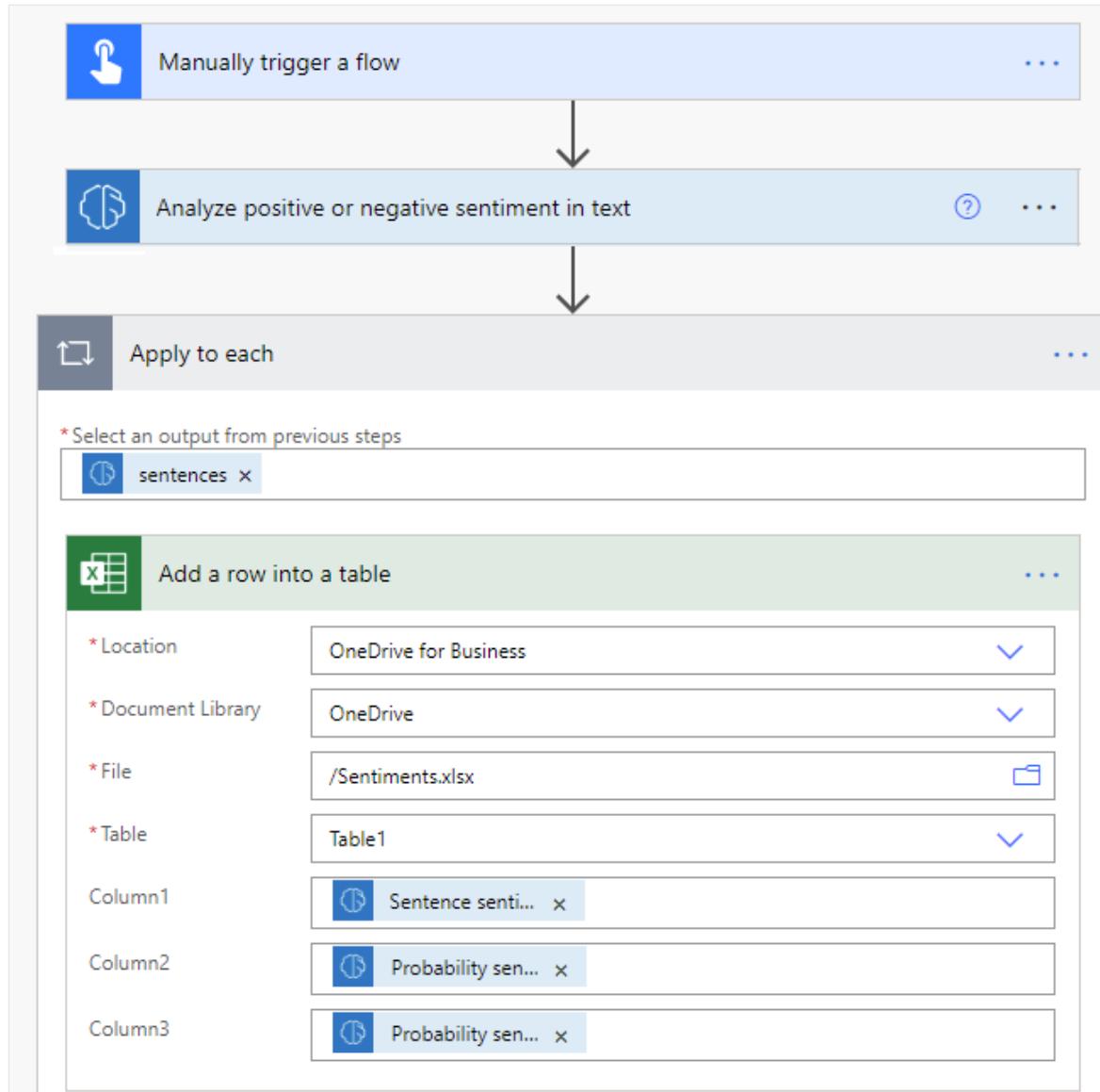
In this article, we will create a flow that uses the AI Builder sentiment analysis prebuilt model.

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text**.
5. Replace **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Analyze positive or negative sentiment** in the list of actions.
7. In the **Language** input, select or enter your language.
8. In the **Text** input, select **My Text** from the **Dynamic content** list:



9. In the successive actions, you can use any columns extracted by the AI Builder model. For example, you can add lines to an Excel file for each sentence using

Sentence sentiment, Probability sentence is positive and Probability sentence is negative:



Congratulations! You've created a flow that uses the sentiment analysis model. Select **Save** on the top right, and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description	Values
Text	Yes	string	Text to analyze	Text sentences
Language	Yes	string	Language of the text to analyze	Item in a list of predefined languages or a language code (ex.: "en", "fr", "zh_chs", "ru")

Output

Name	Type	Description	Values
Overall text sentiment	string	Overall sentiment of the analyzed text	Positive, neutral or negative
Probability overall text is positive	float	Probability of the positive sentiment in the analyzed text	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
Probability overall text is negative	float	Probability of the negative sentiment in the analyzed text	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
Probability overall text is neutral	float	Probability of the neutral sentiment in the analyzed text	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
documentScores	object	Object containing overall scores	Positive, neutral and negative scores
sentences	List	List of sentence data structures containing sentences overall sentiment and scores	Sentence sentiment, positive, neutral and negative scores
Sentence sentiment	string	Sentiment of the analyzed sentence	Positive, neutral or negative
Probability sentence is positive	float	Probability of the positive sentiment in the analyzed sentence	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
Probability sentence is negative	float	Probability of the negative sentiment in the analyzed sentence	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
Probability sentence is neutral	float	Probability of the neutral sentiment in the analyzed sentence	Value in the range of 0 to 1. Values close to 1 indicate greater confidence that the identified sentiment is accurate
sentenceScores	object	Data structure containing sentence scores	Positive, neutral, and negative scores

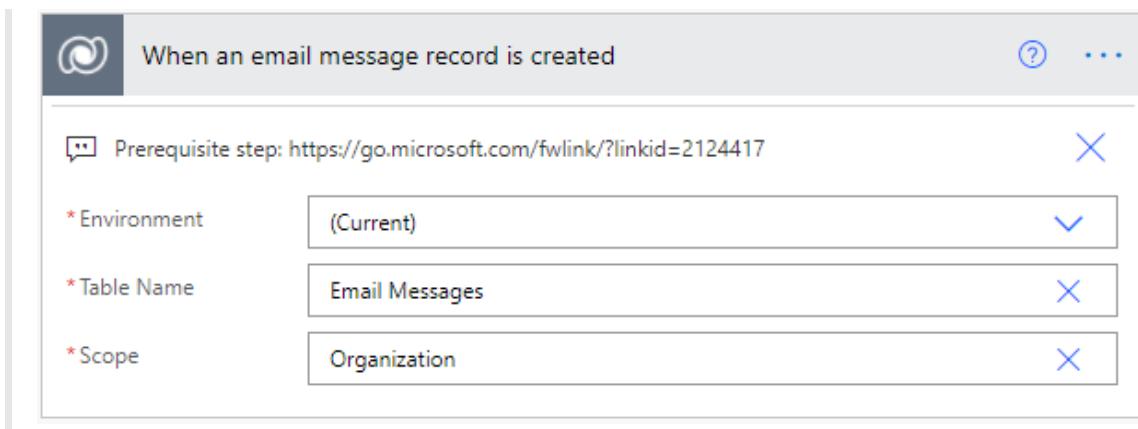
Use sentiment analysis to analyze incoming Dynamics 365 emails

Power Automate provides a template that enables you to analyze incoming Dynamics 365 emails by using AI Builder sentiment analysis. This template requires some customization of your Microsoft Dataverse email table before you can use it.

1. Create an attribute in your Email Messages table in which to save the sentiment analysis results.

For information about how to create an attribute, see [Create and edit column for Dataverse using Power Apps portal](#).

2. Sign in to [Power Automate](#).
3. In the left pane, select **Templates**, and then search for **ai builder sentiment**.
4. Select **Analyze sentiment of Dynamics emails using AI Builder**.
5. Select your environment, and then select **Continue**.
6. Type **Email Messages** in the **Entity Name** input.
7. Type **Organization** in the **Scope** input.



8. Next, the template shows messages from **draft emails** and **received emails**. You can filter these if you want to perform sentiment analysis only on selected email statuses. For a list of status codes, see [StatusCode choices](#).
9. Select **Add sentiment to CDS Email Entity**, select **Show advanced options**, and then locate the attribute you added in step 1.
10. Finally, add **Global sentiment** from the **Dynamic content** list.

The screenshot shows the Microsoft Flow designer interface. On the left, there's a list of fields for an 'Incoming' item, including 'Send Later', 'Sentiment Analysis' (which has a red box around its 'Global sentiment...' entry), 'Sort Date', 'Start Date', 'Status Reason Value', 'Sub-Category', 'Subject', 'Submitted By', 'Time Zone Rule Version Number', 'To Recipients', 'Tracking Token', 'UTC Conversion Time Zone Code', 'Currency', and 'ID for template record'. On the right, a sidebar titled 'used in this flow' is open, showing 'Dynamic content' and 'Expression' tabs. Under 'Dynamic content', there's a search bar and a list of items: 'Global negative score', 'Global neutral score', 'Global positive score', 'Global sentiment', 'Sentence length', 'Sentence negative score', 'Sentence neutral score', and 'Sentence offset'. Each item has a small icon next to it.

If you want this column to be visible in your email grid view, follow these steps:

1. Go to the view/form designer, and add the custom column you created in step 1 of the preceding procedure. For information about how to add the column to your view, see [Add a column to your view](#).
2. Add a field to the form. For details, see [Add a field to a form](#).

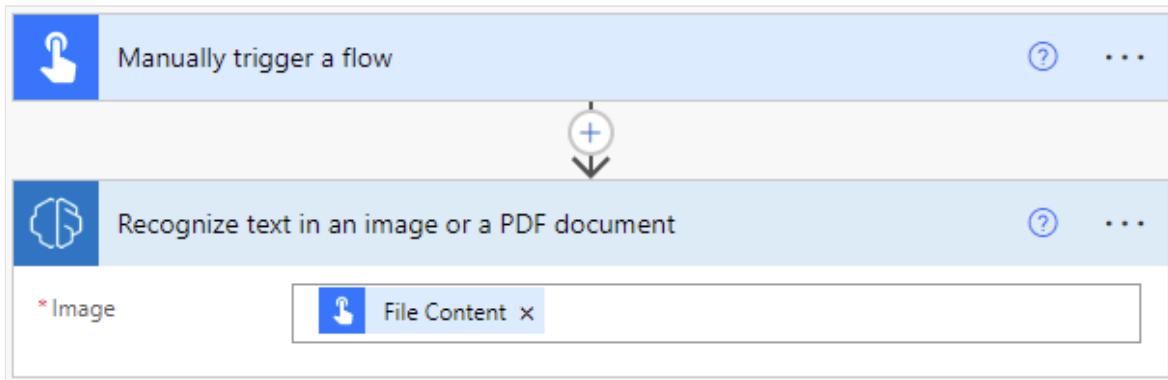
See also

[Sentiment analysis overview](#)

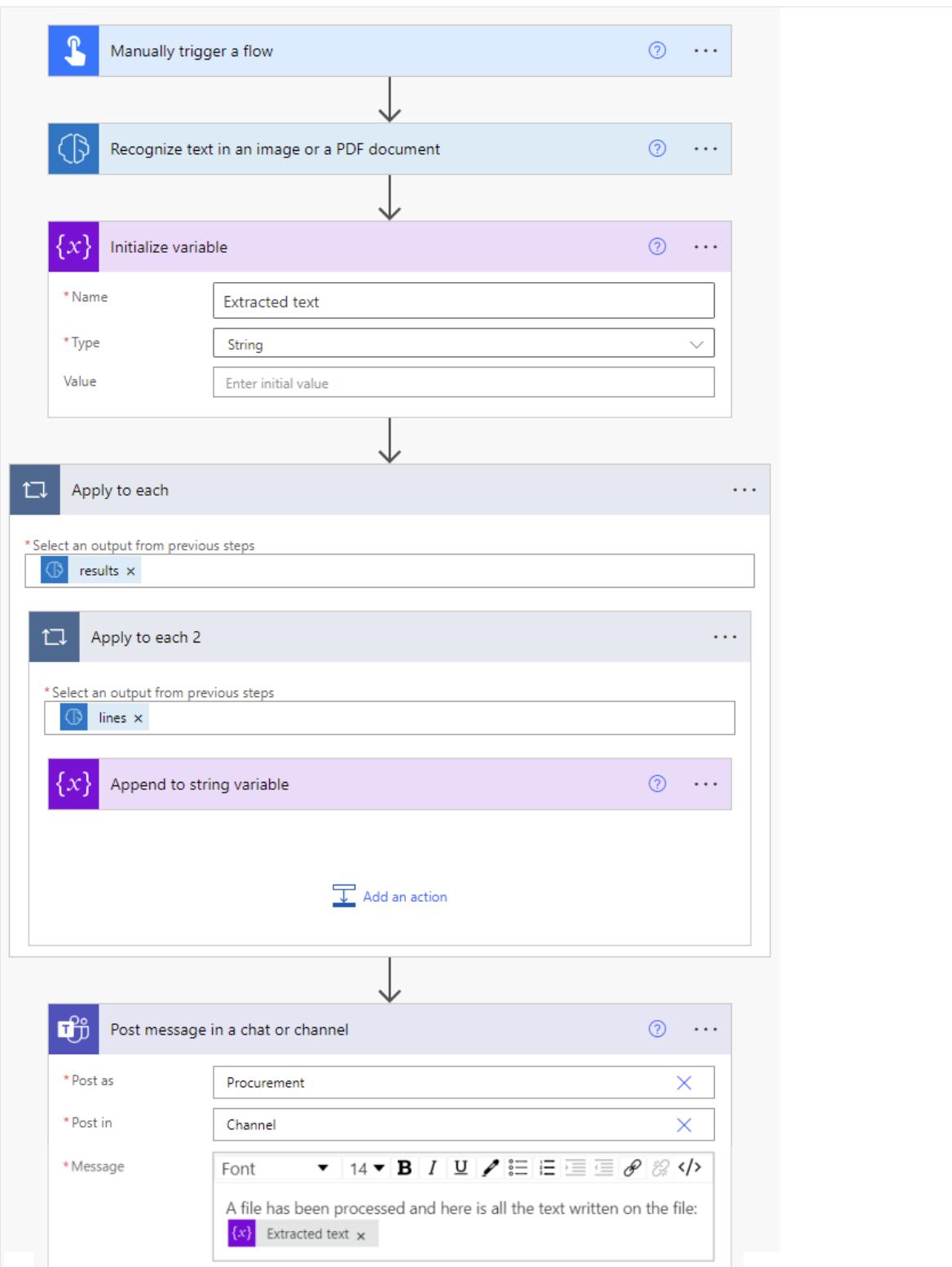
Use the text recognition prebuilt model in Power Automate

Article • 03/05/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > File** as the input type.
5. Select **+New step > AI Builder**, and then select **Recognize text in an image or a PDF document** in the list of actions.
6. Select the **Image** input, and then select **File Content** from the **Dynamic content** list:



7. To process results, select **+New step > Control**, and then select **Apply to each**.
8. Select the input, and then select **lines** from the Dynamic content list. This will add **results** to the input and automatically create another **Apply to each** action.
9. In the successive actions, you can use any columns extracted by the AI Builder model. For example, you can extract **Text** into a variable, and then post all the extracted text in a Teams channel:



Congratulations! You've created a flow that uses a text recognition model. You can continue to build on this flow until it suits your needs. Select **Save** on the top right, and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Type	Description
Image	Yes	file	Image to analyze

Output

The detected text is embedded into `lines` sub list of the `results` list. You first need to select the `lines` column from an **Apply to each** action to view all the following columns.

Name	Type	Description
Text	string	Strings containing the line of text detected
Page number	string	Page number of the text detected
Coordinates	float	Coordinates of the text detected

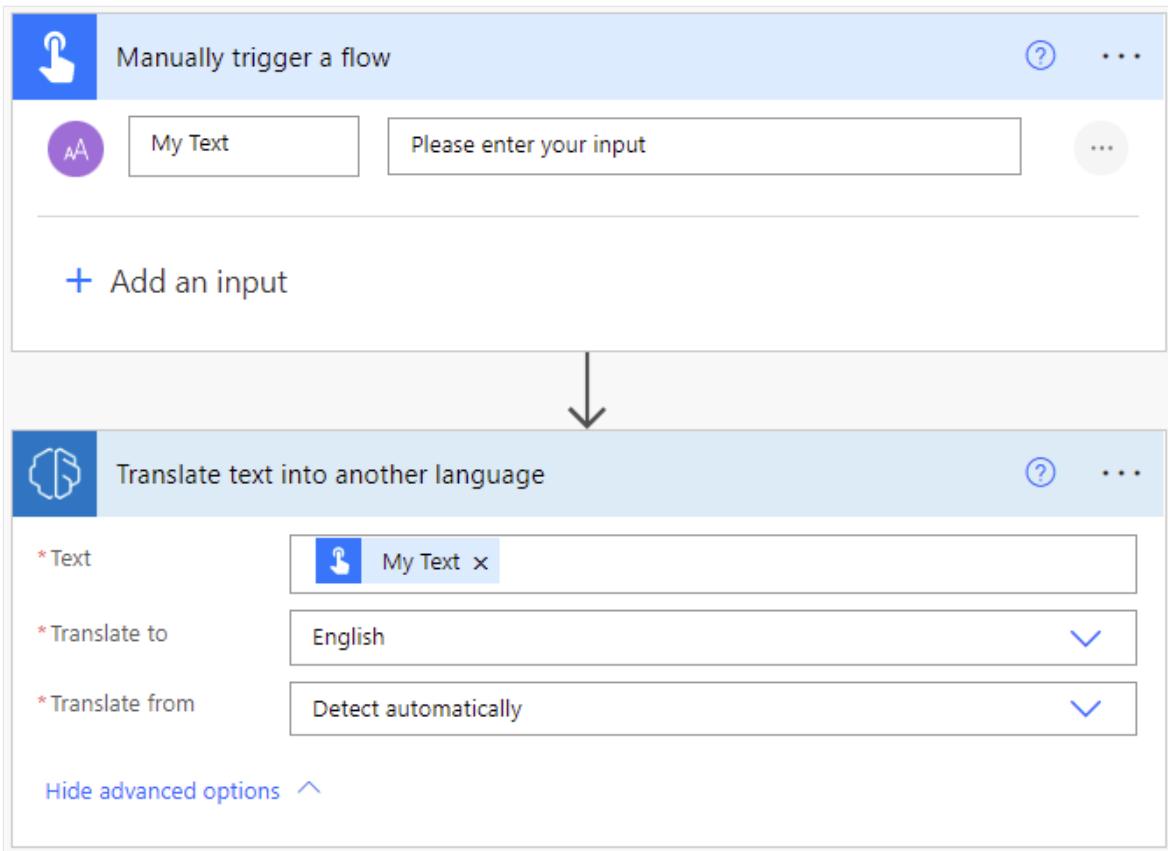
See also

[Text recognition overview](#)

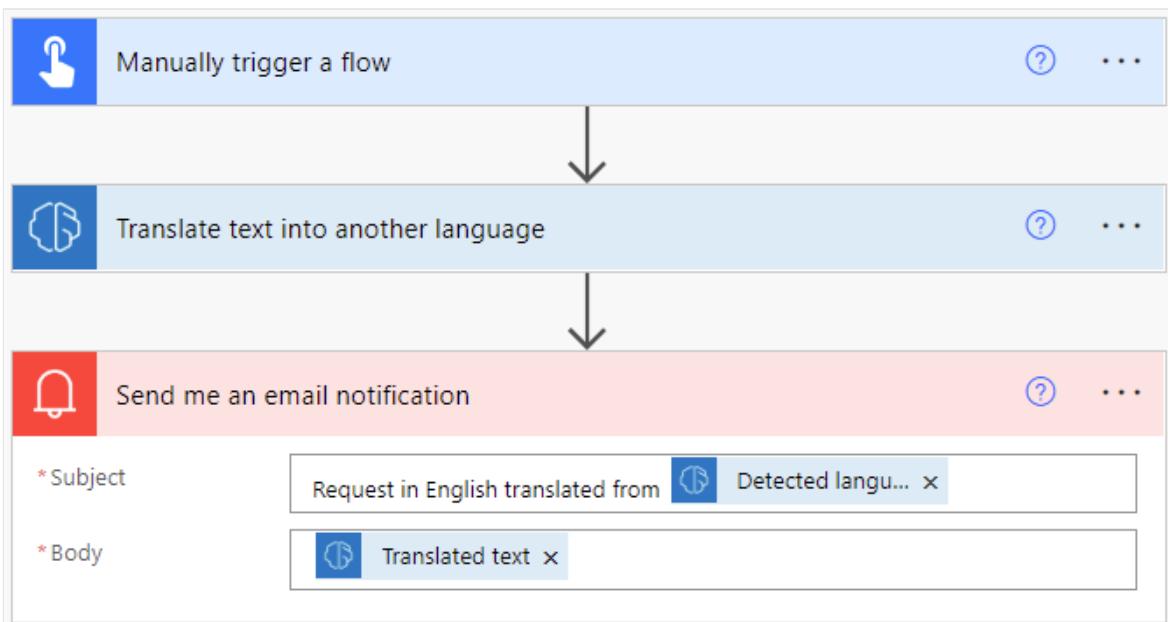
Use the text translation prebuilt model in Power Automate

Article • 10/26/2022

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Expand **Manually trigger a flow**, and then select **+Add an input > Text** as the input type.
5. Replace **Input** with **My Text** (also known as the title).
6. Select **+ New step > AI Builder**, and then select **Translate text into another language** in the list of actions.
7. Select **My Text** from the **Dynamic content** list.
8. Select the target language in the **Translate to** input.
9. Optionally, select the source language in the **Translate from** input to specify the language of the text in **My Text** input. If you don't specify this input, the model will automatically detect the source language for you.



10. In the successive actions, you can use any columns extracted by the AI Builder model. For example, you can use get a notification of the translated text sent out to your email using the **Send me an email notification** and the output property **Text** from the text translation model.



Congratulations! You've created a flow that uses a text translation model. Select **Save** on the top right and then select **Test** to try out your flow.

Parameters

Input

Name	Required	Description	Values
Text	Yes	Text to translate	Text sentences
Translate to	Yes	Target language of the translated text	Item in a list of predefined languages or a language code (ex.: "en", "fr", "zh_chs", "ru")
Translate from	No	Language of the text to be translated	Item in a list of predefined languages or a language code (ex.: "en", "fr", "zh_chs", "ru")

Output

Name	Description	Values
Text	Translated version of the input text	Text sentences
Detected language	Detected language of the input text if the column "Translated from" wasn't specified	Language code (ex.: "en", "fr", "zh_chs", "ru")

Related topic

[Text translation overview](#)

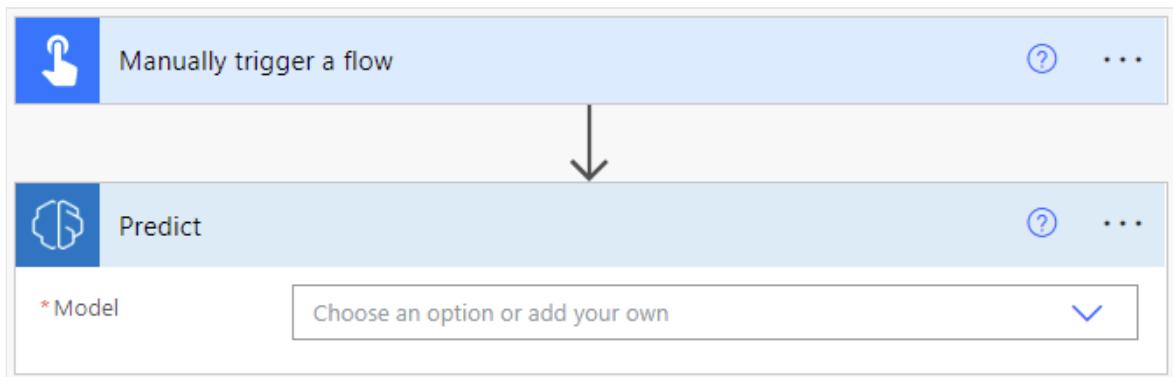
Use predict action in Power Automate

Article • 03/13/2023

You can use dedicated actions for each AI Builder model in Power Automate. However, the **predict** action lets you use many AI Builder model types.

Use a custom or prebuilt model

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Select **+ New step**, and then enter **predict** in the search bar.
5. Select **Predict from AI Builder** or **Predict using AI Builder models from Microsoft Dataverse**. Both actions offer the same features.



6. In the **Model** input, select a custom model you created or choose a prebuilt model. Here is the list of the prebuilt models available:

- Business card reader: BusinessCard model
- Category classification (prebuilt): Category/Classification model
- Entity extraction (prebuilt): EntityExtraction model
- Key phrase extraction: KeyPhraseExtraction model
- Language detection: LanguageDetection model
- Sentiment analysis: SentimentAnalysis model
- Text recognition: TextRecognition model

ⓘ Note

To learn more about the input and output parameters of each model, refer to the documentation explaining how to use the selected model in the following documentation sections:

- Use a custom AI Builder model in Power Automate
- Use a prebuilt AI Builder model in Power Automate

Use a dynamic model ID (advanced)

For some complex use cases, you may need to pass a model ID dynamically to the predict action. For example, if you want to process different types of invoices using different models, you may want to automatically choose a model depending on the type of invoice.

In this section, we'll explain how to configure the AI Builder predict action for this specific purpose depending on the model type.

1. Sign in to [Power Automate](#).
2. Select **My flows** in the left pane, and then select **New flow > Instant cloud flow**.
3. Name your flow, select **Manually trigger a flow** under **Choose how to trigger this flow**, and then select **Create**.
4. Select **+ New step**.
5. Enter **Initialize variable** in the search bar, and then select it in the **Actions** tab.
6. Enter **model id** in the **Name** input, **String** in the **Type** input, and the actual model ID in the **Value** input.

The model ID can be found in the URL of the model's detail page in Power Apps:
make.powerapps.com/environment/[environment id]/aibuilder/models/[model id]

7. Select **+ New step**, search for **predict** and then select **Predict from AI Builder**.
8. Select the input > **Enter custom value**, and then enter **model id** from step 6.

The **Infer request** column value depends on the model type.

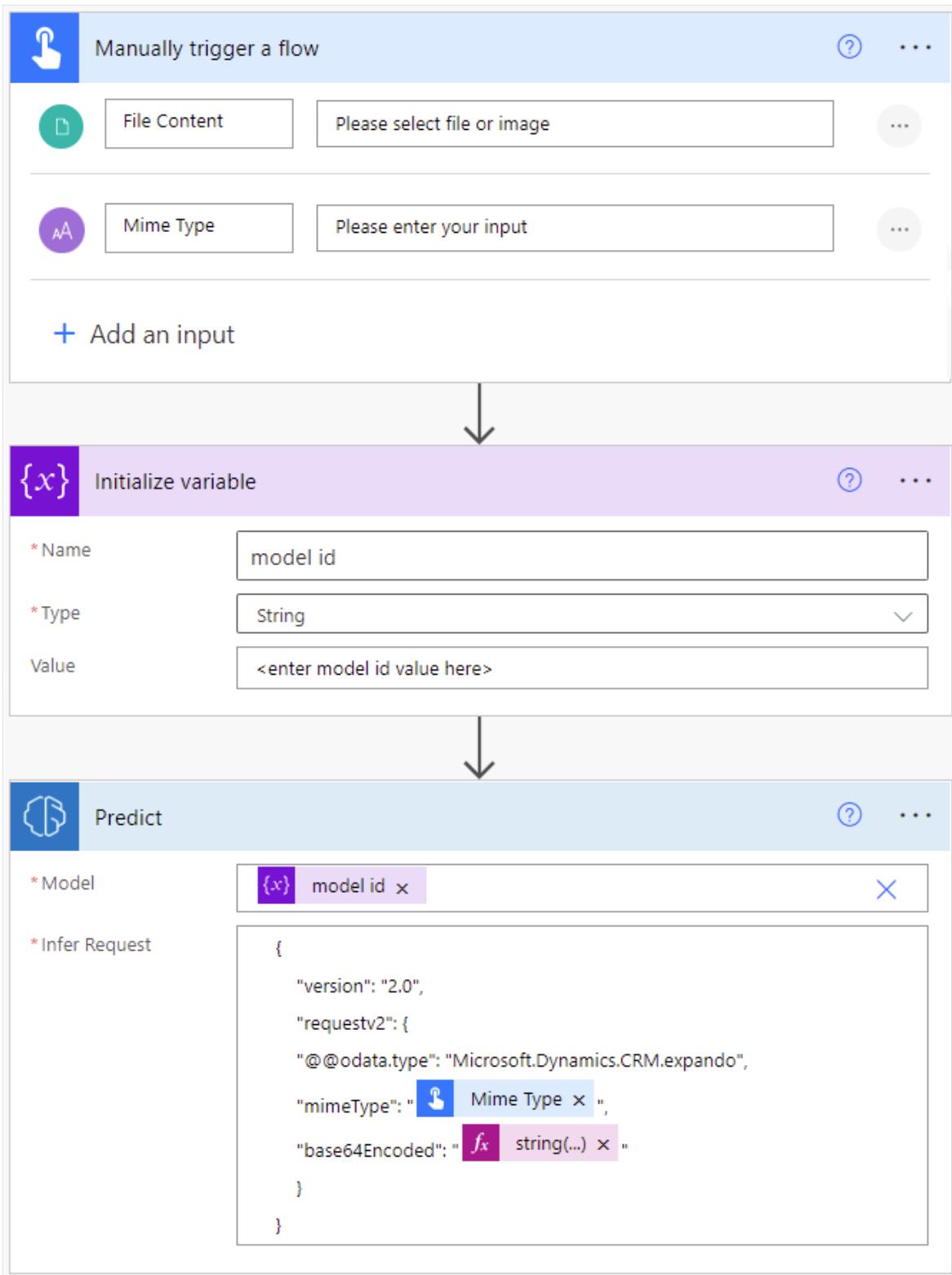
Document processing model

1. In the step **Manually trigger a flow**, add a **File** input, and set its name to **File Content**.

2. In the step **Manually trigger a flow**, add a **Text** input, and set its name to **Mime Type**.
3. In the step **Initialize variable**, enter a document processing model ID.
4. In the step **Predict**, enter following value in the **Infer request** column:

JSON

```
{  
    "version": "2.0",  
    "requestv2": {  
        "@@odata.type": "Microsoft.Dynamics.CRM.expandable",  
        "mimeType": "@{triggerBody()['text']}",  
        "base64Encoded": "@{string(triggerBody()?['file'])?['contentBytes'])}"  
        "pageRange": "1"  
    }  
}
```



5. Select **Save** in the upper-right corner, and then select **Test** to try out your flow:

Run flow

X

FP dynamic

Owner:

File Content

InvoiceFrom.png

Import

Mime Type *

image/png

This flow uses Common Data Service (current environment).

[Review connections and actions](#)

Run flow

Cancel

6. In the flow run details, get the model JSON output in the **OUTPUTS** section of the predict action. This output is useful to build downstream actions using values of the model.

Predict

12s

INPUTS

Model

Infer Request

```
{ "version": "2.0", "requestv2": { "mimeType": "image/png", "base64Encoded": "iVBORw0KGgoAAAANSUhEUgAABMYAAAYsCAYAAADtTYEBAQ@odata.type": "Microsoft.Dynamics.CRM.expando" } }
```

Infer Request

```
{ "version": "2.0", "requestv2": { "mimeType": "image/png", "base64Encoded": "iVBORw0KGgoAAAANSUhEUgAABMYAAAYsCAYAAADtTYEBAQ@odata.type": "Microsoft.Dynamics.CRM.expando" } }
```

OUTPUTS

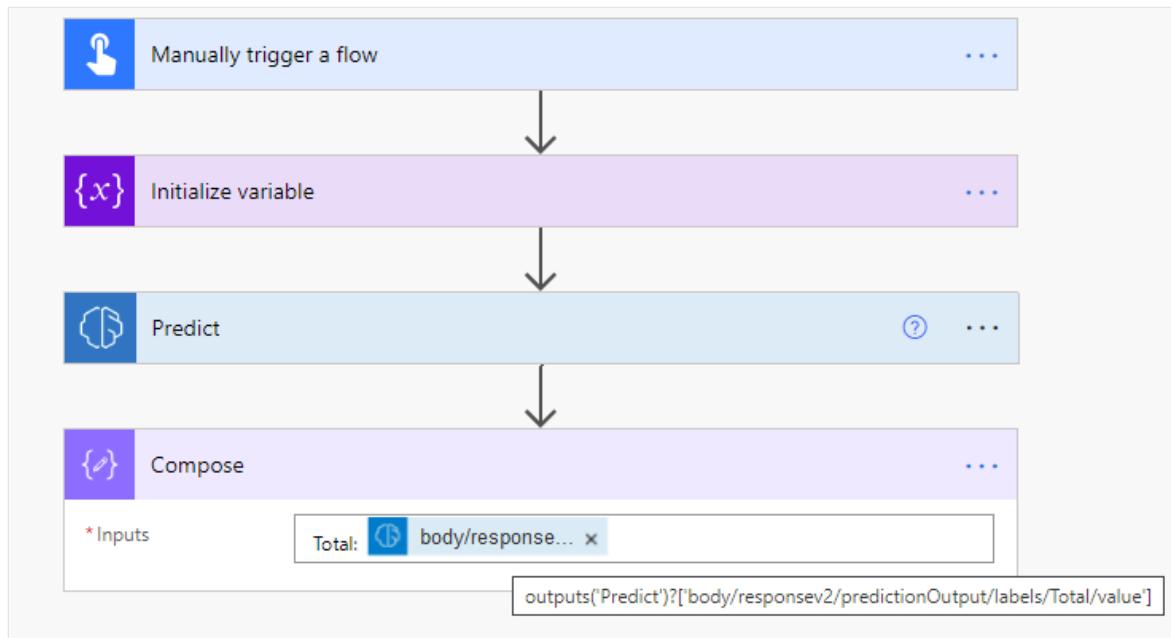
body

"labels": ["@odata.type": "#Microsoft.Dynamics.CRM.expando", "Total": { "@odata.type": "#Microsoft.Dynamics.CRM.expando", "displayName": "Total", "value": "\$ 1013.50", "confidence": 1, "localization": "en-US" }] }

7. Go back to your flow in edit mode.

8. Select **+ New step** and select the **Compose** action (or any other action to process your model output). Let's say your model output has the **Total** column. You can get it with the following formula:

```
@{outputs('Predict')?  
['body/responsev2/predictionOutput/labels/Total/value']}
```



Object detection model

This process is similar to the infer request in step 4 in the *Document processing model* section:

JSON

```
{  
    "version": "2.0",  
    "requestv2": {  
        "@@odata.type": "Microsoft.Dynamics.CRM.expando",  
        "base64Encoded": "@{string(triggerBody())?['file']?  
['contentBytes'])}"  
    }  
}
```

Category classification model

This process is similar to the infer request in step 4 in the *Document processing model* section:

JSON

```
{  
    "version": "2.0",
```

```
"requestv2": {  
    "@@odata.type": "Microsoft.Dynamics.CRM.expando",  
    "language": "Detect automatically",  
    "text": "The text to categorize"  
}  
}
```

AI Builder in Power Apps overview

Article • 01/05/2023

You can use AI Builder in Microsoft Power Apps in one of two ways, depending on the model you will be using. You can:

- Use AI models in the formula bar, or
- Add AI Builder components

You can now use [Power Fx](#) expressions to consume AI Builder models in Power Apps. This feature is available in preview.

Use AI models in the formula bar

You can get support for the following AI Builder models that use Power Apps through the formula bar. To learn more, see [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Model type	Build type
Sentiment analysis	Prebuilt
Entity extraction	Prebuilt and Custom
Key phrase extraction	Prebuilt
Language detection	Prebuilt
Category classification	Prebuilt and Custom

Add AI Builder components

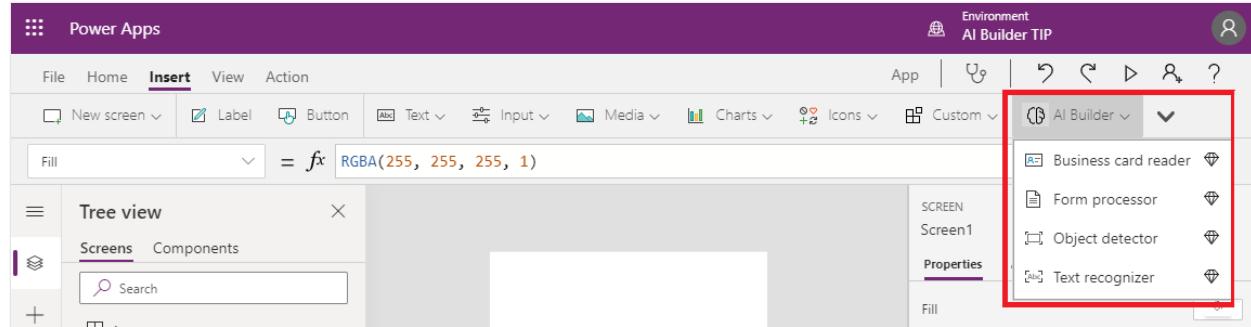
AI Builder provides two kinds of Power Apps components. Choose your component based on the models you want to use.

- Components that use prebuilt AI models that are ready to use right away:
 - [Business card reader \(canvas app\)](#)
 - [Business card reader \(model-driven app\)](#)
 - [Receipt processor \(canvas app\)](#)
 - [Text recognizer \(canvas app\)](#)

For more information on canvas apps, see [What are canvas apps in Power Apps?](#)

- Components that use custom AI models that you build and train:
 - Form processor
 - Object detector

The AI Builder components for [canvas apps](#) are available in [Power Apps Studio](#) and appear on the **Insert** tab when you build your canvas app.



Use Power Fx to consume AI models in Power Apps (preview)

The easiest, most versatile way you can build expressions for AI Builder models in Power Apps is with [Power Fx](#). This is the low-code formula language that works much like Excel. Power Fx in Power Apps supports all prebuilt and custom AI models. It also supports models built on the Microsoft Azure Machine Learning platform.

To learn more, go to [Use Power Fx in AI Builder models in Power Apps \(preview\)](#).

Property name changes in AI Builder components for canvas apps

With the [AI Builder component improvements](#) released in April 2020, some of the property names in the AI Builder components for canvas apps have changed. In most cases, your existing apps will be automatically updated to use the new property names without any action required from you. However, in cases where the automatic updates to your app are unsuccessful, here are the property name changes you must make in your apps:

- For the [form processor](#) component:

Previous property name	New property name
{Control Name}.FormContent.Fields	{Control Name}.Fields
{Control Name}.FormContent.Tables	{Control Name}.Tables

- For the [text recognizer](#) component:

Previous property name	New property name
{Control Name}.SelectedText	{Control Name}.Selected.Text
{Control Name}.OcrObjects.text	{Control Name}.Results.Text

- For the [object detector](#) component:

Previous property name	New property name
{Control Name}.VisionObjects.id	{Control Name}.GroupedResults.TagId
{Control Name}.VisionObjects.displayName	{Control Name}.GroupedResultsTagName
{Control Name}.VisionObjects.count	{Control Name}.GroupedResults.ObjectCount

If your app uses a **data table** component to display results from the **object detector** component, and you don't see the content after this update:

1. Remove the **data table** component from your app.
2. Add it again.
3. Reset the item's property to correctly display the content from the **object detector** component.

See also

- [Feature availability by region](#)
- [AI Builder in Power Automate](#)
- [What are canvas apps?](#)
- [What are model-driven apps?](#)
- [Training: Use AI Builder models in Power Apps \(module\)](#)
- [Training: Improve business performance with AI Builder \(learning path\)](#)

Use your Azure OpenAI Service model in Power Apps (preview)

Article • 06/06/2023

[This topic is pre-release documentation and is subject to change.]

GPT (Generative Pre-trained Transformer) models are a type of natural language processing model. GPT models are trained on a large body of content to generate human-like text from a prompt. Use them in your apps to interactively fill in forms or questionnaires, generate reports and summaries from a dataset, create automated chatbot conversations, and more. GPT models are especially helpful in generating responses for customer service teams that need to quickly reply to customer inquiries.

ⓘ Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.
- [View our preview terms ↗](#).
- This capability may not be available in your region yet.
- This capability may be subject to usage limits or capacity throttling.

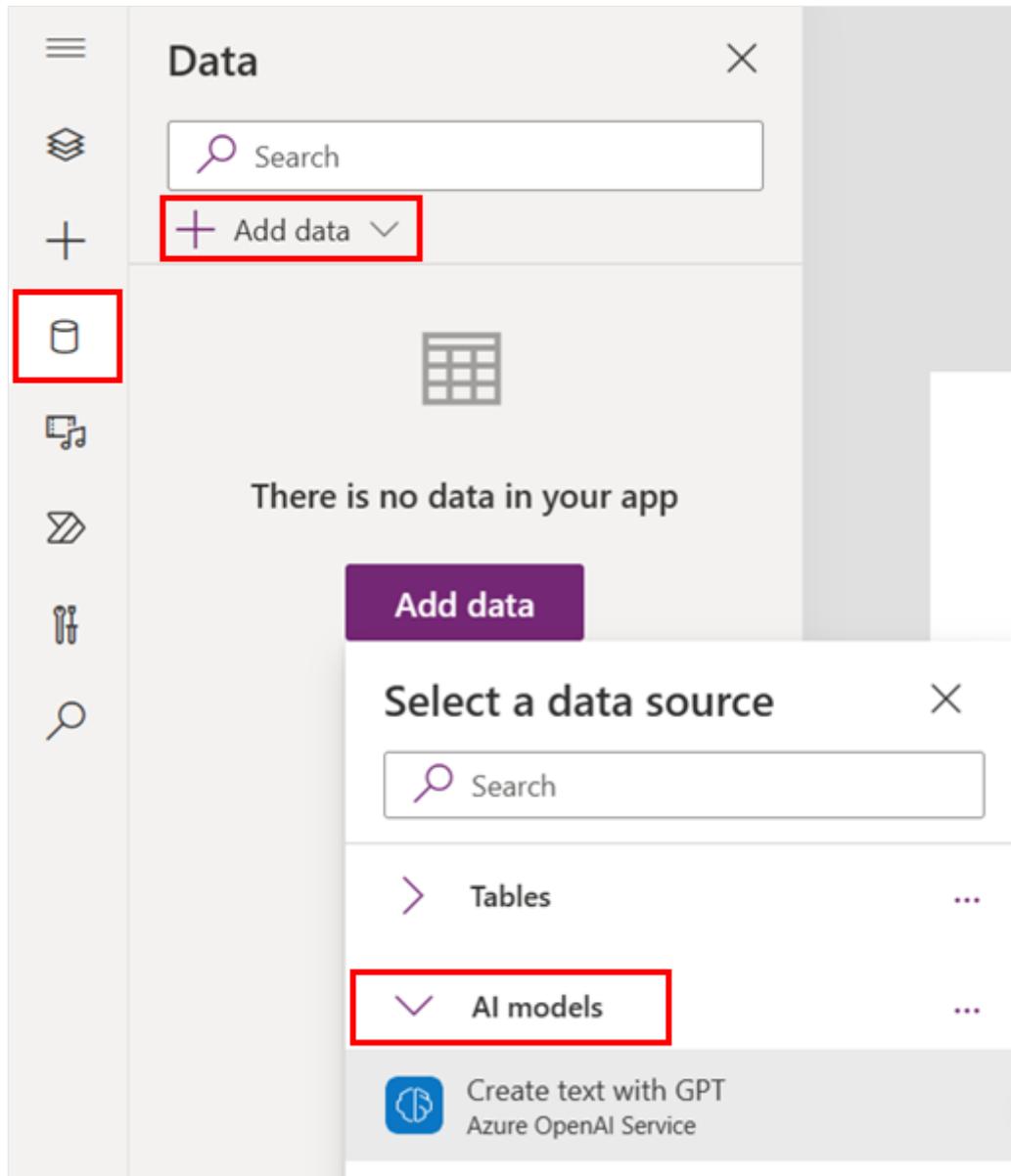
Add an AI model as a data source

The following example creates a simple app that answers a question entered in a text box.

1. Sign in to [Power Apps ↗](#).
2. On the left navigation pane, select **Apps**.
3. On the menu at the top, select **+ New app > Canvas**.
4. Enter a name for the app and choose between **Tablet** and **Phone** for the format of the app.

5. Select **Create**.

6. On the list of icons to the left of the Tree view, select **Data > Add data > AI models**.



7. Select one or more models to add.

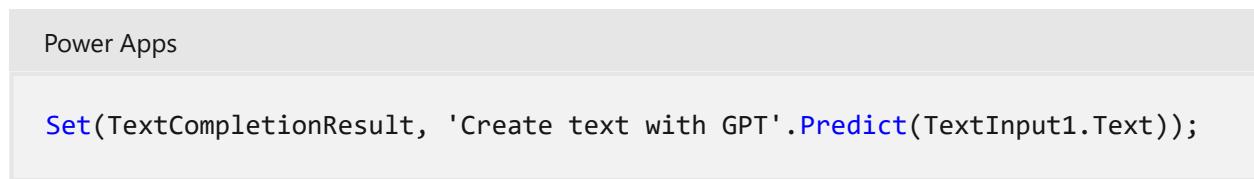
If you don't see your model in the list, you might not have permission to use it in Power Apps. Contact your administrator to resolve this problem.

Bind the model prediction to a control

Next, bind the model prediction with a control or an event to trigger the model response. The screenshot in this section shows a simple application to answer any question specified in the instruction text box.

To bind the **Create text with GPT** model to a control, identify an event of the control that you want to invoke the model prediction. In this case, we're binding the model to the **Generate Text** button and the **OnSelect** event on the button. The result is that whenever the button is selected, the **OnSelect** event is triggered, which triggers the Power Fx function mentioned here.

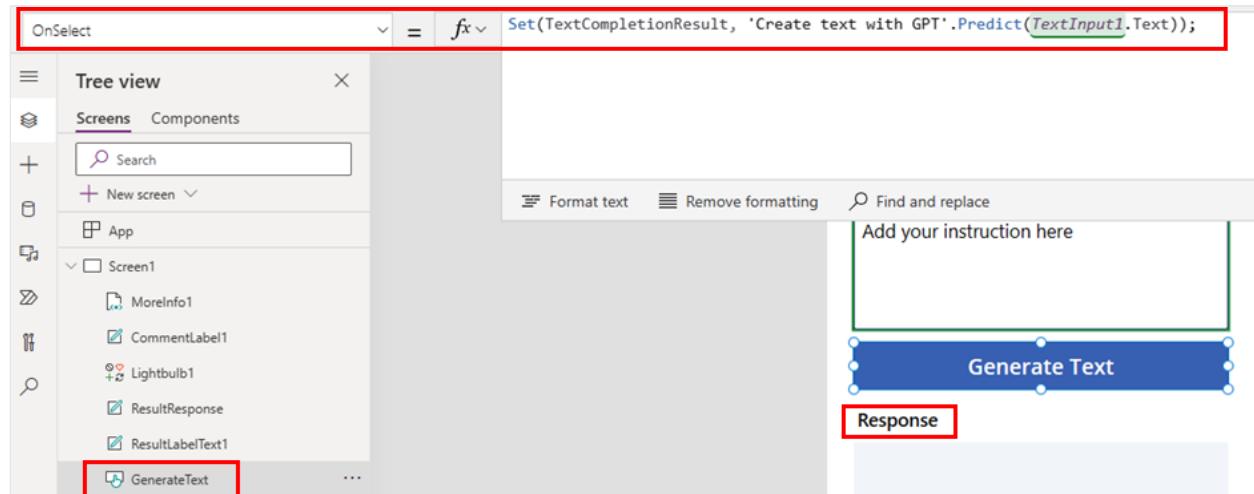
1. On the list of icons to the left, select **Tree view**.
2. Above the **Tree view** heading, select **OnSelect** in the dropdown menu.
3. On the **Screens** tab, select **GenerateText**.
4. Notice the Power Fx function:



```
Power Apps

Set(TextCompletionResult, 'Create text with GPT'.Predict(TextInput1.Text));
```

The `.Predict()` on this model accepts a string as a parameter and returns the generated text as a response text. In the following example, we're passing the instruction from the text box as a prompt to the **Create text with GPT** model, and the response from the model appears in the **Response** label.



Congratulations! You've created an app that uses an AI Builder Create text with GPT capability. On the top of the screen, select **Save** to save all the changes in the app, and then select **Play** to test the application.

Parameters

For a list of the parameters used in Azure OpenAI Service, go to [input parameters](#) and [output parameters](#) in Power Automate.

See also

[Azure OpenAI Service model overview \(preview\)](#) [How text generation in Azure OpenAI Service works \(preview\)](#) [Use your Azure OpenAI Service model in Power Automate \(preview\)](#)

Use your prediction model in Power Apps

Article • 03/05/2022

This topic takes you through the process of creating a model-driven app that can sort your rows by the probability value in the AI model output.

ⓘ Note

For more information about model-driven apps, see [What are model-driven apps in Power Apps?](#)

1. Sign in to [Power Apps](#).
2. Select **Apps** from the pane on the left.
3. Select **New app > Model-driven** from the menu at the top.
4. On the **Create a New App** screen, complete the fields, and then select **Done**.
5. On the **App Designer** screen, do the following:
 - a. Create a view of your table that contains the predicted columns from the prediction model output.
 - b. Add the view to the site map.
6. On the **Components** tab, select **Entities**, select your table, and then select **Back**.
7. Select **Save** at the top of the screen so that you don't lose your progress when you create a view.
8. Under **Entity view**, select **Views**, and then on the **Components** tab, select **Create New**.
9. On the **Components** tab, select **Column Attributes – Primary Entity**, and then select the columns generated by prediction output.
10. Select **Save And Close**, give your view a name, and then select the view you created.
11. Select **Save** again.
12. Select the pencil icon next to **Site Map** and rename **New Group** appropriately.

13. Select **New Subarea** and select **Entity** as the **Type**, and then select your table from the **Entity** drop-down menu.
14. Select **Save And Close**, and then select **Publish**.
15. Select **Play**. Your output should look something like the following image.

The screenshot shows the Dynamics 365 interface with the title bar "Dynamics 365". The left sidebar has sections for Home, Recent, Pinned, and Predictions, with "Adult Census Incomes" selected. The main area displays a list titled "Adult Census Income Predictions" with columns: Id, Income - Predic..., Income - >50K ..., and Income - Explan.... The list contains records with Id values 5, 10, 18, 19, 20, and 17, all showing three dashes in the other columns. A search bar at the top right says "Search for records". At the bottom, it shows "ABC 1 - 50 of 99 (0 selected)" and navigation buttons for Page 1.

Id	Income - Predic...	Income - >50K ...	Income - Explan...
5	---	---	---
10	---	---	---
18	---	---	---
19	---	---	---
20	---	---	---
17	---	---	---

Congratulations, you've created your first model-driven app that uses an AI Builder prediction model.

Use the form processor component in Power Apps

Article • 01/05/2023

You can add the AI Builder form processor component to your screen in your canvas apps. This component takes a photo or loads your image. Then it extracts text based on your trained AI model. If it detects a form that the AI model is trained for, the form processor extracts the field values and identifies them by using rectangles.

ⓘ Note

For more information about creating a new document processing model, see [Get started with document processing in AI Builder](#)

For more information about canvas apps, see [What are canvas apps in Power Apps?](#).

Prerequisites

This component requires a published AI Builder document processing model. Then that model must be bound to the component by using the AI model property in the properties panel.

When you add the component to the screen, it automatically opens the AI models pane. There, you select a model that has been published in your environment. The component is initialized after an AI model is bound to it.

Key properties

- **ModelId** ("AI model" in the properties panel): AI model information to which the component is bound.
- **OriginalImage**: The original image before processing.
- **Fields**: The extracted fields by the AI model.
- **Tables**: The extracted tables by the AI model.
- **Results**: Contains all the outputs returned by the AI Model along with advanced properties. For each value returned by the AI model, you can access:

- **BoundingBox** The coordinates on the form for the extracted field.
- **Confidence** How confident the model is in its prediction of the extracted text.
- **PageNumber** The number of the page where the extracted text is located.
- **Value** The extracted text value.

① Note

Some of these property names changed with the April 2020 updates. If you aren't seeing these properties in your app, you'll have to manually update the property names. More information: [Property name changes in AI Builder components for canvas apps](#)

Additional properties

- **Text:** Text that appears on the button that activates the form processor.
- **ImageDisplayed (Show image in the properties panel):** Determines whether the component displays the image. When set to **On**, rectangles are displayed around field values detected in the image.

Three colors can be used to draw the rectangle depending on the confidence level:

 - **Red:** Confidence level is between 0 percent and 39 percent.
 - **Orange:** Confidence level is between 40 percent and 59 percent.
 - **Blue:** Confidence level is between 60 percent and 100 percent.
- **ShowConfidence (Show confidence in the properties panel):** Determines whether the component displays confidence levels along with the rectangles in the image.
- **DisplayMode:**
 - **Edit:** Allows user input.
 - **View:** Only displays data.
 - **Disabled:** Is disabled.
- **Height:** The height of the component.
- **Visible:** Whether the component appears or is hidden.
- **Width:** The width of the component.

- X: The distance between the left edge of the component and the left edge of its parent container or screen.
- Y: The distance between the top edge of the component and the top edge of the parent container or screen.

Additional design properties are available in the **Advanced** panel.

Accessibility guidelines

These [guidelines](#) for the Power Apps button control also apply to the form processor component.

See also

- [Overview of the object detection model](#)
- [Core properties in Power Apps](#)
- [Training: Process custom documents with AI Builder \(module\)](#)

Use the object detection component in Power Apps

Article • 01/05/2023

The object detection component takes a photo or loads an image file to do an object detection scan. On a mobile device, the user chooses between taking a photo or selecting one already available in the device user interface. When an image is selected, the component automatically scans it to identify objects.

ⓘ Note

For information about canvas apps, see [What are canvas apps in Power Apps?](#).

Prerequisites

To start, you need a trained and published AI Builder object detection model. You bind that model to the component using **AI Model** in the **Properties** panel. When you add it to the screen, the component automatically opens the AI models pane. There you select a model from the models published in that environment.

The component is initialized after an AI model is bound to it.

Properties

Key properties

- **ModelId** (AI model in the properties panel): AI model information to which the component is bound.
- **OriginalImage**: The original image before processing.
- **GroupedResults**: The details of detected objects. For each object, these properties are available:
 - **TagId** The ID of the object detected.
 - **TagName** The name of the object detected.
 - **ObjectCount** The number of times the object is present on the image.

- **Results**: Contains all the outputs returned by the model.
 - **BoundingBox** The coordinates on the image of the object detected.
 - **Confidence** How confident the model is in its prediction of the object detected.
 - **TagId** The ID of the object detected.
 - **TagName** The name of the object detected.

 **Note**

Some of these property names changed with the April 2020 updates. If you aren't seeing these properties in your app, you'll have to manually update the property names. More information: [Property name changes in AI Builder components for canvas apps](#)

Additional properties

- **Text**: Text that appears on the button that activates the object detector.
- **ImageDisplayed (Show image in the properties panel)**: Determines whether the component displays the image.
- **DisplayMode**:
 - **Edit**: Allows user input.
 - **View**: Only displays data.
 - **Disabled** is disabled.
- **Height**: The height of the component.
- **Visible**: Whether the component appears or is hidden.
- **Width**: The width of the component.
- **X**: The distance between the left edge of the component and the left edge of its parent container or screen.
- **Y**: The distance between the top edge of the component and the top edge of the parent container or screen.

Additional design properties are available in the **Advanced** panel.

Accessibility guidelines

These [guidelines](#) for the Power Apps button control also apply to the form processor component.

See also

- [Overview of the object detection model](#)
- [Use the object detection model in Power Automate](#)
- [Core properties in Power Apps](#)
- [Training: Detect objects with AI Builder \(module\)](#)

Use the business card reader component in a canvas app in Power Apps

Article • 01/05/2023

Use the AI Builder business card reader component to detect business cards and extract their information. You can take photos directly in the component or load images that you've taken. Data is extracted and identified by using the properties listed below.

For information about canvas apps, see [What are canvas apps in Power Apps?](#)

Licensing requirements

AI Builder is licensed as an add-on to your Power Apps or Power Automate licenses. For information about license capacity, pricing, and restrictions, see [AI Builder licensing](#).

Role requirements

You need the Basic User role to use the business card reader.

Key properties

If a business card is detected, the business card reader will extract information that it finds based on the following properties.

Property	Definition
AddressCity	City
AddressCountry	Country
AddressPostalCode	Postal code
AddressPostOfficeBox	Post office box
AddressState	State address
AddressStreet	Street address
BusinessPhone	The first phone or fax number

Property	Definition
CleanedImage	The image after processing, where the business card appears cropped and enhanced from the original image
CompanyName	Company name
Department	Organization department
Email	The contact's email address, if any
Fax	The third phone or fax number
FirstName	The contact's first name
FullAddress	The contact's full address
FullName	The contact's full name
JobTitle	The contact's job title
LastName	The contact's last name
MobilePhone	Second phone or fax number
OriginalImage	The original image, before processing
Website	Website

Additional properties

Name	Definition
Text	Text that appears on the button that activates the business card reader
ImageDisplayed (Show image in the properties panel)	Whether the component displays the image
DisplayMode (Edit)	Allows user input
DisplayMode (View)	Only displays data
DisplayMode (Disabled)	Is disabled
Height	The height of the component
Visible	Whether the component appears or is hidden
Width	The width of the component

Name	Definition
X	The distance between the left edge of the component and the left edge of its parent container or screen
Y	The distance between the top edge of the component and the top edge of the parent container or screen

Additional design properties are available in the **Advanced** panel.

Accessibility guidelines

These [guidelines](#) for the Power Apps button control also apply to the business card reader component.

See also

- [Core properties in Power Apps](#)
- [Training: Extract information from business cards with AI Builder \(module\)](#)

Use the business card reader component in model-driven apps in Power Apps

Article • 01/05/2023

Use the AI Builder business card reader component to detect business cards and extract their information. You can take photos directly in the component or load images that you've taken. Data is extracted and identified by using the properties listed below.

For information about model-driven apps, see [What are model-driven apps in Power Apps?](#)

Licensing requirements

AI Builder is licensed as an add-on to your Power Apps or Power Automate licenses. For information about license capacity, pricing, and restrictions, see [AI Builder licensing](#).

Role requirements

You need the Basic User role to use the business card reader.

Add the business card reader

Add the business card reader to a contact or lead form by using the form editor, and bind it to a placeholder **SingleLine.Text** or **Multiple** field.

1. Select the placeholder field, and then select **Properties**.
2. Select the **Controls** tab.
3. Select **Add Control**.
4. Select **AI Builder Business Card Control**.

Field Properties

Modify this field's properties.

Display Formatting Details Events Business Rules **Controls**

Add Control

Select a custom control from the field.

AI Builder Business Card control

Attribute Picker Control

Auto-Complete

Barcode Scanner

AI Builder Business Card control

Modes:

Types: SingleLine.Text, Multiple

Use this AI Builder control to automatically extract contact information from images.



Add

OK

Cancel

Configure the business card reader

After you select the platform where the business card reader should appear (web, phone, or tablet), you can bind the component properties that you need.

Field Properties

? X

Modify this field's properties.

Display Formatting Details Events Business Rules Controls

Control

Web Phone Tablet

Text Box (default)

AI Builder Business Card control **X**

[Add Control...](#)

AI Builder Business Card control

Full Name	fullname (SingleLine.Text)	
First Name	firstname (SingleLine.Text)	
Last Name	lastname (SingleLine.Text)	
Job Title	jobtitle (SingleLine.Text)	
Mobile Phone	mobilephone (SingleLine.Phone)	
Business Phone	telephone1 (SingleLine.Phone)	

Full Name

Compatible types: SingleLine.Text

Extracted Full Name

Hide Default Control

OK

Cancel

ⓘ Note

The **Company Name** property can't be bound to a field of type **Lookup.Customer**. The only supported type is **SingleLine.Text**. More information: [Add code components to a column or table in model-driven apps](#)

Key properties

If a business card is detected, the business card reader will try to extract information that it finds based on the following properties.

Property	Definition	Type
Full Name	The contact's full name	SingleLine.Text
First Name	The contact's first name	SingleLine.Text
Last Name	The contact's last name	SingleLine.Text
Job Title	The contact's job title	SingleLine.Text
Mobile Phone	The mobile phone number detected	SingleLine.Phone
Business Phone	The business phone number detected	SingleLine.Phone
Fax	The fax number detected	SingleLine.Phone
Email	The contact email found in the business card, if any	SingleLine.Email
Company Name	The company name	SingleLine.Text
Website	The website detected	SingleLine.URL
Department	The organization department found	SingleLine.Text
Full Address	The contact full address	SingleLine.TextArea, Multiple
Address Street	The street address detected	SingleLine.Text
Address City	The city address detected	SingleLine.Text
Address State	The state address detected	SingleLine.Text
Address PostalCode	The postal code address detected	SingleLine.Text
Address Country	The country address detected	SingleLine.Text
Address Post Office Box	The post office box address detected	SingleLine.Text
Cleaned Image	The image after processing where the business card appears cropped and enhanced from the original image.	Multiple

Customization properties

The following properties are available for advanced customization:

Property	Definition
Default Image	To replace the default placeholder image with a different one. The image (JPEG or PNG) should be encoded with base-64 digits.
Text	To override the default button text.

See also

[Training: Extract information from business cards with AI Builder \(module\)](#)

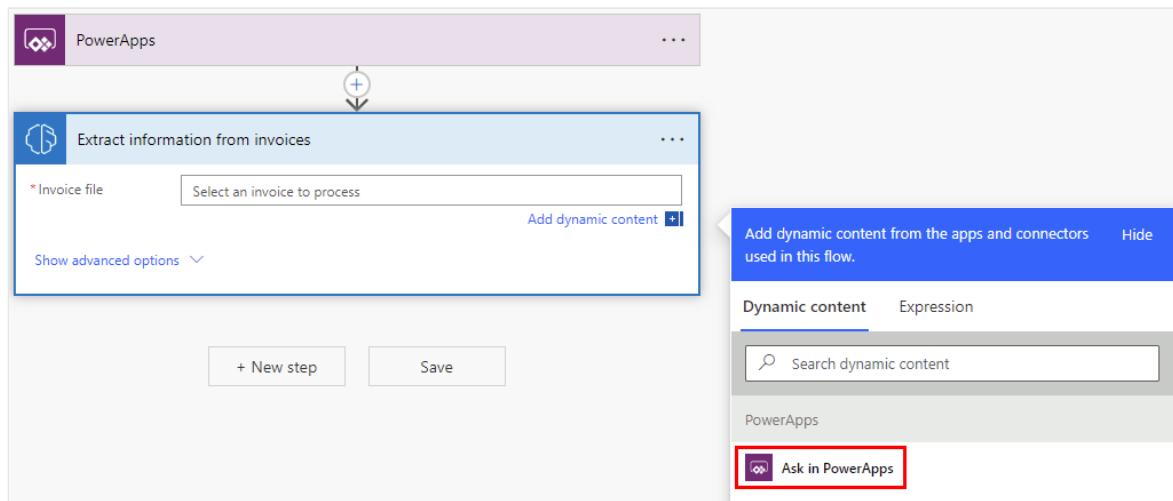
Use invoice processing in Power Apps

Article • 01/05/2023

Currently, there's no invoice processing control available for canvas apps in Power Apps. But you can create an app that calls a Power Automate flow that uses the invoice processing prebuilt AI model, and then return the results back to your app. The following example shows how to do this:

Build your flow

1. Sign in to [Power Automate](#).
2. Make sure you are in the same Power Platform environment where you'll have your canvas app. Check the environment on the top right of the page.
3. On the pane to the left, select **Flows**.
4. Select **New flow > Instant cloud flow**.
5. Name your flow "Invoice processing flow", and then select **Power Apps** under **Choose how to trigger this flow**.
6. Select **Create**.
7. Select **+ New step > AI Builder**, and then select **Extract information from invoices** in the **Actions** list.
8. Select the **Invoice file** input and then select **Ask in Power Apps** in the **Dynamic content** list.

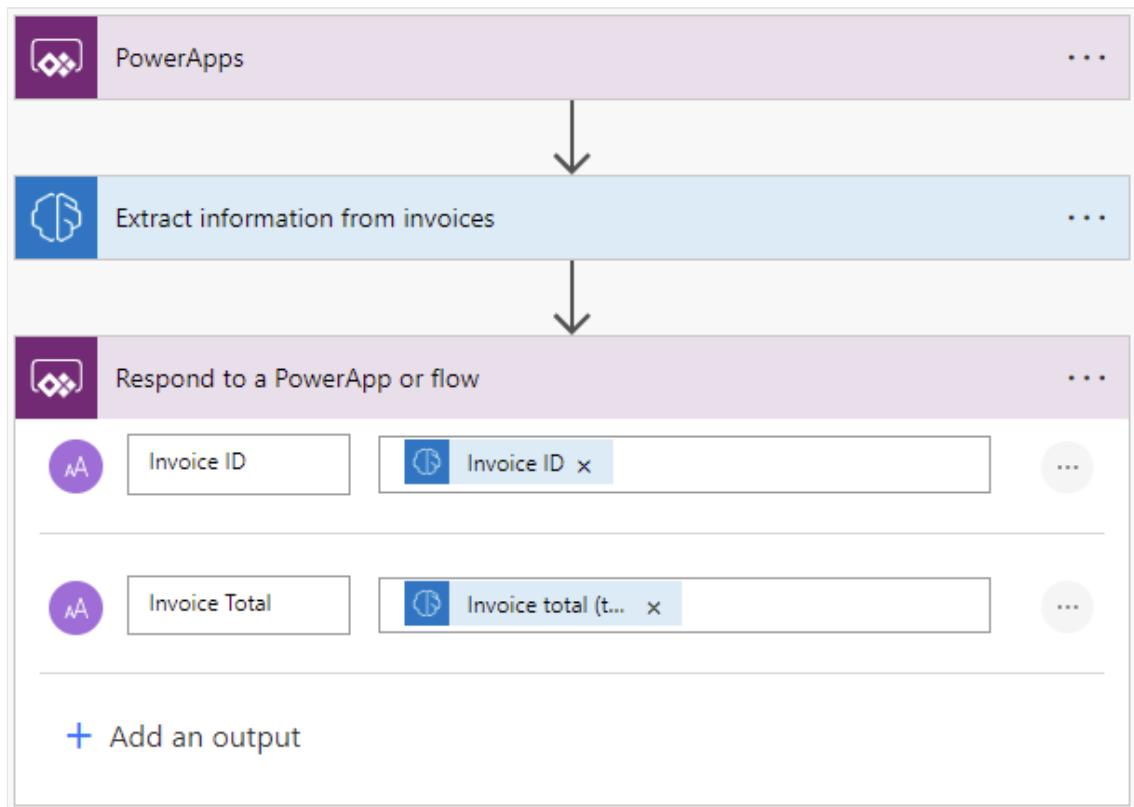


9. Select **+ New step**, search for *respond to a powerapp*, and then select **Respond to a PowerApp or flow** in the **Actions** list.

For this example we're going to add two text outputs: *Invoice ID* and *Invoice Total*. You might want to add more or different outputs based on the extracted invoice fields you want to send back to your canvas app.

To add the inputs:

- a. Select **+Add an output > Text**.
- b. Replace **Enter title** with **Invoice ID**.
- c. Select the new **Invoice ID** input, and then select **Invoice ID** from the **Dynamic content** list.
- d. Select **+ Add an input > Text**.
- e. Replace **Enter title** with **Invoice Total**.
- f. Select the new **Invoice Total** input, and then select **Invoice total (text)** from the **Dynamic content** list.

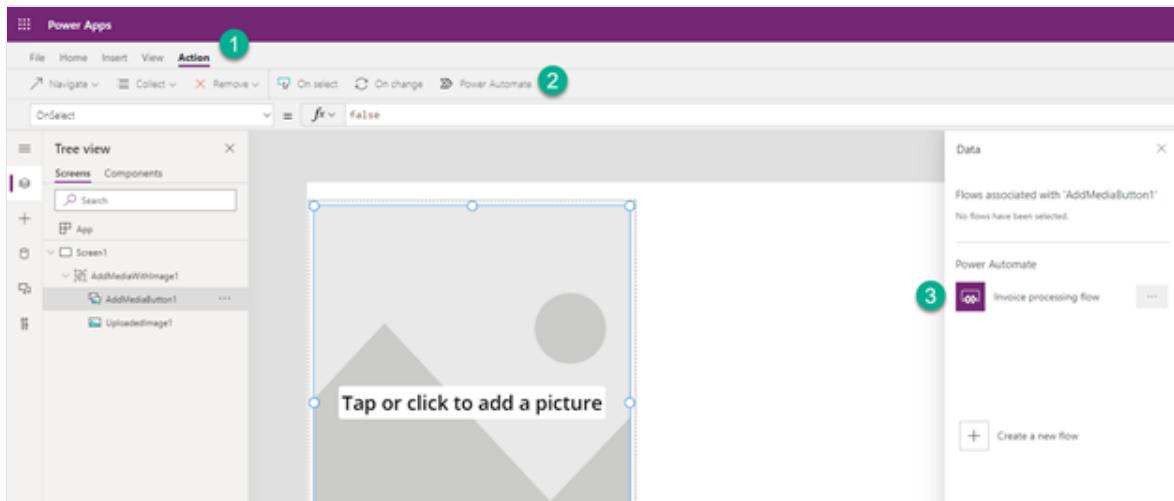


10. Save your flow.

Build your canvas app

1. Sign in to [Power Apps](#).

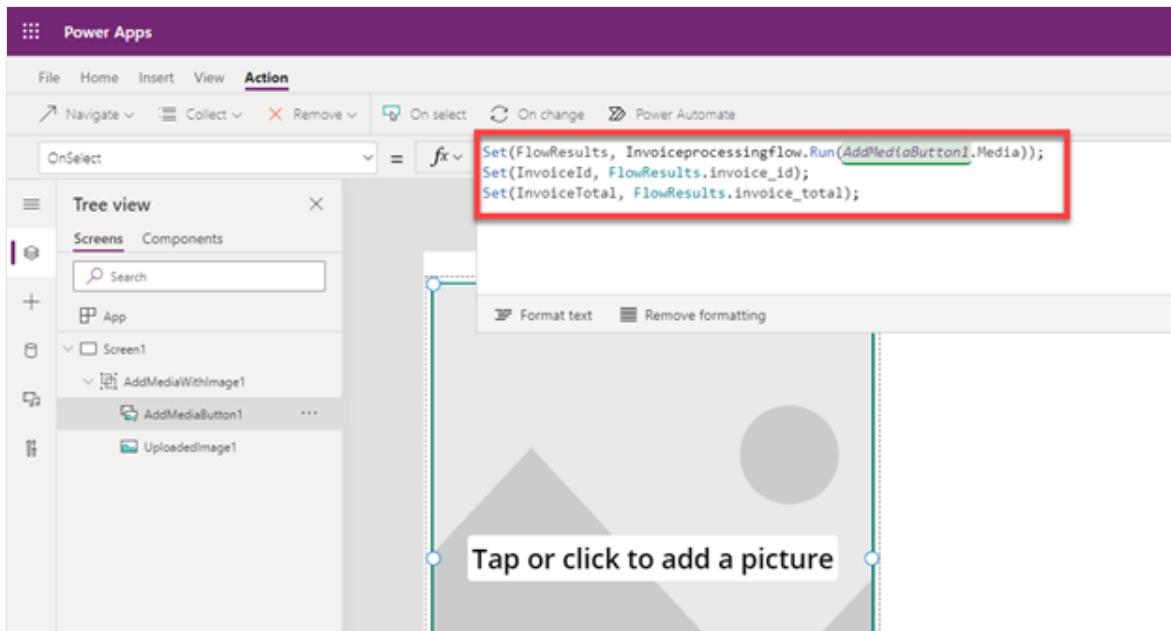
2. Make sure you are on the same Microsoft Power Platform environment where you created the flow on the top right of the page.
3. Select **+Create** in the left-side navigation pane.
4. Select the **Canvas app from blank tile**.
5. Name your app, select either **Tablet** or **Phone** format, and then select **Create**.
6. In the app editor, select **Insert > Media > Add picture** to insert a control into your app where users can upload a picture from a device or camera.
7. Select the **AddMediaButton1** control on the left.
8. On the formula bar on the top, select the **OnSelect** event.
9. Select the **Action** menu and select **Power Automate**. Choose the flow that we created in the previous steps. If you don't see the flow, make sure you are on the same Power Platform environment as where you created the flow, you can check the environment on the top right of the page.



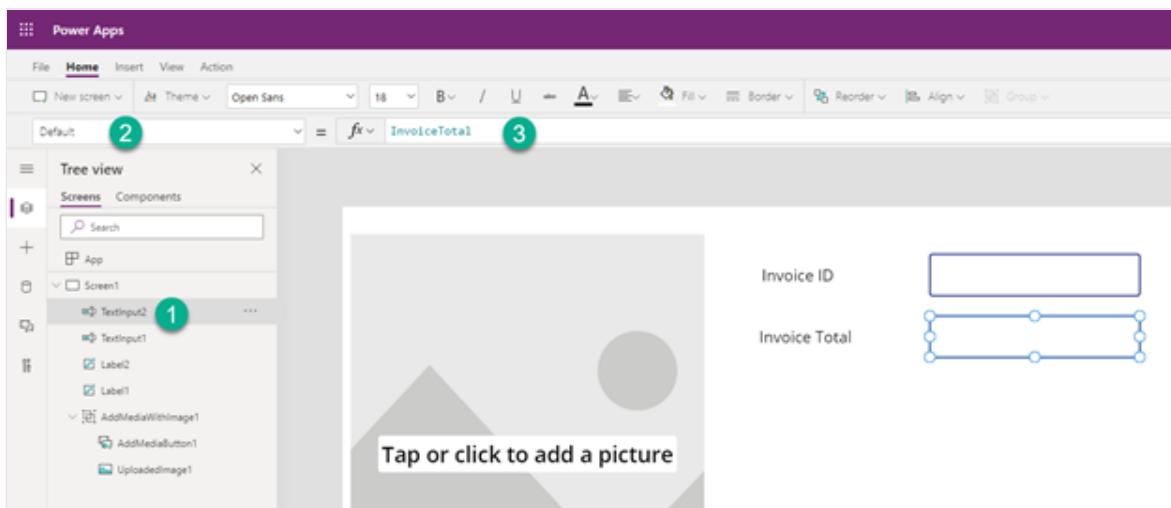
10. Enter the following formula on the **OnSelect** event for the **AddMediaButton1** control. This formula tells the app to call the flow we've built once a new image is uploaded and saves the results we received from the flow in variables.

The name of the variables and name of your flow might be different depending on how you configured it when building the flow:

```
Set(FlowResults, Invoiceprocessingflow.Run/AddMediaButton1.Media));
Set(InvoiceId, FlowResults.invoice_id);
Set(InvoiceTotal, FlowResults.invoice_total);
```



11. Now let's add two labels and two text inputs to display the results we'll get from the flow. Change the Default property for each text input to take the values from the variables we've defined in the previous step.



12. The app is now ready to run! Select the Play icon on the top right to test it.



Invoice ID

1975

Invoice Total

\$457.87

See also

[Training: Extract invoice data with AI Builder's prebuilt model \(module\)](#)

Use the receipt processor component in Power Apps

Article • 01/05/2023

[This topic is pre-release documentation and is subject to change.]

The AI Builder receipt processor component scans and extracts information from receipts. You can take photos directly within the component or load images that have already taken. The data is recognized and extracted using the properties below.

For more information about canvas apps, see [What are canvas apps in Power Apps?](#)

ⓘ Important

- This is a preview feature.
- Preview features aren't meant for production use and may have restricted functionality. These features are available before an official release so that customers can get early access and provide feedback.

Requirements

The receipt processor component works best with sales receipts, those commonly used by restaurants, gas stations, and retailers, and others. Both print and handwritten text can be detected.

Only English receipts from the United States are currently supported.

In order to get the best results, provide one clear photo or scan per receipt.

- The image format must be JPEG, PNG, or PDF.
- The file size must be less than 20 MB.
- The image dimensions must be between 50 x 50 pixels and 10,000 x 10,000 pixels.
- PDF dimensions must be at most 17 x 17 inches, which is the equivalent of the legal or A3 paper sizes or smaller.
- For PDF documents, only the first 200 pages are processed.

Receipt properties

Property	Definition
MerchantName	Merchant name
MerchantAddress	Merchant address
MerchantPhone	Merchant phone number
TransactionDate	Transaction date
TransactionTime	Transaction time
PurchasedItems	<p>The list of purchased items</p> <ul style="list-style-type: none"> • Name: Name of the purchased item • Price: Price of the purchased item • Quantity: Quantity of the purchased item • TotalPrice: Total price of the purchased item
Subtotal	Subtotal
Tax	Tax
Tip	Tip
Total	Total

ⓘ Note

Receipt values are returned as strings. To manipulate them as numbers, you can use the **Value** function. To manipulate them as dates or times, you can use the **DateValue** and **TimeValue** functions. You can also specify the language of the text with a language tag, such as "en-US".

Additional properties

Property	Definition
DetectedFields	<p>Additional information for each of the receipt properties</p> <ul style="list-style-type: none"> • BoundingBox: The coordinates of the field • Confidence: How confident the model is in the detection of the field • PageNumber: Which page the field is found on • Value: The value of the field

Property	Definition
DetectedText	The list of all recognized lines of text on the receipt <ul style="list-style-type: none"> • BoundingBox: The coordinates of the line of text • PageNumber: Which page the line of text is found on • Value: The line of text
OriginalImage	The original image before processing
DisplayMode	<ul style="list-style-type: none"> • Edit: Allows user input • View: Only displays data • Disabled: Does not allow user input
Height	The height of the component
ImageDisplayed	Whether the component displays the image or not
ShowBoundingBoxes	Whether the component displays the bounding boxes or not
Text	The text that appears on the button that activates the receipt processor
Visible	Whether the component appears or is hidden
Width	The width of the component
X	The distance between the left edge of the component and the left edge of the parent container or screen
Y	The distance between the top edge of the component and the top edge of the parent container or screen

Additional design properties are available in the **Advanced** panel.

Accessibility guidelines

These [guidelines](#) for the Power Apps button control also apply to the text recognizer component.

See also

- [Receipt processing overview](#)
- [Core properties in Power Apps](#)
- [Training: Process receipts with AI Builder \(module\)](#)

Use the text recognizer component in Power Apps

Article • 01/05/2023

Create a canvas app and add the text recognizer AI Builder component to your screen. This component takes a photo or loads an image from the local device, and then processes it to detect and extract text based on the text recognition prebuilt model. If it detects text in the image, the component outputs the text and identifies the instances by showing a rectangle for each instance in the image.

ⓘ Note

For information about canvas apps, see [What are canvas apps in Power Apps?](#)

Key properties

- **OriginalImage**: The original image before processing.
- **Results**: The list of detected text lines. For each object, these properties of extracted form fields and tables are available. At component initialization (AI model binding step), the potential fields and tables that can be extracted by the models are populated. These include:
 - **BoundingBox**: The coordinates for the detected text line.
 - **PageNumber**: The number of the page where the detected text line is located.
 - **Text**: The detected text line.
- **Selected**: The detected box selected by the user on the control.
 - **BoundingBox**: The coordinates for the detected text line selected by the user.
 - **PageNumber**: The number of the page where the detected text line selected by the user is located.
 - **Text**: The detected text line selected by the user.

ⓘ Note

Some of these property names changed with the April 2020 updates. If you aren't seeing these properties in your app, you'll have to manually update the property names. More information: [Property name changes in AI Builder components for canvas apps](#)

Additional properties

- **Text:** Text that appears on the button that activates the text recognizer.
- **ImageDisplayed (Show image in the properties panel):** Determines whether the component displays the image. When set to **On**, rectangles are displayed around column values detected in the image.
- **DisplayMode:**
 - **Edit:** Allows user input.
 - **View:** Only displays data.
 - **Disabled** is disabled.
- **Height:** The height of the component.
- **Visible:** Whether the component appears or is hidden.
- **Width:** The width of the component.
- **X:** The distance between the left edge of the component and the left edge of its parent container or screen.
- **Y:** The distance between the top edge of the component and the top edge of the parent container or screen.

Additional design properties are available in the **Advanced** panel.

Accessibility guidelines

These [guidelines](#) for the Power Apps button control also apply to the text recognizer component.

See also

- [Text recognition model](#)

- Core properties in Power Apps
- Training: Recognize text with AI Builder (module)

Use AI Builder models in Power Apps

Article • 02/24/2023

With the use of [Power Fx](#), the open-source low-code formulas, you can add more powerful and flexible integrations of AI models into your Power App. AI model prediction formulas can be integrated with any controls in canvas app. For example, you can detect the language of text in a text input control and output the results to a label control as can be seen in the [Use a model with controls](#) section below.

Requirements

To use Power Fx in AI Builder models, you must have:

- Access to a [Microsoft Power Platform environment with a database](#).
- AI Builder license (trial or paid). To learn more, go to [AI Builder licensing](#).

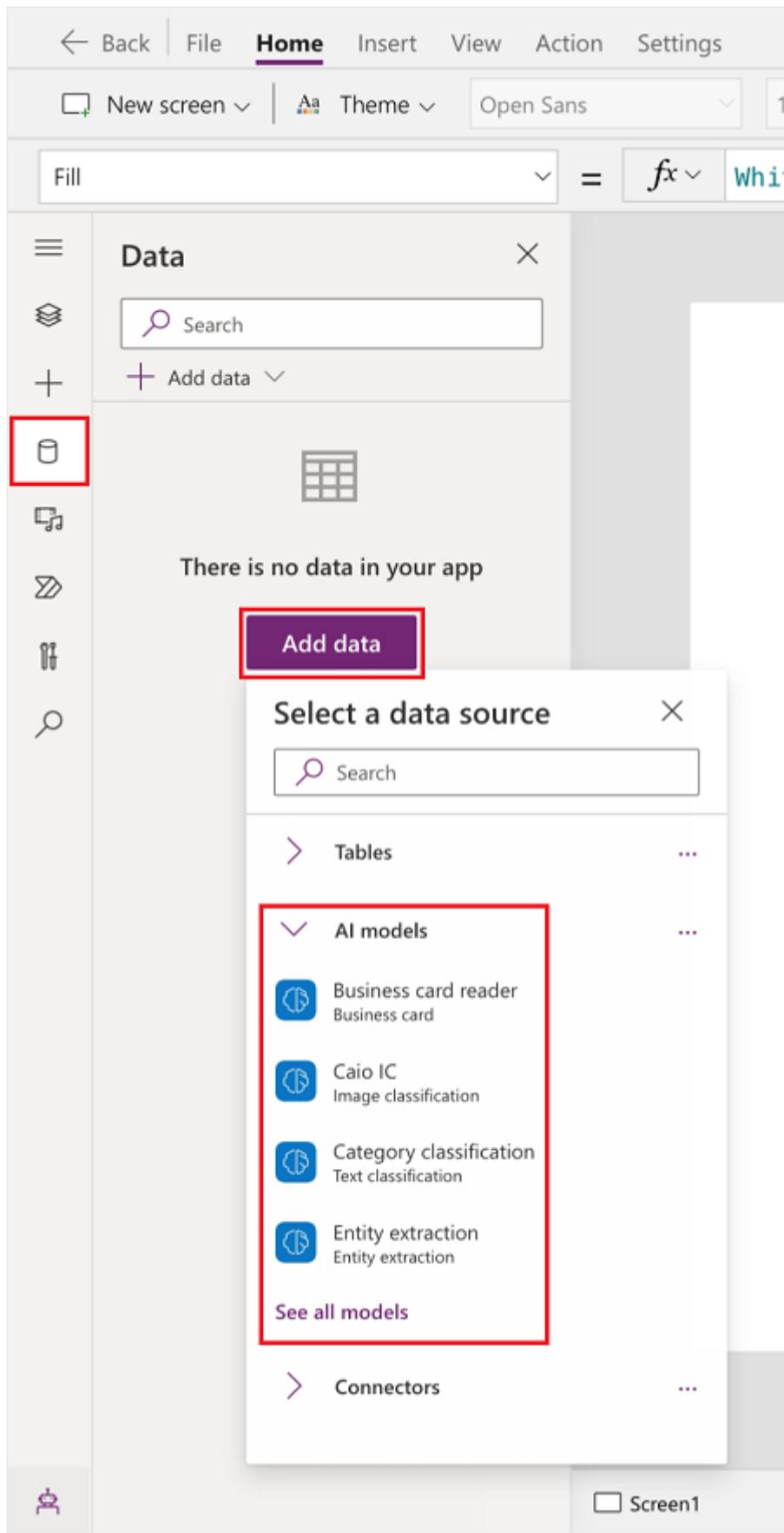
Select a model in canvas apps

To consume an AI model with Power Fx, you'll need to create a canvas app, choose a control, and assign expressions to control properties.

Note

For a list of AI Builder models you can consume, see [AI models and business scenarios](#). You can also consume models built in Microsoft Azure Machine Learning with the [bring your own model](#) feature.

1. Create an app. More information: [Create a blank canvas app from scratch](#).
2. Select **Data > Add data > AI models**.



3. Select one or more models to add.

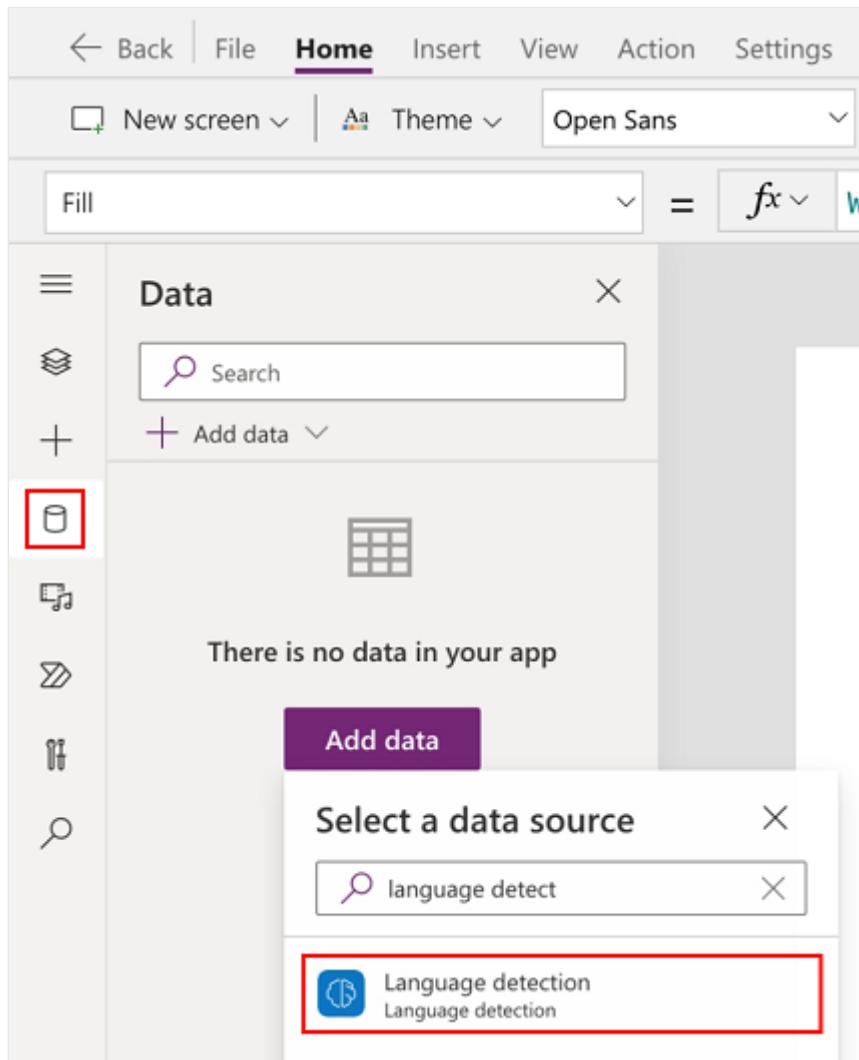
If you don't see your model in the list, you might not have permissions to use it in Power Apps. Contact your administrator to resolve this problem.

Use a model with controls

Now that you've added the AI model to your canvas app, let's see how to call an AI Builder model from a control.

In the following example, we'll build an app that can detect the language entered by a user in the app.

1. Create an app. More information: [Create a blank canvas app from scratch](#).
2. Select **Data > Add data > AI models**.
3. Search for, and select **Language detection** AI model.



ⓘ Note

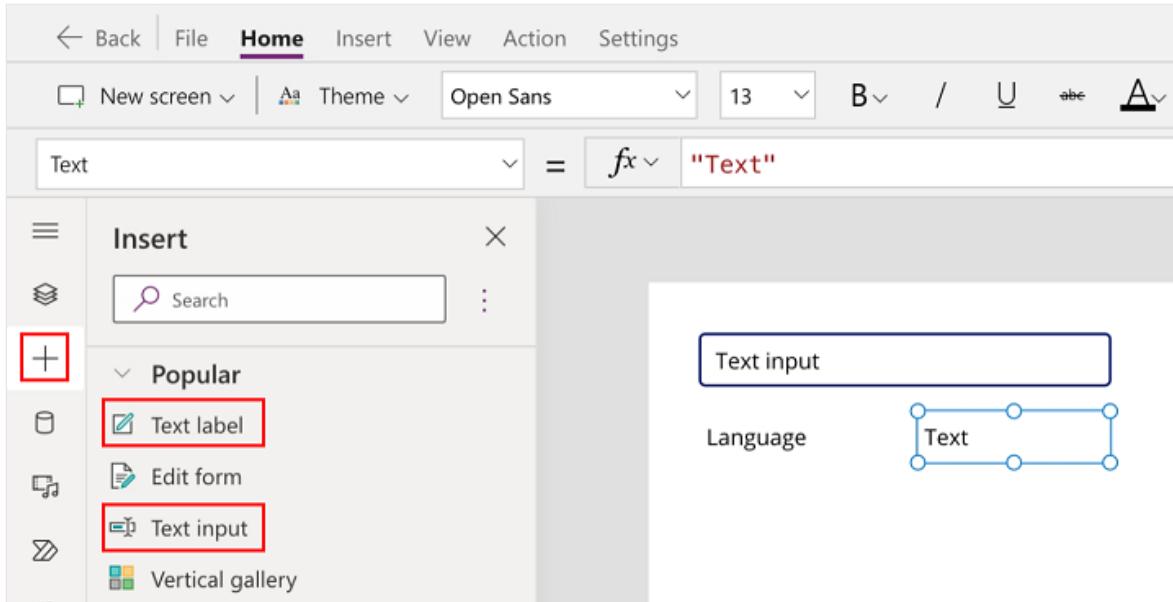
You'll have to manually add the model to the app again in the new environment upon moving the app across environments.

4. Select **+** from the left-pane, and then select **Text input** control.

5. Repeat the previous step to add a **Text label** control.

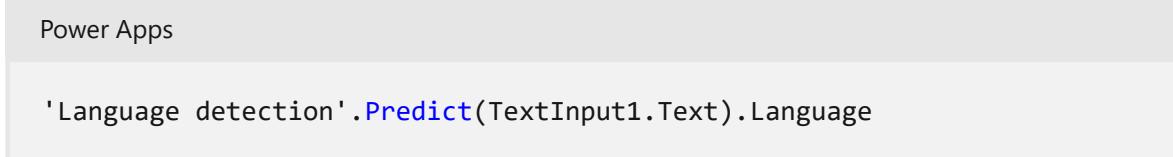
6. Rename the text label to **Language**.

7. Add another text label next to the "Language" label.

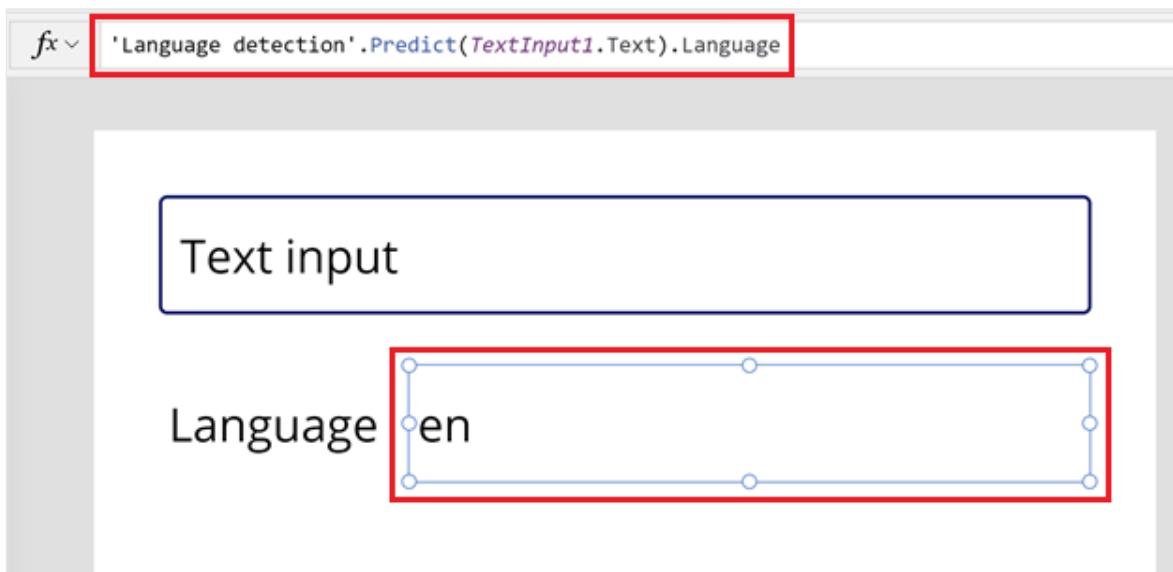


8. Select the text label added in the previous step.

9. Enter the following formula in the formula bar for the text label's **Text** property.



The label changes to the language code based on your locale. For this example, **en** (English).



10. Preview the app by selecting the **Play** button from the top-right corner of the screen.



11. In the textbox, enter `bonjour`. Notice that the language for French language (fr) appears below the textbox.



12. Similarly, try other language text. For example, entering `guten tag` changes the detected language to `de` for German language.

Best practices

- Try triggering model prediction from singular actions such as **OnClick** using a button rather than the **OnChange** action on a text input to ensure efficient use of AI Builder credits.
- To save time and resources, save the result of a model call so you can use it in multiple places. You can save an output into a global variable. After you save the model result, you can use the language elsewhere in your app to show the identified language and its confidence score in two different labels.

```
Power Apps
Set(lang, 'Language detection'.Predict("bonjour").Language)
```

Input and output by model type

This section provides inputs and outputs for custom and prebuilt models by model type.

Custom models

Model type	Syntax	Output
Category classification	'Custom text classification model name'.Predict(Text: String, Language?: Optional String)	{AllClasses: {Name: String, Confidence: Number}[], TopClass: {Name: String, Confidence: Number}}
Entity extraction	'Custom entity extraction model name'.Predict(Text: String, Language?: String(Optional))	{Entities:[{Type: "name", Value: "Bill", StartIndex: 22, Length: 4, Confidence: .996}, {Type: "name", Value: "Gwen", StartIndex: 6, Length: 4, Confidence: .821},]]}
Object detection	'Custom object detection model name'.Predict(Image: Image)	{ Objects: {Name: String, Confidence: Number, BoundingBox: {Left: Number, Top: Number, Width: Number, Height: Number}}[]}

Prebuilt models

ⓘ Note

Prebuilt model names are shown in your environment's locale. The following examples show the model names for English language (en).

Model type	Syntax	Output
Business card reader	'Business card reader'.Predict(Document: Base64 encoded image)	{Fields: {FieldName: {FieldType: "text", Value: {Text: String, BoundingBox: {Top: Number, Left: Number, Height: Number, Width: Number}}}}}
Category classification	'Category classification'.Predict(Text: String, Language?: Optional String,)	{AllClasses: {Name: String, Confidence: Number}[], TopClass: {Name: String, Confidence: Number}}
Identity document reader	'Identity document reader'.Predict(Document: Base64 encoded image)	{Context: {Type: String, TypeConfidence: Number}, Fields: {FieldName: {FieldType: "text", Confidence: Number, Value: {Text: String, BoundingBox: {Top: Number, Left: Number, Height: Number, Width: Number}}}}}

Model type	Syntax	Output
Invoice processing	<pre>'Invoice processing'.Predict(Document: Base64 encoded image)</pre>	<pre>{ Fields: { FieldName: { FieldType: "text" "date" "number", Confidence: Number, Value: { Text: String, [Date: Date] [Number: Number], BoundingBox: { Top: Number, Left: Number, Height: Number, Width: Number } } }, Tables: { Items: { Rows: { FieldName: { FieldType: "text" "date" "number", Confidence: Number, Key: { Name: String, }, Value: { Text: String, [Date: Date] [Number: Number], BoundingBox: { Top: Number, Left: Number, Height: Number, Width: Number } } } } } }</pre>
Key phrase extraction	<pre>'Key phrase extraction'.Predict(Text: String, Language?: Optional String))</pre>	<pre>{ Phrases: String[]}</pre>
Language detection	<pre>'Language detection'.Predict(Text: String)</pre>	<pre>{ Language: String, Confidence: Number}</pre>
Receipt processing	<pre>'Receipt processing'.Predict(Document: Base64 encoded image)</pre>	<pre>{ Context: { Type: String, TypeConfidence: Number }, Fields: { FieldName: { FieldType: "text" "date" "number", Confidence: Number, Value: { Text: String, [Date: Date] [Number: Number], BoundingBox: { Top: Number, Left: Number, Height: Number, Width: Number } } }, Tables: {Items: {Rows: {FieldName: { FieldType: "text" "date" "number", Confidence: Number, Key: { Name: String, }, Value: { Text: String, [Date: Date] [Number: Number], BoundingBox: { Top: Number, Left: Number, Height: Number, Width: Number } } } } } }</pre>
Sentiment analysis	<pre>'Sentiment analysis'.Predict(Text: String, Language?: Optional String)</pre>	<pre>{ Document: { AllSentiments: [{ Name: "Positive", Confidence: Number }, { Name: "Neutral", Confidence: Number }, { Name: "Negative", Confidence: Number }], TopSentiment: { Name: "Positive" "Neutral" "Negative", Confidence: Number } } Sentences: { StartIndex: Number, Length: Number, AllSentiments: [{ Name: "Positive", Confidence: Number }, { Name: "Neutral", Confidence: Number }, { Name: "Negative", Confidence: Number }], TopSentiment: { Name: "Positive" "Neutral" "Negative", Confidence: Number } [] }</pre>

Model type	Syntax	Output
Text recognition	'Text recognition'.Predict(Document: Base64 encoded image)	{Pages: {Page: Number, Lines: {Text: String, BoundingBox: {Left: Number, Top: Number, Width: Number, Height: Number}, Confidence: Number}[]} } []}
Text translation	'Text translation'.Predict(Text: String, TranslateTo?: String, TranslateFrom?: String)	{Text: String, // Translated text DetectedLanguage?: String, DetectedLanguageConfidence: Number} }

Examples

Every model is invoked using the predict verb. For example, a language detection model takes text as an input and returns a table of possible languages, ordered by that language's score. The score indicates how confident the model is with its prediction.

Input	Output
'Language detection'.Predict("bonjour")	{ Language: "fr", Confidence: 1}
'Text Recognition'.Predict(Image1.Image)	{ Pages: [{Page: 1, Lines: [{Text: "Contoso account", BoundingBox: {Left: .15, Top: .05, Width: .8, Height: .1}, Confidence: .97}, {Text: "Premium service", BoundingBox: {Left: .15, Top: .2, Width: .8, Height: .1}, Confidence: .96}, {Text: "Paid in full", BoundingBox: {Left: .15, Top: .35, Width: .8, Height: .1}, Confidence: .99}] }] }

See also

- [What are canvas apps?](#)
- [Get started with formulas in canvas apps](#)
- [Microsoft Power Fx](#)
- [Training: Improve business performance with AI Builder \(learning path\)](#)

AI Builder in SharePoint with Microsoft Syntex overview

Article • 01/03/2023

With the Microsoft Syntex service, you're able to create AI Builder models in SharePoint. To learn more, go to [Overview of Microsoft Syntex](#).

From the **Classify and Extract** menu in your SharePoint library, you can apply an existing AI model to a library or create a new one. Two model types are powered by AI Builder:

- **Freeform document processing:** Creates AI Builder document processing models for unstructured or freeform documents like contracts, statements of work, letters, and more.
- **Structured document processing:** Creates AI Builder document processing models for structured documents like invoices, purchase orders, delivery orders, tax documents, and more.

Once an AI Builder model is applied to a library, every document added to the library will be processed by the applied model. The results are displayed as new library columns.

To learn about requirements and get step-by-step instructions on how to use this service, go to [Work with models](#).

Training data storage

If you're using AI Builder models, training data is stored in [Microsoft Dataverse](#).

Only training data is stored in Dataverse. It's used solely to train the AI Builder model and is never used for any other purpose. Training data is never shared externally.

Dataverse has strong security mechanisms that prevent unauthorized access to user data. Data stored to train your AI Builder model is only accessible by:

- The owner of the model.
- Individuals with Power Platform **System Administrator** and **System Customizer** roles in your organization.

To learn more, go to [Roles and security in AI Builder](#).

See also

[AI Builder document processing models](#)

[Feature availability by region](#)

[Roles and security in AI Builder](#)

Use AI Builder in Teams

Article • 02/14/2023

You can use Power Automate templates in Teams that solve a specific business purpose using AI Builder actions.

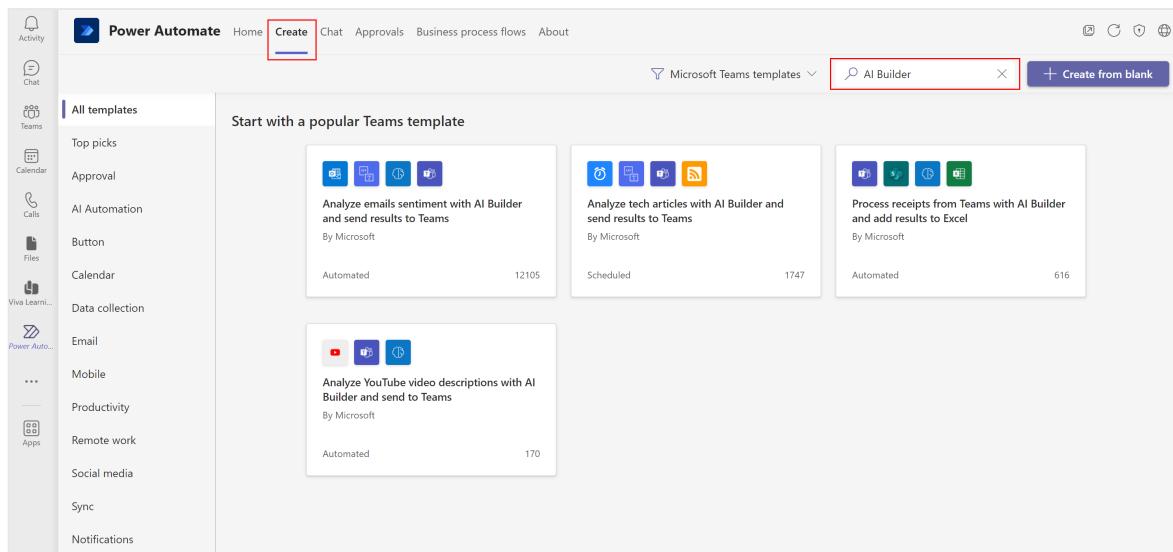
Discover AI Builder templates in Teams

1. Go to your Teams app or [Teams web](#).
2. On the bottom left, select the **Apps** icon.
3. Search for **Power Automate** and install it. The app appears in the left panel in Teams.

You can pin the app to keep it there when you reopen Teams.

4. Select the Power Automate app, and then select the **Create** tab.

From there, you can search for the AI Builder templates listed in the next section in this article.



Create a flow from a template

1. From the list of AI Builder templates, choose the one that fits your needs the most.
 - If you don't have a [Microsoft Power Platform environment with database](#), you'll see the following dialog. Select **Continue** to add a database. It usually

takes a couple of minutes (in some cases it can take more than one hour) and has to be done only once.

Create a flow X

Send a Teams message when receiving an email
Automated

Send a private message in Teams when receiving an email in Outlook. Message will contain extract of the email, along with the text sentiment analyzed using AI Builder



No Dataverse database found

This flow requires a [Dataverse database](#). By clicking on continue, a Dataverse database will be installed in this environment.

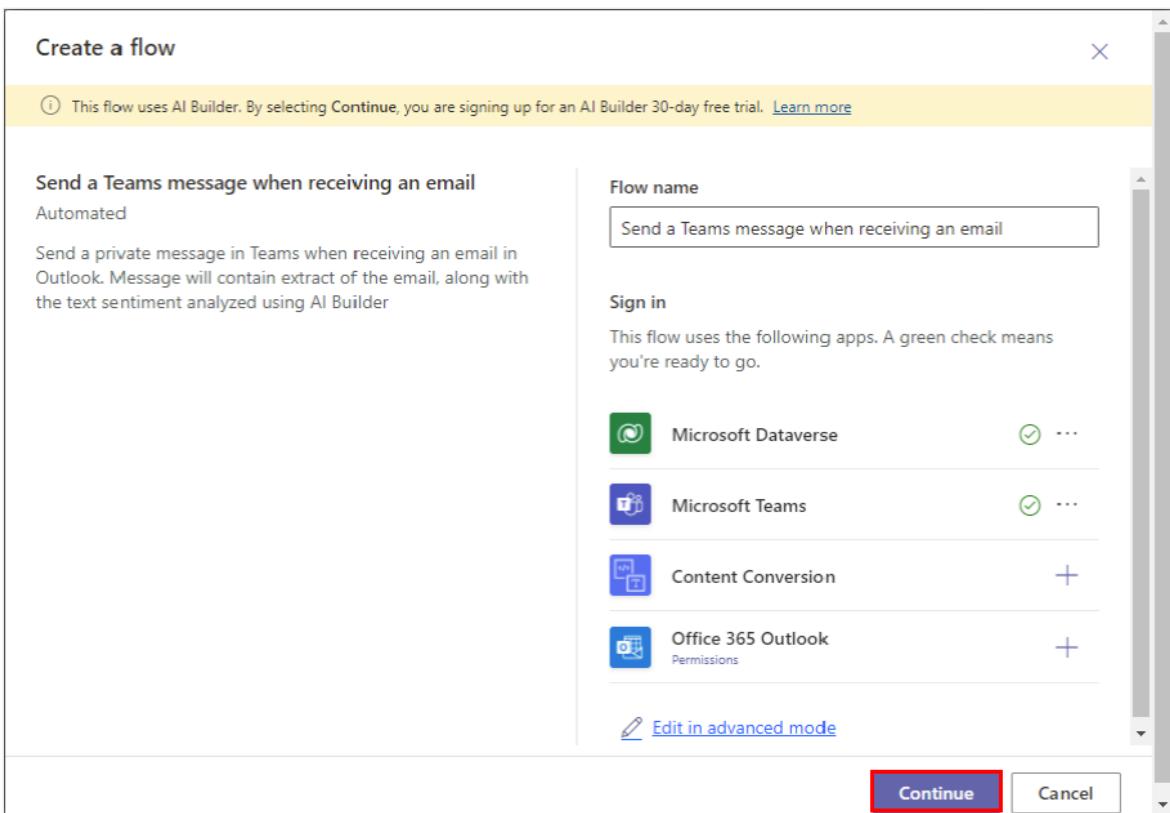
Continue Cancel

- If you already have a Power Platform environment with database, you'll land on the simplified template experience for Teams. The first screen asks you to validate connections.

2. Once your connections are valid, select **Continue**.

! Note

An AI Builder trial will be started or extended if needed. This action will be performed silently upon selecting **Continue**.



3. Enter the parameters required by the template to create the flow.

! Note

It's also possible to select **Edit in advanced mode**. This will open the full flow editor within Teams.

4. Once you entered all the parameters, select **Create flow**. After few seconds, your flow will be created and ready to use.

Create a flow

X

Send a Teams message when receiving an email

Automated

Send a private message in Teams when receiving an email in Outlook. Message will contain extract of the email, along with the text sentiment analyzed using AI Builder

Set up your flow

Office 365 Outlook From

Sender email addresses separated by semicolons (If any match)

* Microsoft Teams Recipient

Add an email address

 [Edit in advanced mode](#)

 Back

Create flow

Cancel

Created flows are visible and can be modified from the Power Automate home page.

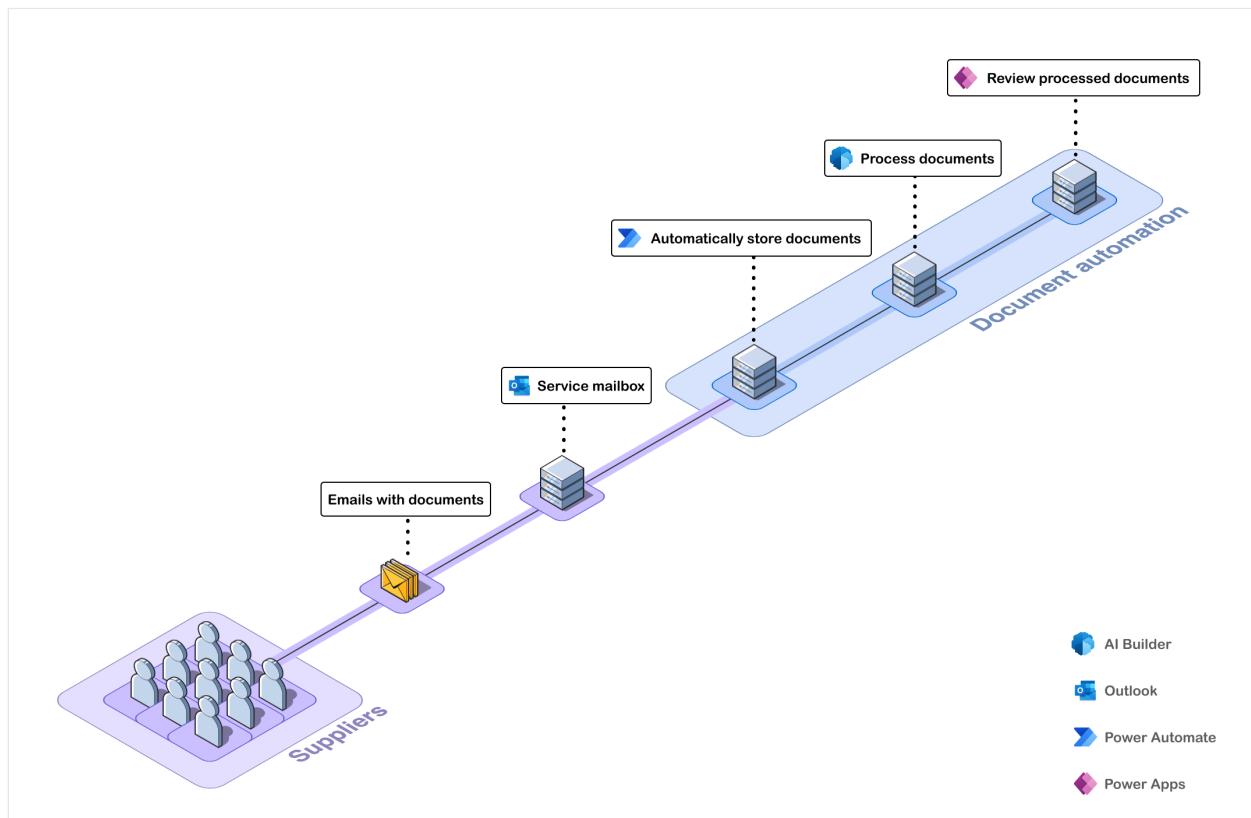
See also

- [Use flows in Microsoft Teams](#)
- [Video: How to set up instant Microsoft Teams notifications for negative emails in 1 minute ↗](#)

Document automation toolkit

Article • 01/05/2023

The Document automation toolkit allows you to easily set up a rich and robust document processing solution using AI Builder, Power Automate, Power Apps, and Microsoft Dataverse.



Power Automate will orchestrate the overall process while AI Builder will bring the intelligence required to efficiently extract information from documents. Power Apps will allow users to manually review and approve documents, and Dataverse will manage the document queue, store all the data, files, and configuration information.

Get started

You can access the [document automation toolkit](#) in Power Automate.

Learn about the toolkit

To get step-by-step instructions on how to install, configure, and use the document automation kit, go to [Training: Automate the processing of documents with the AI Builder prepackaged solution \(module\)](#).

See also

[Create a document processing model](#)

Administer AI Builder

Article • 12/13/2022

Microsoft Power Platform administrators can use the [Power Apps admin center](#) and the [Power Platform admin center](#) to manage environments and settings for Power Apps and AI Builder. For more information, go to [Power Platform administrator guide](#).

The availability, which is also known as the release status, of AI Builder is dependent on your region. For a breakdown of AI Builder feature availability by region, go to [Feature availability by region](#).

For more information, go to the [AI Builder governance whitepaper](#) and select Download.

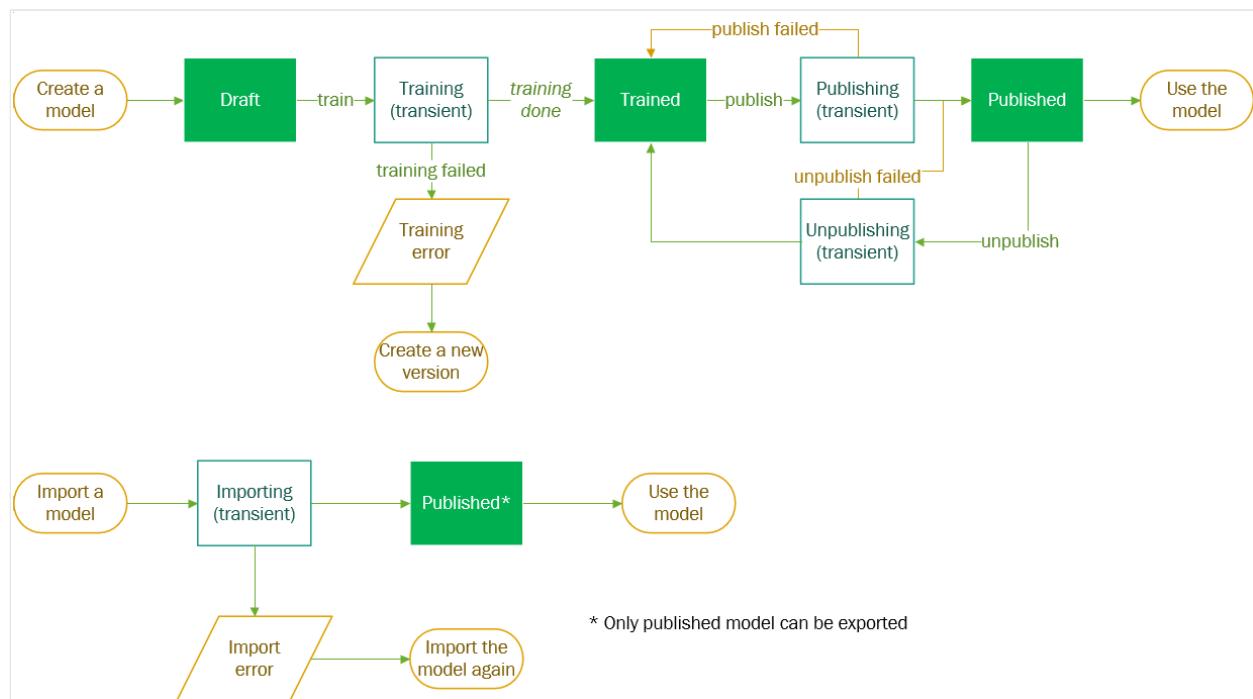
AI custom model and environment lifecycle

This section applies to AI Builder custom models. It doesn't apply to prebuilt models.

Lifecycle states of a model

A model can go through different states depending on the makers' actions. The model states are stored through configurations within the AI Configuration table.

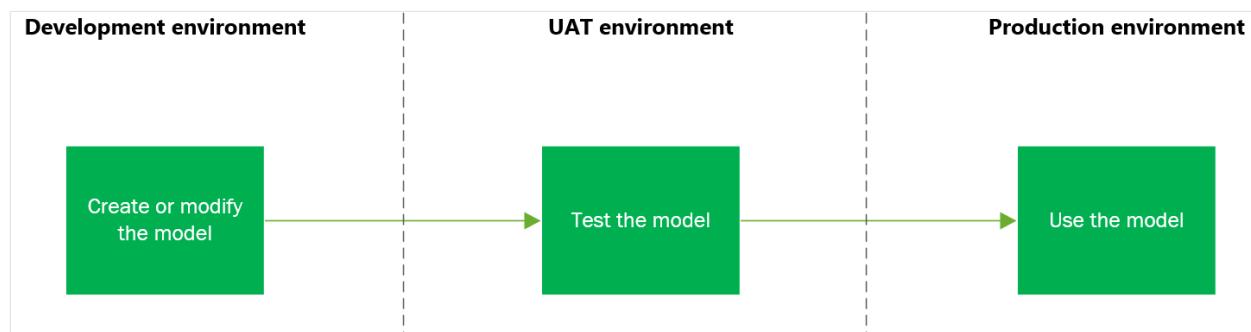
The states are draft, training (transient), trained, publishing (transient), published, unpublishing (transient), training error, importing (transient), and import error.



Application lifecycle management

Makers should be able to continuously update and deploy their models across single or multiple environments.

Managing new versions of a model often requires going through different environments. A typical scenario would be to make model changes in a *development environment*, qualify the model in a *test environment*, and predict in a *production environment*.



In AI Builder, all the environments need to be provided with a Microsoft Dataverse database.

Moving models between environments can be done through the solution concept. Solutions are vehicles to move components between Microsoft Power Platform environments. To learn more, go to [Introduction to solutions](#).

For more information on how to distribute an AI model as a solution component, go to [Distribute your AI model](#).

Environment lifecycle

AI Builder models are fully moved, along with user data, during environment backup/restore and environment copy operations.

After restore and copy operations, document processing and object detection models may be in the importing state for a few minutes while copies are made in the back end.

Backup and restore

Microsoft Dataverse has backup and restore capabilities to help protect your apps' data, providing continuous availability of service. System administrators and delegated admin users can use the standard capabilities described in [Back up and restore environments](#).

- Backup and restore are fully supported for prediction, object detection, document processing, and prebuilt models.

 **Note**

For object detection and document processing models, the restore process might take some time to be completed. The AI Builder models list shows an "importing" status message while the restore operation is in progress.

- **For models not supported by backup and restore:** If you restore an environment, you'll have to retrain and republish these models to make them available again.

Manage capacity

AI Builder consumption model

AI Builder offers a subscription model allowing you to purchase add-ons.

Only certain actions in the product consume credits. The following list isn't all-inclusive and preview scenarios don't consume credits.

AI Builder Studio	Power Apps	Power Automate
Train an object detection model.	Select...	Run a flow using any of the actions inside the AI Builder category.
Perform a Quick test on a trained object detection and document processing model.	Scan a business card with the business card reader.	Run the generic action Perform a bound action of Dataverse on the entity AI Models and action name Predict .
Use the Try it out of models with custom documents (appearing in the Get straight to productivity section).	Analyze with the document processor.	
Batch runs of the prediction and trainable category classification models for each row to be predicted.	Detect with the object detector. + New image with the text recognizer.	

Each AI Builder model has a different credit consumption mechanism. To perform an assessment, go to the [AI Builder calculator](#).

Allocate credits

By default, the credits are unallocated and available as a pool on the tenant, which can be used on any environment. The administrator can restrict usage by allocating all credits to specific environments.

Note

This is how administrators stay in control of *where* AI is used in their organization and, with the role assignments described in [Roles and security in AI Builder](#), *who* is using it.

As an administrator, you'll assess which environments must consume AI Builder credits. Use the [AI Builder calculator](#) to define how many predictions will happen in a monthly period on each one and assess the credits to allocate.

To learn how to allocate credits in the Power Platform admin center, go to [Allocate or change capacity in an environment](#).

Monitor usage

As an administrator, you have access to a [consumption report](#) that provides the AI credits consumption on a chosen period per environment. This will allow you to fine-tune the credits allocation, which can be updated at any time.

To learn how to download reports, go to [Allocate or change capacity in an environment](#).

Where and how are data stored in Dataverse?

Your AI model is deployed in the region that hosts the environment. For example, if your environment is created in the Europe region, your model is deployed in datacenters in Europe. For more information, go to [Environments overview](#).

Images and documents used for training purposes in object detection and document processing models are persisted in Dataverse. In contrast, images and documents used at prediction time aren't persisted. Examples of non-persisted images and documents are those in a Power Apps component framework (PCF) control and in Power Automate.

Enable or disable AI Builder preview features

Some AI Builder features are released for general availability. Others remain in preview release status.

Preview features appear on the Explore page with the **Preview** label. In the Power Platform admin center, administrators control whether users have access to preview features.

By default, the **AI Builder preview models** feature is enabled for any eligible environment. Eligible environments must have Microsoft Dataverse and be in a [supported region](#). If the environment isn't eligible, the **AI Builder preview models** feature doesn't appear in the Power Platform admin center.

To control AI Builder preview feature availability:

1. Sign in to the [Power Platform admin center](#).
2. In the admin center, go to **Environments** > *[select an environment]* > **Settings** > **Features**.
3. On the **Features** settings page, under **AI Builder**, enable or disable **AI Builder preview models**.

Important points related to enabling or disabling the feature

- If you disable **AI Builder preview models**:
 - We don't delete existing models that users of this environment have created.
 - AI Builder components are disabled.
 - Existing experiences that use existing AI Builder components will fail or return errors.
 - Admins and owners can delete preview models.
- If you enable **AI Builder preview models** again:
 - AI Builder components are available again.
 - Components function as they did before the feature was disabled (assuming nothing else has changed).

For more information about enabling or disabling features in the Power Platform admin center, go to [Manage feature settings](#).

Data loss prevention (DLP)

You can control data loss prevention (DLP) policies within Power Platform admin center, Data policies menu item.

Connectors can be listed in three (3) categories: *Business*, *Non-business*, and *Blocked*.

- AI Builder is part of the Dataverse connector.
- Business and Non-business connectors can't share data within the same consumption experience in Microsoft Power Platform.
 - For example, if you add the Dataverse connector in the *Business* category, and Microsoft Outlook in the *Non-business* category, you won't be able to create a Power Automate flow that gets the output of an AI Builder model and sends it to a recipient in Outlook.
- Blocked connectors can't be used in Power Platform consumption experiences.

To learn more, go to [Data loss prevention policies](#).

Move and copy environments

For prediction and prebuilt models, moving and copying environments is fully supported. For other models, after you move or copy an environment, you have to retrain and republish existing models to make them available again.

See also

[Roles and security in AI Builder](#)

[Feature availability by region](#)

[AI Builder consumption report](#)

AI Builder consumption report

Article • 12/13/2022

AI Builder is licensed as a capacity add-on that must be allocated to a Microsoft Power Platform environment by an administrator. Each AI builder capability consumes credits at a different rate.

- The [AI Builder calculator](#) allows you to perform an assessment of what you need for your business.
- The [AI Builder licensing page](#) gives you more details on how AI Builder capacity works.

In addition, administrators can download an Excel report that shows the actual capacity consumption in your tenant. The consumption report shows the capacity used in the 30 days preceding the selected target date for each environment. This makes it easier to compare your allocated capacity with the actual capacity consumption of your organization so you can fine tune your capacity allocation.

Consumption reports aren't available in Government Community Cloud (GCC and GCC High).

View the consumption report

To download and view an Excel file showing your AI credit consumption report:

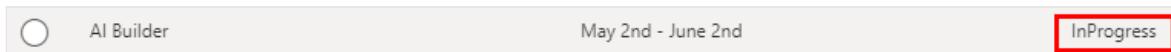
1. Sign in to the [Power Platform admin center](#).
2. On the left pane, select **Resources > Capacity**.
3. On the **Summary** tab, select **Download reports** in the **Add-ons** section.

The screenshot shows the Power Platform admin center interface. The left sidebar has a 'Capacity' menu item highlighted with a red box. The main content area is titled 'Capacity' and has tabs for 'Summary', 'Storage capacity', 'Add-ons', and 'Microsoft Teams capacity usage'. The 'Summary' tab is selected. It displays storage capacity usage with a note: 'You have available capacity. Learn more about managing capacity.' Overall usage is shown as 149.56 TB used, with 21.66% available. Below this, there are sections for 'Database', 'Log', and 'File' usage. To the right, there's a table for 'Storage capacity, by source' showing 'Org (tenant) default', 'User licenses', and 'Additional capacity'. At the bottom, the 'Add-ons' section includes a list of items like 'App passes', 'Portal page views', 'AI Builder credits', and 'Portal logins'. The 'AI Builder credits' item is highlighted with a red box. A 'Download reports' button is also highlighted with a red box.

4. On the menu at the top, select **+New**.
5. In the **Choose a report** dropdown list, select **AI Builder**.
6. Select **Submit**.

The screenshot shows the Power Platform admin center interface. On the left, there's a sidebar with various navigation options like Home, Environments, Analytics, Resources, Help + support, Data integration, Data (preview), Policies, and Admin centers. The main area displays a list of 'Downloadable Reports' with columns for Report Type and Report Dates. A red box highlights the '+ New' button at the top left of the main content area. To the right, a modal window titled 'Request a report' is open. It contains a message about notifications for ready reports and a dropdown menu labeled 'Choose a report' which is also highlighted with a red box. At the bottom of the modal are 'Submit' and 'Cancel' buttons, with 'Submit' also highlighted with a red box.

The **Downloadable Reports** list shows that a new report is being generated.



7. After the new report is generated, select the report in the list.
8. At the top of the screen, select **Download** to download the report as an Excel file.

Understand the consumption report

The report shows AI credits consumed by date for each environment.

	A	B	C	D	E
1	Date	UserId	EnvironmentId	AIConsumption	IsTrial
2	06/04/2022	8c8ba765-7348-4bd3-bcd8-d0c55c99f5b9	ed5cafb1-959d-42dc-8a8a-32aa6a6ce28d	46000	FALSE
3	06/04/2022	9b11ad8e-fda3-49b5-bac5-88adb2778f9a	e0414dba-7f44-48e4-831a-54281add2b62	1047000	FALSE
4	06/04/2022	3d76d363-0a68-4a40-a7c7-0e3ea02b3b1c	22cbf046-c4b3-439a-9ff5-ccd67ca526f1	611000	FALSE
5	06/04/2022	3d76d363-0a68-4a40-a7c7-0e3ea02b3b1c	398896f1-0a7a-416d-87ee-30af26255ca3	<1K	FALSE
6	06/04/2022	726fd730-ec57-4c40-9aa4-6e321e514012	cc16f503-805b-4209-96ba-639d07791dbe	<1K	FALSE
7	06/04/2022	45167683-508c-42b7-8bb3-85e979b9057b	0bb5d634-4d1f-446c-a534-e3e9fc090796	<1K	FALSE
8	06/04/2022	69634744-284a-428e-a2f8-e67332b3401b	86db6687-38b0-43ed-bdb9-436fa3a7547d	<1K	FALSE
9	06/04/2022	ee4ef84f-a01c-4307-b802-e6fce0d4af85	0bb5d634-4d1f-446c-a534-e3e9fc090796	10000	FALSE
10	06/04/2022	f2889753-2469-47ac-bd9e-ccf7c8e4a96a	d1f35a86-6e21-41bb-a5b3-91dd95978dc8	<1K	FALSE
11	06/04/2022	2c6b99ac-1b86-49cd-a344-72498fb67ff0	d1f35a86-6e21-41bb-a5b3-91dd95978dc8	26000	FALSE
12	06/04/2022	471b4c1b-4785-4d02-a575-9c1468c87f58	d1f35a86-6e21-41bb-a5b3-91dd95978dc8	<1K	FALSE
13	06/04/2022	fd7a1bfd-0765-4167-86e5-2e4fb0ca839	398896f1-0a7a-416d-87ee-30af26255ca3	<1K	FALSE
14	06/04/2022	c5ebbf5b-7fa8-4708-8603-3d0fd4ba4891	1efe9e51-3f7e-48fe-b0cb-f9def730ad7f	1000	TRUE
15	06/04/2022	26d2e646-d9a2-4208-bb6f-7b9a0906393c	4f6ee499-57d2-4768-967d-802893f11649	67000	TRUE
16	06/04/2022	66371b18-ada9-495c-9a79-a5410fe9fa43	09809f57-fb68-4915-b673-a9a09b4bcdb4	1000	TRUE
17	06/04/2022	fa597fed-098b-4e28-b1b7-e848b4458c5d	7f0defc6-9dbe-4f62-9893-52abf0e9b56f	22000	TRUE
18	06/04/2022	e912cb9f-d9a8-46d0-b218-5e6d847b2c8e	839eace6-59ab-4243-97ec-a5b8fcc104e4	<1K	TRUE
19	07/04/2022	8c8ba765-7348-4bd3-bcd8-d0c55c99f5b9	ed5cafb1-959d-42dc-8a8a-32aa6a6ce28d	40000	FALSE
20	07/04/2022	3d76d363-0a68-4a40-a7c7-0e3ea02b3b1c	22cbf046-c4b3-439a-9ff5-ccd67ca526f1	595000	FALSE
21	07/04/2022	9b11ad8e-fda3-49b5-bac5-88adb2778f9a	e0414dba-7f44-48e4-831a-54281add2b62	894000	FALSE
22	07/04/2022	726fd730-ec57-4c40-9aa4-6e321e514012	cc16f503-805b-4209-96ba-639d07791dbe	<1K	FALSE
23	07/04/2022	45167683-508c-42b7-8bb3-85e979b9057b	0bb5d634-4d1f-446c-a534-e3e9fc090796	<1K	FALSE
24	07/04/2022	ee4ef84f-a01c-4307-b802-e6fce0d4af85	0bb5d634-4d1f-446c-a534-e3e9fc090796	178000	FALSE
25	07/04/2022	d3a18191-3c83-4d9a-b33f-82bf32d9d64e	4719799d-f124-441e-895d-2897812f6792	9000	FALSE

- **UserId** is the identifier visible in the Dataverse "User" table.
- **EnvironmentId** is the identifier visible in your Power Apps or Power Automate URL (for example, <https://make.powerapps.com/environments/%GUID%>).
- **<1K** in the **AIConsumption** column means that you've consumed fewer than 1,000 credits on that day and that environment.

You can compare the aggregated consumption for the current calendar month on each environment with what has been allocated. Using this information, you can take action to allocate more credits to environments that are in overage. Remember, the number of allocated credits by environment is visible and can be changed in the Power Platform admin center [Add-ons](#) page.

See also

- [Microsoft Power Apps and Power Automate Licensing Guide](#)
- [AI Builder licensing FAQ](#)

Roles and security in AI Builder

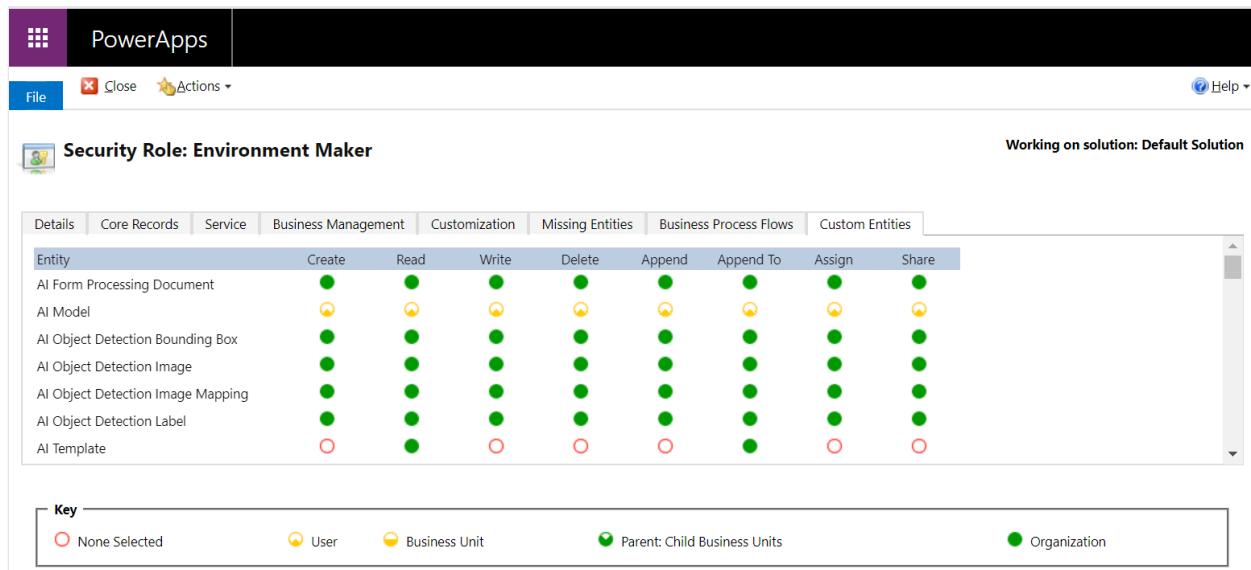
Article • 07/08/2022

AI Builder relies on environment security and Microsoft Dataverse security roles and privileges to grant access to AI features in Microsoft Power Apps. For more information, see [Security overview](#).

Some privileges are set by default in Dataverse. This allows built-in security roles to use AI Builder without further actions from system administrators. Specifically:

- Environment Makers can use AI Builder to create AI models.
- Basic Users can access data by using the models embedded in Power Apps.
- System Administrators and System Customizers can access all AI models created in the environment.

These security roles have privileges to the AI Builder tables in Dataverse. Custom security roles can create AI models if they have the same access to the AI Builder tables as the Environment Maker role.



The screenshot shows the Microsoft Power Apps security roles interface. The top navigation bar includes 'PowerApps', 'File', 'Close', 'Actions', and 'Help'. The main title is 'Security Role: Environment Maker' with a subtitle 'Working on solution: Default Solution'. Below the title, there's a 'Key' section with four items: 'None Selected' (red circle), 'User' (blue circle), 'Business Unit' (yellow circle), 'Parent: Child Business Units' (green circle), and 'Organization' (dark green circle). A table below lists entities and their permissions across various operations: Create, Read, Write, Delete, Append, Append To, Assign, and Share. The table rows include 'AI Form Processing Document', 'AI Model', 'AI Object Detection Bounding Box', 'AI Object Detection Image', 'AI Object Detection Image Mapping', 'AI Object Detection Label', and 'AI Template'. Most entities have 'User' and 'Organization' selected for most operations, except for 'AI Model' which has 'Business Unit' selected for Read, Write, and Append.

Entity	Create	Read	Write	Delete	Append	Append To	Assign	Share
AI Form Processing Document	●	●	●	●	●	●	●	●
AI Model	○	○	○	○	○	○	○	○
AI Object Detection Bounding Box	●	●	●	●	●	●	●	●
AI Object Detection Image	●	●	●	●	●	●	●	●
AI Object Detection Image Mapping	●	●	●	●	●	●	●	●
AI Object Detection Label	●	●	●	○	●	●	○	●
AI Template	○	●	○	○	○	●	○	○

Scenarios such as object detection, category classification, and prediction need read access to Dataverse tables. Make sure Environment Makers have access to them. They need those tables for objects to detect, tagged text, and input data.

Some features need System Customizer privileges to publish your AI models and to allow them to be consumed. These actions can make changes to the Dataverse schema. Administrators should assign System Customizer privileges to users who want to create such AI models.

When you create a prediction AI model, a new data column is added to the input table to store the prediction results. For this reason, you need at least System Customizer

rights to publish the model for the first time.

For category classification AI models, a data table is created for every new model as soon as the model runs for the first time. Therefore, only System Customizers or System Administrators can run the model. After the model runs, System Administrators must modify the access rights to the newly created category classification table in Dataverse to allow users to use the results.

Roles

Microsoft Dataverse permissions have been mapped to the Dataverse standard roles. Assigning these roles to a user will provide the necessary privileges to use AI Builder features as described in this table:

Privilege	System Administrator/Customizer	Environment Maker	Basic User	No privilege
View AI Builder Explore page	✓	✓	✓	X
Create a model	✓	✓	X	X
View and use a created model	✓	owned or shared model	owned or shared model	X
Create a flow to call a model	✓	✓	X	X
Create an app to call a model	✓	✓	X	X
Run a flow using a model	✓	owned or shared flow using an owned or shared model	owned or shared flow using an owned or shared model	X
Run an app using a model	✓	owned or shared app using an owned or shared model	owned or shared app using an owned or shared model	X

Tables and privileges

By default, a model is only accessible by the owner of the model, so it must be shared to be used by other users. To share a model:

1. In the left panel, select **AI Builder > Models**.
2. Find and select the model's name to access its details page.
3. Select **Share** in the top-left corner.

The following table shows the AI Builder system tables, which are installed by default on every new environment. The tables are used to store the model configuration and training data. Each table shows the Dataverse privilege applied by AI Builder, including when a user shares a model. See the legend below the table.

Dataverse table	Contains	Create Dataverse privilege	Use Dataverse privilege
AI Builder Dataset (FP, OD, EE)	Model's training configuration	<input type="radio"/>	
AI Builder Dataset File (FP, OD)	Model's training configuration	<input type="radio"/>	
AI Builder Dataset Record (EE)	Model's training data	<input type="radio"/>	
AI Builder Dataset Container (FP, OD, EE)	Model's training configuration	<input type="radio"/>	
AI Builder File (FP, OD)	Model's training files	<input type="radio"/>	
AI Builder File Attached Data (FP, OD)	Model's training configuration	<input type="radio"/>	
AI Configuration	Model's versions	<input type="radio"/>	<input type="radio"/> (when shared)
AI Model	Model	<input type="radio"/>	<input type="radio"/> (when shared)
AI Template	Model type stereotype	<input checked="" type="radio"/>	<input checked="" type="radio"/>

Dataverse table	Contains	Create Dataverse privilege	Use Dataverse privilege
User-defined table to be predicted (batch P & CC only)		<input type="radio"/>	<input checked="" type="radio"/>

- Users have access to the rows they've created.
- Users must be granted access to the subset of rows required for the business.
- Users have access to all the rows of the table.

FP: document processing OD: object detection EE: entity extraction P: prediction CC: category classification

The training files stored in the AI Builder File table are accessible only by the person who has created the model. The exception is the administrator, who can view and delete any model, related data, and configuration.

AI Builder doesn't support shared ownership of a model. It's possible to change the owner by following this procedure in [Share your AI model](#).

See also

[Security concepts in Dataverse](#)

Feature availability by region

Article • 05/26/2023

AI Builder was initially released in Europe and the United States. Other regions continue to be added; however, the availability and release status of AI Builder features vary by location.

Which region does my AI Builder subscription belong to?

Your AI Builder models are deployed in the region that hosts your Microsoft Power Platform environment. For example, if your environment is created in the Europe region, your AI Builder models are deployed in data centers in Europe and are subject to the availability status for Europe.

Availability and release status of features by region

The following tables show which AI Builder features are available in each region, and the release status (general availability or preview) for each feature. A dash (-) indicates that the feature isn't available.

Note

For a view of what's planned for AI Builder, including new features, release status, and regional availability, go to the [AI Builder release plans](#).

Custom models

Feature	Asia	Australia	Canada	Europe	France	Germany	India	Japan	Norway	South Africa	South America	South Korea	Switzerland	United Arab Emirates	United Kingdom
Prediction	GA	GA	GA	GA	GA	GA	GA	GA	-	-	GA	-	GA	GA	GA
Category classification	GA	GA	GA	GA	GA	GA	GA	GA	-	-	GA	-	GA	GA	GA
Entity extraction	GA	GA	GA	GA	GA	GA	GA	GA	-	-	GA	-	GA	GA	GA
Object detection	GA	GA	-	GA	-	-	GA	GA	-	-	-	-	-	-	GA
Document processing (for structured and semi-structured documents)	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Document processing (for unstructured documents)	GA	GA	GA	GA	GA	-	GA	GA	-	-	GA	-	-	-	GA

Prebuilt models

Feature	Asia	Australia	Canada	Europe	France	Germany	India	Japan	Norway	South Africa	South America	South Korea	Switzerland	United Arab Emirates	United Kingdom
Business card reader	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Category classification	Preview	Preview	Preview	Preview	Preview	Preview	Preview	Preview	Preview	-	-	Preview	-	Preview	Preview

Feature	Asia	Australia	Canada	Europe	France	Germany	India	Japan	Norway	South Africa	South America	South Korea	Switzerland	United Arab Emirates
Entity extraction	GA	GA	GA	GA	GA	GA	GA	-	-	-	GA	-	GA	GA
Identity document reader	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Invoice processing	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Key phrase extraction	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Language detection	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Receipt processing	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Sentiment analysis	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Text recognition	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA	GA
Text translation	-	-	-	GA	-	-	-	-	-	-	-	-	-	-

US Government

AI Builder is available in US Government environments with the features listed in the following tables. A dash (-) indicates that the feature isn't available.

Note

For more information on the Power Platform US Government environments and features, see these topics:

- [Power Apps US Government](#)
- [Power Automate US Government](#)

Custom models

Feature	Government Community Cloud (GCC)	Government Community Cloud – High (GCC High)	Department of Defense (DoD)
Prediction	GA	GA	-
Category classification	GA	GA	-
Entity extraction	GA	GA	-
Object detection	GA	GA	-
Document processing (for structured and semi-structured documents)	GA	GA	-
Document processing (for unstructured documents)	GA	GA	-

Prebuilt models

Feature	Government Community Cloud (GCC)	Government Community Cloud – High (GCC High)	Department of Defense (DoD)
Business card reader	GA	GA	-
Category classification	Preview	Preview	-

Feature	Government Community Cloud (GCC)	Government Community Cloud – High (GCC High)	Department of Defense (DoD)
Entity extraction	GA	GA	-
Identity document reader	GA	GA	-
Invoice processing	GA	GA	-
Key phrase extraction	GA	GA	-
Language detection	GA	GA	-
Receipt processing	GA	GA	-
Sentiment analysis	GA	GA	-
Text recognition	GA	GA	-
Text translation	GA	GA	-

US Government feature limitations

The following features available in the commercial version of AI Builder aren't available to US Government customers:

- 30-day user trials
- AI model copy across cloud boundaries (for example between Public and GCC or between GCC and GCC High)
- [Document Automation](#) isn't available in GCC and GCC High

For more information about other limitations of Power Platform US Government, go to these topics:

- [Power Apps US Government feature limitations](#)
- [Power Automate US Government feature limitations](#)

See also

[AI model types](#)

Common issues and resolutions for AI Builder

Article • 12/13/2022

The information about common issues in AI Builder and their potential workarounds is now available at [AI Builder troubleshooting](#).

See also

[Get support for AI Builder](#)

Get support for AI Builder

Article • 06/23/2022

You can get support for AI Builder on the AI Builder community forums. Features that have been released for general availability are eligible for support through Microsoft Support.

AI Builder community forums

Check Microsoft community forums to see if other users have posted a solution that might help you.

- [AI Builder in Power Apps community forum ↗](#)
- [AI Builder in Power Automate community forum ↗](#)

Microsoft support

For features that have been released for general availability, you can use the Power Platform admin center to request support from Microsoft.

1. Sign in to the [Power Platform admin center](#) ↗.
2. In the left pane, select **Help + Support**, and then select **New support request**.

For more information about the Microsoft Power Platform **Help + Support** experience, see [Get Help + Support](#).

Note

You can't create a new support request to get support from Microsoft for preview features. Go to the [AI Builder community forums](#) ↗ to get help with any issue that's related to a preview feature.

See also

- [Microsoft Dataverse in Power Apps community forum](#) ↗
- [Connect To Data in Power Automate community forum](#) ↗