

Mirasvit AI Assistant Manual

Getting Started

Welcome to the **AI Assistant** documentation.

Whether you are a new or an advanced user, you can find some useful information here.

Go ahead, dive in!

Firstly, please, find our extension in [My Downloadable Products](#) section of our store. Learn [how to install extension](#), and proceed with Quick Start, which will guide you to set up your **AI Assistant** service.

How to install the extension

How to install the extension using composer

1. Backup your store's database and web directory.
2. Login to the SSH console on your server and navigate to the root directory of the Magento 2 store.
3. Copy the installation instructions from the page [My Downloadable Products](#) to the SSH console and press ENTER.
4. Run command `php -f bin/magento module:enable Mirasvit_Core Mirasvit_Assistant` to enable the extension.
5. Run command `php -f bin/magento setup:upgrade` to install the extension.
6. Run command `php -f bin/magento cache:clean` to clean the cache.
7. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f bin/magento setup:static-content:deploy
```

How to install the extension manually

1. Backup your store's database and web directory.
2. Download archive from [My Downloadable Products](#).
3. Unzip the extension locally.
4. Copy unzipped folder to the root directory of the Magento store.
5. Run command `php -f bin/magento module:enable Mirasvit_Core Mirasvit_Assistant` to enable the extension.
6. Run command `php -f bin/magento setup:upgrade` to install the extension.
7. Run command `php -f bin/magento cache:clean` to clean the cache.

8. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f  
bin/magento setup:static-content:deploy
```

Note

If you install the module manually to the "app/code/" directory, you additionally need to install the required libraries through the composer:

```
composer require "liquid/liquid": "~1.4"
```

Configuration


AI Assistant integrates your Magento store with GPT3/ChatGPT, allowing you to leverage the latest AI technology advances in your business.



The AI will help you with SEO, customer support, and your blog:

- save time and make your store more visible in search engines by generating meta title and meta description for your products with the artificial intelligence.
- generate clear and polite messages when responding to your customer support tickets. Additionally, AI can check the grammar of your responses, bringing your customer support to a higher quality level.
- enhance your blog articles with a catchy headlines generated by AI. Additionally, the AI Assistant module can help you in enhancing the SEO of your blog by creating meta title, meta description, and meta keywords. The extension will also generate you a catchy Tweet for your blog article to enhance your SMM.

General Settings

Configuration

Scope: Default Config 

 MIRASVIT EXTENSIONS 

AI Assistant

General Settings

OpenAI Secret Key [global]

.....

Generate an OpenAI Secret Key if you don't have one.

OpenAI Model [global]

GPT-3.5 (gpt-3.5-turbo)
GPT-4o (gpt-4o)
GPT-4 Turbo (gpt-4o-turbo)
GPT-4 (gpt-4)
GPT-3.5 (gpt-3.5-turbo)
GPT-3 (text-davinci-003)

Enable logging [global]

No

Enable logging on the rule editor.

OpenAI Secret Key

Navigate to **Stores > Settings > Configuration > Mirasvit extensions > AI Assistant > General**. Paste the API key into the field **OpenAI Secret Key** in order to use the artificial intellect in your store.

Note

The AI Assistant extension accesses ChatGPT via an API. You need to sign up for a ChatGPT account in order to obtain the OpenAI secret key.

Generate the OpenAI secret API key on page platform.openai.com/account/api-keys. Click **Create new secret key** and copy the key.



ORGANIZATION

Personal ⓘ

Settings

Usage

Members

Billing

USER

API Keys

API keys

API key generated

Please save this secret key somewhere safe for your own reasons, **you won't be able to view it again** once you leave this account. If you lose this secret key, you'll need to create a new one.

```
sk-kN8W2lrT98y0Wf2IGgh6T3BlbkFJ5ta
```

+ Create new secret key

Default organization

If you belong to multiple orga

Using ChatGPT API costs money, which are billed from your ChatGPT account. The total pricing depends on the number of tokens generated by this language model. Refer to openai.com/api/pricing/ for more info on prices.

OpenAI Model

Choose a most suitable language models for you to utilize. AI Assistant works via various language models.

- **GPT-4o (gpt-4o)** is improved GPT-4 Turbo model. It is 2 times faster and up to 50% cheaper than GPT-4 Turbo model. Also has improved capabilities in non-English languages and uses a new tokenizer which tokenizes non-English text more efficiently than GPT-4 Turbo. Maximal support input is 128 000

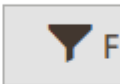
tokens. It was trained on data existing before Oct 2023.

- **GPT-4 Turbo (gpt-4-turbo-preview)** is improved GPT-4 model featuring improved instruction following, JSON mode and more. Maximal support input is 128 000 tokens. It was trained on data existing before Apr 2023.
- **GPT-4 (gpt-4)** is more capable for complex tasks and gives better results on large texts. Maximal support input is 8 192 tokens. It was trained on data existing before Sep 2021.
- **GPT-3.5 (gpt-3.5-turbo)** is optimized for chat. It is considered by OpenAI as a most capable GPT-3.5 model, and its price is 1/10th the cost of text-davinci-003. Maximal support input is 16 385 tokens. It was trained on data existing before Sep 2021.
- **GPT-3 (text-davinci-003)** is suited for any language task. It is configured for longer output of better quality and consistent instruction-following. Its input can be up to 4 097 tokens. Its trained on data before Jun 2021.

Prompt management

ChatGPT starts generating text in respond to a text prompt. Configure prompts for your store by navigating to **System > AI Assistant > Prompts**.

Prompts



14 records found

20

ID	Code	Title	Active	Position ↓	Scopes
1	mst_meta_title	Meta Title	Yes	10	product
2	mst_meta_keywords	Meta Keywords	Yes	10	product
3	mst_meta_description	Meta Description	Yes	10	product
4	mst_short_description	Short Description	Yes	10	product
5	mst_description	Description	Yes	10	product
6	mst_ticket	Message	Yes	10	ticket
7	mst_grammar	Fix Grammar	Yes	10	
8	mst_friendly	Friendly	Yes	10	
9	mst_blog_post_title	Title	Yes	10	post
10	mst_blog_post_meta_title	Meta Title	Yes	10	post
11	mst_blog_post_meta_keywords	Meta Keywords	Yes	10	post
12	mst_blog_post_meta_description	Meta Description	Yes	10	post

Add a new prompt or duplicate an existing one to use it as a template.

Configure the prompt

☰ Prompt "Meta Title" ▾

← Back

Delete

General Information

Active Yes

Code *

Title *

Prompt *

Scopes

Field Selectors *

One selector per line

Open AI Model ▾

Frequency penalty *

from 0 to 1 (e.g. 0.3)

Stop Sequences

- **Active** - Activates the prompt to be accessible for usage.
- **Code** - Prompt identifier to be used by the extension.
- **Title** - Prompt name that is visible in the Magento admin to the user.



Meta Title

Me

Meta Title
[store view]

- **Prompt** - Contains a short prompt for ChatGPT. Use variables to adjust the ChatGPT response to the specific request.
- **Scopes** - Select where the prompt will be available: product, blog, ticket.
- **Open AI Model** - Select Open AI model for a particular prompt. The **From config** value for this option means the prompt will use the default Open AI model from the configurations of the extension.
- **Stop Sequences** - Used to stop generation when a value is encountered in output.
- **Frequency penalty** - Controls the tendency for repeating words in the output. It can be from 0 to 1.
- **Field Selectors** - Add a Magento selector field in which the extension will insert the ChatGPT output.
- **Sort Order** - Sets the prompt position in the list of all prompts.
- **Open In Window** - When activated a window appears so you can see the Help Text or alter the prompt. When disabled the ChatGPT output immediately appears in the text field.

Joust Duffle Bag

Images And Videos

Search Engine Optimization

AI Assistant

Generate

- unique selling point

Meta Keywords
[store view]

Meta Description
[store view]

Maximum 255 chars. Meta Description should optimally be between

- **Help Text** - Additional text for ChatGPT to use when generating content. For example, additional info on the product.
- **Get Variables' Values From** - Select the store from which the assistant should grab values for variables. "Current store" means the assistant will autodetect the store. For automation rules "Current store" will be the store for which the prompt applies at the moment. This setting most useful for translations. Applicable for Products and Categories. Most useful for translation purposes.
- **Remove HTML Tags from Input** - If disabled, the extension will not remove HTML tags from variable values. The prompt size might increase which can lead to OpenAI maximum content length error (depends on the model). Enabled by default. Disabling this option most useful for translating content from WYSIWYG editor and the Page Builder.
- **Convert output to HTML** - Places the ChatGPT output within the HTML tags. Only available when the **Remove HTML Tags from Input** option is enabled.

Available variables

Use the variables in the prompt so the ChatGPT could generate a response that includes specific details on your store, blog content, customer.

Variable	Description
{{ global.input }}	Identifier of the input field.
{{ store.name }}	Store name.

Helpdesk

Variable	Description
{{ ticket.lastMessage }}	Includes the content of the last
{{ ticket.customer }}	Customer full name
{{ user.first_name }}	Agent first name
{{ user.name }}	Agent name
{{ user.last_name }}	Agent last name

Product edit page

Variable	Description
{{ product.name }}	Name of the product
{{ product.some_attribute }}	One product attribute
{{ product.attributes }}	List of useful attributes

Category edit page

Variable	Description
{{ category.some_attribute }}	One product attribute
{{ category.attributes }}	List of useful attributes

Blog edit page

Variable	Description
{{ post.title }}	Title of the blog post
{{ post.content }}	Main content of the blog post
{{ post.short_content }}	Short content of the blog post
{{ post.meta_title }}	Meta Title of the blog post
{{ post.meta_description }}	Meta Description of the blog post
{{ post.meta_keywords }}	Meta Keywords of the blog post
{{ post.data }}	Combined data of the blog post

Available field selectors

Field selector	Description
[name="product[meta_title]"]	Meta title on product page
[name="product[meta_keyword]"]	Meta keyword on product page
[name="product[meta_description]"]	Meta description on product page
textarea[name=short_description]	Product short description
textarea[name=description]	Product description
textarea[id=reply]	Ticket reply
[name="name"]	Blog post title

Field selector	Description
textarea[name="meta_title"]	Blog post meta title
textarea[name="meta_keywords"]	Blog post keywords
textarea[name="meta_description"]	Blog post meta description
textarea[name="short_content"]	Blog post summary

Translate to different Languages

OpenAI language model can be used for translating text to another language. You need to create a corresponding prompt in AI Assistant to use ChatGPT for translating purposes.

There are several approaches to create a prompt for ChatGPT to be used as a translator:

- When creating a Prompt, specify in which language you want to receive the response;
- Create a separate Prompt in which you ask to translate an existing text to another language (similar to how Fix Grammar prompt);
- Write the question in the language you want to receive the answer. For example, if you want to translate from English to German - write the prompt in German.

Note

When using Automation rules functionality, the language of the ChatGPT output is affected by the selected Store view. The Store View language will signal to ChatGPT to switch the language for its output.

Examples of prompts

Prompt for Meta Title

URL Key
[store view]



Create Permanent Redirect for old URL



Meta Title
[store view]

Meta Keywords
[store view]



Meta Description
[store view]

Maximum 255 chars. Meta Description should optimally be between 150-160 characters

Related Products, Up-Sells, and Cross-Sells

Code	meta_title
Title	Meta Title
Prompt	Write a meta title on the following product information: {{ global.input }} Product attributes: {{ product.attributes }}
Scopes	Product
Field Selectors	[name="product[meta_title]"]
Help Text	unique selling point
Convert output to HTML	No

Prompt for Meta Keywords

URL Key
[store view]

Create Permanent Redirect for old URL

Meta Title
[store view]

Meta Keywords
[store view]

Meta Description
[store view]

Maximum 255 chars. Meta Description should optimally be between 150-160 characters

Related Products, Up-Sells, and Cross-Sells

Code	meta_keywords
Title	Meta Keywords
Prompt	Write a meta keywords on the following product information: {{ global.input }} Product attributes: {{ product.attributes }}
Scopes	Product
Field Selectors	[name="product[meta_keyword]"]
Help Text	unique selling point
Convert output to HTML	No

Prompt for Meta Description

URL Key
[store view]



Create Permanent Redirect for old URL



Meta Title

Meta Title
[store view]

Meta Keywords
[store view]



Meta Description
[store view]

Maximum 255 chars. Meta Description should optimally be between 150-160 characters

Related Products, Up-Sells, and Cross-Sells

Code meta_description

Title Meta Description

Write a meta description on the following product information:

Prompt {{ global.input }}

Product attributes:

{{ product.attributes }}

Scopes Product

Field Selectors [name="product[meta_description]"]

- unique selling point 1

Help Text - unique selling point 2

- unique selling point 3

Convert output to HTML No

Prompt for Product Description

Code description

Title Description

Write a long marketing product description by the plan:

- Introduction (don't print this as a header)
- Benefits
- Key features
- Feature
- Feature
- Feature
- Feature
- Feature
- Feature
- Summary

Prompt

Use the following product information:

{{ global.input }}

Product attributes:

{{ product.attributes }}

Scopes

Product

Field Selectors

textarea[name=description]

- unique selling point 1

Help Text

- unique selling point 2

- unique selling point 3

Convert output to HTML Yes

Prompt for Product Short Description

Short description is a block of text with all the major info on a product. This block is displayed when the product is displayed beyond its product page.

Content 

Short Description
[store view]

Show / Hide Editor



Code	short_description
Title	Short Description
Prompt	<p>Write a short marketing product description using the following product information:</p> <p>{{ global.input }}</p> <p>Product attributes:</p> <p>{{ product.attributes }}</p>
Scopes	Product
Field Selectors	textarea[name=short_description]
Help Text	<ul style="list-style-type: none">- unique selling point 1- unique selling point 2- unique selling point 3
Convert output to HTML	No

Prompt for Ticket response message

To

Public Reply ▾

Your message will be emailed to the customer

ann@example.com

Template

Customer Summary

Assigned To Customer

Ann Peterson 

Cc 

Bcc 

Assigned To Order

Customer does not have any orders.

Katie Couris added 6 days ago (Feb 17, 2023 7:35 AM)

some internal message here...

Code ticket
Title Message

Customer Name: {{ ticket.customer }}

There is a last message from customer:

...

{{ ticket.lastMessage }}

...

Prompt Write a personal message that explain next things:

...

{{ global.input }}

...

Message:

Scopes Ticket
Field Selectors textarea[id=reply]

- key point 1

Help Text - key point 2

- key point 3

Convert output to HTML No

Prompt for more friendly reply message

The screenshot shows a ticket management interface. At the top, there is a header with a ticket ID "[#SUS-980-24985] What is the status ..." and navigation links for "Back" and "Create New Ticket". Below the header, there is a sidebar on the left and a main content area on the right. The sidebar contains a text input field with "ann@example.com", a "Customer Summary" section, and fields for "Assigned To Customer" (Ann Peterson), "Cc", "Bcc", and "Assigned To Order" (Customer does not have any orders). The main content area shows a "Template" section with a message body: "Hi Ann, Thank you for your inquiry. We hope we have been able to answer all of with the information that you needed. If there is anything else we can d reach out." Below the message body, there is a redacted area with the text "some internal message here..." and a timestamp "Katie Couris added 6 days ago (Feb 17, 2023 7:35 AM)".

Code friendly
Title Friendly

Make the following message more sociable and friendly:

...

Prompt {{ global.input }}

...

Message:

Scopes Ticket
Field Selectors textarea[id=reply]

Help Text N/A
Convert output to HTML No

Prompt for grammar fix

[#SUS-980-24985] What is the status ...

← Back

Create New Ticket

ann@example.com

Customer Summary

Assigned To Customer

Ann Peterson

Cc

Bcc

Assigned To Order

Customer does not have any orders.

Template

Hi Ann,

Thank you for your inquiry. We hope we were able to answer all of your information that you need. If there is anything else we can do to help, p

Thanks again for your interest in our products and services!

Katie Couris added 6 days ago (Feb 17, 2023 7:35 AM)

some internal message here...

Code grammar
Title Fix Grammar

Fix grammar in the following message. Keep all original line breaks.

...

Prompt {{ global.input }}

...

Message:

Scopes Ticket
Field Selectors textarea[id=reply]
Help Text N/A
Convert output to HTML No

Prompt for blog title

☰ How to choose the right blog solution for Magento 2 store marketing ▾

Preview

← Back

Delete

Save and Continue

Status *

Published ▾

Published on *

02/20/2023 6:55 AM 

Title *

How to choose the right blog solution for Magento 2 store marketing

Content

A major marketing channel for any online store is creating additional

Code	blog_post_title
Title	Title
	There is a blog post:
	...
Prompt	{{ post.content }}
	...
	Write a title:
Scopes	Blog post
Field Selectors	[name="name"]
Help Text	N/A
Convert output to HTML	No

Prompt for blog meta title

How to choose the right blog solutio...

Preview

← Back

Delete

Save

URL Key

how-to-choose-the-right-blog-solution-for-magento-2-store-mar

Meta Title

Get the most marketing results out of your current and future bl

Meta Keywords

marketing, extension, eCommerce, business, store

Code	blog_post_meta_title
Title	Meta Title
	There is a blog post:
	...
Prompt	{{ post.title }} {{ post.content }}
	...
	Write a meta title:
Scopes	Blog post
Field Selectors	textarea[name="meta_title"]
Help Text	N/A
Convert output to HTML	No

Prompt for blog meta keywords

How to choose the right blog solutio...

Preview

← Back

Delete

Save

Meta Keywords

marketing, extension, eCommerce, business, store

Meta Description

Blog is a major marketing channel for any online store that utilizes social media to explain the brand and products the store sells. There are several ways to choose the right meta description when choosing them to suit the...

Related Products

Code	blog_post_meta_keywords
Title	Meta Keywords
Prompt	<p>There is a blog post:</p> <p>...</p> <p>{{ post.title }}</p> <p>{{ post.content }}</p> <p>...</p> <p>Write meta keywords:</p>
Scopes	Blog post
Field Selectors	textarea[name="meta_keywords"]
Help Text	N/A
Convert output to HTML	No

Prompt for blog meta description

How to choose the right blog solutio...

Preview

← Back

Delete

Save

Meta Description

Looking for a blog solution for your Magento 2 store? Compare the Blog Extension versus a standalone Wordpress to find the right one for your business. Learn about installation, integration, features set and their potential, customers registration differences and more.

Related Products

Related Posts

Code	blog_post_meta_description
Title	Meta Description
Prompt	There is a blog post: ... {{ post.title }} {{ post.content }} ... Write a meta description:
Scopes	Blog post
Field Selectors	textarea[name="meta_description"]
Help Text	N/A
Convert output to HTML	No

Prompt for blog post summary

How to choose the right blog solutio...

Preview

← Back

Delete

Sav

Excerpt



Search Engine Optimization

Related Products

Related Posts

Code	blog_post_summary
Title	Summary
	There is a blog post:
	...
Prompt	{{ post.title }} {{ post.content }}
	...
	Write a summary:
Scopes	Blog post
Field Selectors	textarea[name="short_content"]
Help Text	N/A
Convert output to HTML	No

Prompt for tranlating a text

Prompt "Translate blog article"

← Back

Delete

Duplicate

General Information

Active Yes

Code *

Title *

Prompt *

Frequency penalty *

from 0 to 1 (e.g. 0.3)

Scopes

Field Selectors *

Code	translate_blog
Title	Translate blog article
	Translate the following blog post from English to German. Keep all original line breaks.
	...
Prompt	{{ post.content }}
	...
	Post:
Scopes	Blog
Field Selectors	.blog-post-edit textarea[name="html"], .blog-post-edit textarea[name="content"]
Help Text	Translate the following blog post from English to German.
Convert output to HTML	Yes

Automation rules

Automation rules functionality allows to run the selected prompts by Cron.

Navigate to **System > AI Assistant > Automation rules** and add a new rule.

Automation rule configurations are split in 2 stages.

On the **first stage** you'll need to set the following information for the rule (General Information section):

- **Title** - The name of the rule that is visible in the grid of automation rules.
- **Entity Type** - type of an entity to configure the rule for. Currently the extension allows to create rules for Products and Categories.
- **Active** - set to Yes if the rule should be executed.
- **Apply Only Once** - set to Yes if the rule should be executed only one time.
- **Overwrite Not Empty Field** - set to Yes if the rule should overwrite any existing text in the selected text field.
- **Store Views** - select the applicable scope of the rule.

☰ New Automation Rule ▾

General Information

Title *

Entity Type *

Select...

Active No

Apply Only Once No

Overwrite Not Empty Field No

Store View *

All Store Views

Main Website

Main Website Store

Default Store View

Second Store View

Logs

After specifying all data press the **Save and Continue** button to proceed to the second stage.

Note

You won't be able to change the entity type after you saved the rule.

On the **second stage** you'll need to set additional information:

- **Prompt** - select the existing prompt that should be used with this automation rule. For example, a prompt for generating meta title.
- **Field** - select one field that will be used for inserting the prompt output. For example, a *Product Meta Title* field.
- **Conditions** - select conditions to adjust entities for which the rule should be applied. Leave blank if the rule should be applied for all entities of the type selected at the stage 1.



To apply this rule please use the following command. You can add it in Cron if necessary.

```
bin/magento mirasvit:assistant:apply-rule --id 1
```

General Information

Title *

Entity Type *

Active Yes

Apply Only Once No

Overwrite Not Empty Field Yes

Store View *

- All Store Views
- Main Website**
- Main Website Store**
- Default Store View
- Second Store View

Configuration

Prompt

Field

Conditions (leave blank for all products)

After specifying all data save the rule.

Write down its ID as it will be used for configuring Cron job. You will need to specify the rule ID in Cron command `bin/magento mirasvit:assistant:apply-rule --id XXX`, where XXX is the rule ID number.

For more details regarding this command check the [Command Line Interface](#) section of this manual.

Note

The GPT/ChatGPT will generate answers on the language depending on the selected Store View.

Command Line Interface

Usage: `php -f bin/magento [options]`

- `mirasvit:assistant:apply-rule` - Apply automation rule for products that match rule's conditions. This command has the following params:
 - `--id` - id of applied rule. This parametr is required.
 - `--entity-id` - allows you to specify only one entity id (e.g. `product_id`) that will be changed by the rule. This parameter is optional.
 - `--reset` - reset already applied rule so it can be re-applied again later. Will affect only rules with "*Apply Only Once*" => "*Yes*". This parameter can be combined with the parameter `--entity-id`. Not applicable along with the `--dry-run` parameter.
 - `--dry-run` - this is a flag which indicates that you would like to run the command without changing anything in database.
 - `--force` - forcibly apply rule (will terminate previous process for current rule if running).

Note

Examples:

- `mirasvit:assistant:apply-rule --id 1` - applies the rule with ID 1 to all products that match conditions in the rule
 - `mirasvit:assistant:apply-rule --id 1 --entity_id 10` - applies the rule with ID 1 to the entity (e.g. product) with ID 10 **ONLY** if that product matches conditions of the rule
 - `mirasvit:assistant:apply-rule --id 1 --reset` - resets the rule with ID 1 **ONLY** if the rule's option "*Apply Only Once*" is set to "*Yes*"
- `mirasvit:assistant:reindex` - Reindexes the "automation rule - product" relations. This command has only one optional parameter `--reset` which can be used to remove all data from the "automation rule - product" relations table

How to upgrade extension

To upgrade the extension, follow these steps:

1. Backup your store's database and web directory.
2. Login to the SSH console of your server and navigate to the root directory of the Magento 2 store.
3. Run command `composer require mirasvit/module-assistant:* --update-with-dependencies` to update current extension with all dependencies.

Note

In some cases, the command above is not applicable, or it's not possible to update just the current module, or you need to upgrade all Mirasvit modules in a bundle. In this case, the command above will have no effect.

Run instead `composer update mirasvit/*` command. It will update all Mirasvit modules installed in your store.

4. Run command `php -f bin/magento setup:upgrade` to install updates.
5. Run command `php -f bin/magento cache:clean` to clean the cache.
6. Deploy static view files

```
rm -rf pub/static/*; rm -rf var/view_preprocessed/*; php -f bin/magento setup:static-content:deploy
```

Disabling the Extension

Temporarily Disable

To temporarily disable the extension please follow these steps:

1. Login to the SSH console on your server and navigate to the root directory of the Magento 2 store.
2. Run the command `php -f bin/magento module:disable Mirasvit_Assistant` to disable the extension.
3. Log in to the Magento backend and refresh the store cache (if enabled).

Extension Removal

To uninstall the extension, please follow these steps:

1. Login to the SSH console on your server and navigate to the root directory of the Magento 2 store.
2. Run command `php -f bin/magento module:disable Mirasvit_Assistant` to disable the extension.
3. Run command `composer remove mirasvit/module-assistant` to remove the extension.
4. Log in to the Magento backend and refresh the store cache (if enabled).

Change Log

1.4.10

(2024-10-30)

Improvements

- PageBuilder compatibility improved

Fixed

- Fixed the issue with unable to write anything in newly added fields (PageBuilder)
-

1.4.9

(2024-08-09)

Fixed

- Fixed the issue with the rule applied to all stores instead of the default store when the rule is configured to apply for All Store Views
-

1.4.8

(2024-07-30)

Fixed

- Fixed the issue with incorrectly applied generated content in PageBuilder (partial text selection)
-

1.4.7

(2024-07-25)

Fixed

- Fixed the conflict with Mageworx Advanced Product Options Suite
 - Fixed the issue with error when 'Remove HTML tags' is disabled (Magefan Blog)
-

1.4.6

(2024-07-24)

Improvements

- Added support for OpenAI model GPT-4o-mini
-

1.4.5

(2024-07-22)

Fixed

- Fixed the issue with Magefan Blog Posts compatibility
-

1.4.4

(2024-07-04)

Fixed

- Fixed the issue with assistant not appeared on CMS pages (since 1.4.0)
-

1.4.3

(2024-06-26)

Fixed

- Fixed the issue with applying incomplete responses from OpenAI API
-

1.4.2

(2024-05-24)

Fixed

- Fixed the issue with the error "Request failed with status code 500"
-

1.4.1

(2024-05-14)

Improvements

- Added support for GPT-4o model
-

1.4.0

(2024-05-07)

Improvements

- Ability to select if HTML tags should be removed from the text in the prompt or not
 - Ability to choose the store from which the extension will retrieve variables' values for prompts
-

1.3.13

(2024-04-30)

Fixed

- Fixed the issue with cleaning content created in the page builder
 - Fixed the issue with error during applying rules related to the attribute 'custom_layout_update_file'
-

1.3.12

(2024-03-11)

Features

- Compatibility with Mirasvit_Brand (prompts)
-

1.3.11

(2024-02-27)

Improvements

- Added support for the GPT-4 Turbo model
-

1.3.10

(2023-12-29)

Improvements

- Page Builder support for pre-installed prompts for category description and product description
-

1.3.9

(2023-10-30)

Fixed

- Fixed the issue with saving alt for product images per store (automation rules)
-

1.3.8

(2023-10-27)

Fixed

- Fixed the issue with the generated product image alt attribute not being saved (prompt)
-

1.3.7

(2023-10-18)

Improvements

- Ability to generate Products images' ALT attributes automatically (automation rules)
-

1.3.6

(2023-10-05)

Improvements

- Added context for Mirasvit Knowledge Base Categories and Articles
-

1.3.5

(2023-09-28)

Fixed

- Fixed the conflict with Mirasvit_Kb (Categories)
-

1.3.4

(2023-09-25)

Fixed

- Fixed the issue with WYSIWYG in Magento 2.3.*
-

1.3.3

(2023-09-08)

Fixed

- Fixed the issue with attribute values from the default (admin) store passed to prompts in automation rules
-

1.3.2

(2023-08-28)

Fixed

- Fixed the issue with the rule execution terminated when error happened during saving entity's attribute
-

1.3.1

(2023-08-25)

Fixed

- Fixed the issue with prompt inserted instead of answer on Apply button click
-

1.3.0

(2023-08-25)

Improvements

- GPT-4 model added
 - Ability to set different OpenAI models for different prompts
 - Ability to choose the OpenAI model in the assistant popup
-

1.2.3

(2023-07-26)

Improvements

- Updated UI library
-

1.2.2

(2023-07-04)

Fixed

- Fixed the issue with foreign key error on setup:upgrade (since 1.2.1)
-

1.2.1

(2023-07-03)

Improvements

- Automation rules for categories
-

1.2.0

(2023-06-08)

Fixed

- Backend issue (conflict with Mirasvit_CatalogLabel)
-

1.1.11

(2023-06-05)

Fixed

- Fixed the issue with saving product when automation rule is present with deleted attribute in conditions
 - Fixed the issue with error on CMS Page edit page
-

1.1.10

(2023-05-31)

Improvements

- Support for Magefan Blog
-

1.1.9

(2023-05-16)

Fixed

- Fixed the issue with incorrect Assistant API URL (Magento < 2.4.4)

1.1.8

(2023-04-27)

Features

- Logging for automation rules

Fixed

- Issue with creating new CMS page
 - Cron issue
-

1.1.7

(2023-04-25)

Improvements

- Prevent executing multiple apply-rule commands for the same rule at the same time

Fixed

- Properly retrieve product data per storeview (multistore)
-

1.1.6

(2023-04-21)

Fixed

- Fixed the issue with analyzing partial select inside WYSIWYG editor
-

1.1.5

(2023-04-20)

Improvements

- Data preparation for request to API improved
 - Compatibility with pagebuilder
-

1.1.4

(2023-04-18)

Improvements

- Compatibility with pagebuilder

Fixed

- Issue with the 'Undo' button when only part of the text analyzed by Assistant
-

1.1.3

(2023-04-17)

Improvements

- Compatibility with WYSIWYG
 - AI answer characters count for meta fields
-

1.1.2

(2023-04-13)

Fixed

- Prevent results like 'No correction needed' on Fix Grammar
 - The issue with errors on product edit pages (attributes added by 3rd-party extensions)
 - The issue with not able to type 'space' in prompts
 - The issue with assistant bar in helpdesk
 - The issue with automation rules in multistore
-

1.1.1

(2023-04-11)

Fixed

- Issue with indexing products per store (all stores)
 - Issue with deleting products
-

1.1.0

(2023-04-05)

Improvements

- Added rule-product index table and the command to reindex rule-product relations
- Performance of the applying rule process
- Added "reset" param to the "apply-rule" command
- Navigation through extension's pages in the backend

Fixed

- Categories show empty pages in the backend when they have Custom Layout Update
 - Stock status condition issue
 - Convert to HTML not working in CLI
 - Issue with the "Apply Only Once" option of automation rules
-

1.0.17

(2023-03-28)

Fixed

- Unable to save field selector
- PHP 8.2 - strpos(): Argument #1 (\$haystack) must be of type string, null given

1.0.15

(2023-03-22)

Features

- Add prompts for categories

Improvements

- Add entity-id param to console
 - Add dry-run and stop_sequences
-

1.0.12

(2023-03-21)

Improvements

- Add support of storeviews for automation rules
-

1.0.11

(2023-03-20)

Features

- Added more fields to Automation Rules

Fixed

- PHP 7.2
-

1.0.10

(2023-03-17)

Features

- Add support of GPT3.5
 - Added Automation Rules
-

1.0.8

(2023-03-15)

Improvements

- Rename prompt codes

Fixed

- Make field selectors field multiline
-

1.0.7

(2023-03-06)

Improvements

- Added more precise selectors for prompts

1.0.5

(2023-02-22)

- Initial release