# Training the machine learning model

In IBM Watson™ Knowledge Studio , the creation of the machine learning model involves training the machine learning model and evaluating how well the model performed when annotating test data and blind data.

## Creating a machine learning model

When you create a machine learning model, you select the document sets that you want to use to train the model and specify the percentage of documents that are to be used as training data, test data, and blind data.

### About this task

By exploring the performance metrics, you can identify ways to improve the model's accuracy.

**Restriction:** Only three machine learning models can be trained at a time per Knowledge Studio instance. If your instance contains multiple workspaces and the number of machine learning models that are being trained in other workspaces totals 3 already, then your request to train the machine learning model in your workspace will be queued until the other training processes are done.

### Procedure

To create a machine learning model:

1. Log in as a Knowledge Studio administrator and select your workspace.
2. Select **Machine Learning Model** > **Performance**.
3. Verify that all of the document sets have been approved and that all annotation conflicts have been resolved through adjudication. Only documents that have become ground truth through adjudication or approval can be used to train the model.
4. Click **Train and evaluate**.
5. Optional: To specify how you want to allocate documents from your document sets to be used by the system-level training, test, or blind sets, click **Edit settings**.

See [Document set management](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-improve-ml#wks_mamanagedata) for help determining which ratios to apply.

1. Click **Train** to train the model, or click **Train & Evaluate** to train the model, evaluate annotations added by the machine learning model, and analyze the performance statistics.
2. Select the document sets that you want to use for training the model.
3. After the model is created, select one of the following actions:

**Evaluating annotations added by the model**

You can compare the ground truth view for annotations added by human annotators to the annotations added by the model.

Procedure

To evaluate the annotations added by the model:

1. Select **Machine Learning Model** > **Performance** > **Train and evaluate**. The Training/Test/Blind Sets page is displayed.
2. Click **View Ground Truth** for the training set or test set to see the annotations that were added through pre-annotation and by human annotators. The ground truth editor opens. Click to open individual documents and see how the mentions, relations, and coreferenced mentions were annotated.
3. On the **Performance** page, click **View Decoding Results** to see the annotations that the machine learning model added to documents in the test set. This button is available only after you evaluate the model. By viewing results, you can see how well the machine learning model labeled mentions, relations, and coreferenced mentions in the test data.
4. If you want to change how the documents are divided between training, test, and blind data sets, click **Performance** > **Train and evaluate** > **Edit Settings**. For example, if initial results seem acceptable, you might want to increase the number of documents in the test set to further test the machine learning model's results. You can change the ratio for how documents are automatically divided for different purposes, or you can select specific document sets to use as training data, test data, and blind data.
5. If you made any changes, click **Train & Evaluate** to retrain the model and re-evaluate the annotations.

**Deleting a machine learning model**

You cannot delete a machine learning model.

You can delete the workspace that was used to develop the model, but you cannot delete the model itself. Deleting a model is not the best approach. Instead, update or replace the artifacts that are used to train the model. Even if the model is not producing the results you expect, you can continue to refine it. Each time you create a new version, the model is built anew. You can edit artifacts like dictionaries and the type system, and choose to use different annotation sets when you train the next version.

**About this task**

You can analyze performance by viewing a summary of statistics for entity types, relation types, and coreferenced mentions. You can also analyze statistics that are presented in a *confusion matrix*. The confusion matrix helps you compare the annotations added by the machine learning model to the annotations in ground truth.

The model statistics provide the following metrics:

* **F1 score**

A measurement that considers both precision and recall to compute the score. The F1 score can be interpreted as a weighted average of the precision and recall values, where an F1 score reaches its best value at 1 and worst value at 0. See [Analyzing low F1 scores](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-evaluate-ml#evaluate-mllowf1).

* **Precision**

A measurement that specifies what fraction of the machine learning model's output was accurate when compared to the human annotator output. Precision is determined by the number of correctly labeled annotations divided by the total number of annotations added by the machine learning model. A precision score of 1.0 for entity type A means that every mention that was labeled as entity type A does indeed belong to that classification.

A measurement that specifies how many mentions that should have been annotated by a given label were actually annotated with that label - the *right* mentions being those that human annotators identified in the same documents. Recall is determined by the number of correctly labeled annotations divided by the number of annotations that should have been created. A recall score of 1.0 means that every mention that should have been labeled as entity type A was labeled correctly..

* **Percentage of total annotations**

A measurement of ground truth that shows how many words were annotated with a given entity type or relation type out of the total number of words that were annotated as any entity type or relation type in the test document set. This statistic is not available for coreferenced mentions. This value can help you to see how prevalent mentions of one type are compared to the other types in your ground truth.

* **Percentage of corpus density (by the number of words)**

A measurement of ground truth that shows the number of words that were annotated with a given entity type or relation type out of the total number of words, whether annotated or unannotated. This statistic is not available for coreferenced mentions. This value can help you to see how prevalent mentions of this type are compared to all of the other words in your domain documents.

* **Percentage of documents that contain the type**

A measurement of ground truth that shows how many documents contain a given entity type or relation type. This statistic is not available for coreferenced mentions. This value can help you to assess whether the documents in the set represent the domain sufficiently. If the percentage is low for key entity types, then you might want to add more documents with mentions of under-represented types.

**Procedure**

To view performance statistics for how well the model was trained:

1. Log in as a Knowledge Studio administrator or project manager, and select your workspace.
2. Select **Machine Learning Model** > **Performance**.
3. For the mentions, relations, or coreferences, select the **Detailed Statistics** link.
4. In the **Summary** view, specify whether you want to evaluate test data or training data, and then specify the type of annotations you want to see statistics for: entity types, relation types, or coreferenced mentions. As you scroll through the data, items that have low scores are flagged and highlighted to indicate that they require investigation and improvement. The triangle warning icon indicates that the F1 value is less than the fixed value, 0.5.

For example, the F1 score for some entity types might be high because the document was annotated through pre-annotation as well as by a human annotator. But the F1 score for other entity types might be low because differences in phrasing, and differences in how human annotators interpret the text or annotation guidelines, make it more difficult for the machine learning model to recognize the pattern and apply the correct annotation.

1. In the **Confusion Matrix** view for test data, specify the type of annotations that you want to see statistics for: entity types or relation types. For each entity type or relation type:
   * Each row in the matrix shows ground truth (entity types and relation types added by a human annotator).
   * Each column in the matrix shows the decoding results (annotation tokens added by the machine learning model).
   * The numbers in each cell represent the number of annotation tokens per mention, not the occurrences of the mention.

Tokens are used to delineate text strings. They loosely correspond to words. A mention can contain more than one token. For example, *Barack Obama* is a mention that contains two tokens. In the PERSON row or column, a value of 2 would be used to represent the single mention of *Barack Obama* in a document. The token count only loosely corresponds to the word count because tokens are counted differently in some situations. For example, punctuation at the end of a sentence is counted as a token, and contractions are often expanded into two tokens.

* + The column labeled **O** identifies tokens that should have been annotated by the machine learning model as the type that is identified in the intersecting row (the human annotation), but they are not annotated as any type.
  + The value **N/A** is shown when there are no annotations for a given type in the document sets. For example, if there is no PERSON mention in the document sets that were annotated for ground truth or as test data, the scores for the PERSON entity type is N/A.

For example, the following confusion matrix example shows results of a machine learning model run on documents that deal with traffic incidents.

Table 1. Confusion matrix example

| **Entity types** | **MANUFACTURER** | **MODEL** | **O** |
| --- | --- | --- | --- |
| MANUFACTURER | 515 | 5 | 44 |

You can learn the following things from this matrix:

* + The model correctly recognized 515 tokens as mentions of the MANUFACTURER entity type.
  + The model incorrectly labeled 5 tokens as mentions of the MODEL type that should have been labeled as MANUFACTURER.
  + There were 44 tokens that comprise mentions of the MANUFACTURER type that the model failed to annotate as any entity type.

When you evaluate model performance and view the decoding results, you can start by investigating the tokens that were mislabeled or missed completely.

**Performance improvement cheat sheet**

Use this cheat sheet to help you determine the steps you can take to improve the machine learning model performance.

It's difficult to dictate rules for improving performance scores that would be applicable across domains, where type system complexity, appropriateness of training documents, human annotator skills, and other factors influence the outcome. However, the following table can help you make an initial assessment and take steps to improve performance, especially during the early stages of model development and testing.

The following table suggests fixes for the most common machine learning model performance problems.

# Making machine learning model improvements

After you determine areas in which the model is having trouble, take steps to improve its performance.

## Creating model versions

After you create a machine learning model, you can take a snapshot to keep a backup version of the current resources in case you want to restore the resources in a future iteration.

### About this task

The F1 score provides an indication of the quality of the model. If the model performance results are good, you might want to store a version of the component before changing any of the resources. If changes that you make result in poorer quality, you can revert to a version that you stored. When you revert to an older version, all annotation tasks are archived because they are no longer valid.

You can have a maximum of 10 versions of a workspace. If you reach that limit, delete older versions or versions that you no longer need before creating a new version.

When you create a new version, the following resources are captured:

* Type system
* Corpus
* Ground truth
* Machine learning model
* Machine learning model evaluation results

The following resources are excluded:

* Annotation tasks, because they are temporal by design, used only for determining ground truth
* Dictionaries, because dictionaries can be large, and various types of dictionaries are managed in different ways

### Procedure

To create and restore machine learning model versions:

1. Log in as a Knowledge Studio administrator or project manager, and select your workspace.
2. Select **Machine Learning Model** > **Performance**. Performance statistics about the current version, labeled version 1.0, are displayed.
3. To take a snapshot of the current version, click **Machine Learning Model** > **Versions**, and then click **Take Snapshot**. The resources in version 1.0 are frozen, and a new version, labeled 1.1, becomes the current version. For each new version that you create, the minor version number is incremented, for example, 1.0 becomes 1.1 and then becomes 1.2.
4. Revise the workspace resources as needed, re-train, and re-evaluate the model.
5. If you are pleased with the performance results and want to store the new version before making future changes, create another version. Continue revising resources and re-training the model as needed, creating a new version for each iteration that you want to retain.
6. If performance results are worse, and you want to revert to a previous version before testing any further:
   1. Open the **Assets** > **Dictionaries** page and download any dictionaries that you want to re-use in the restored model.
   2. Click **Machine Learning Model** > **Versions** and click **Promote** for the version that you want to restore. The version that you promote becomes the current version, and the version number changes to 2.0. When you promote a version, the major version number is incremented and the minor version number becomes 0, for example, 1.1 becomes 2.0.
   3. Open the **Dictionaries** page and upload the dictionaries that you downloaded.
   4. If testing of the new version requires changes to ground truth, open the **Machine Learning Model** > **Annotations** page. Click the **Annotation Tasks** tab and create a new annotation task.

## Modifying a type system without losing human annotations

You might need to make modifications while you train a model, based on the performance statistics. But, generally, you want the type system to be as close to final as possible before you begin large-scale annotation tasks. If you change the type system after human annotators began their work, they must revisit the documents that they annotated. They must assess the applicability of the type system changes.

### About this task

This process propagates the current type system, ground truth editor keyboard shortcuts, and color settings to all document sets in a task.

### Procedure

To modify the type system without losing the work that was done by human annotators:

1. Change the type system. For example, you can add or remove entity types or relation types.
2. Decide whether you want to propagate the changes to existing human annotation tasks.
3. Open the **Machine Learning Model** > **Annotations** page and click the **Annotation Tasks** tab. Open each task that you want to update and click **Apply Type System Updates**.

If you removed entity types or relation types from the type system, all occurrences of those types are highlighted in gray in the documents. These invalid types are ignored by the machine learning model. They do not prevent you from submitting and approving document sets.

1. Provide details to the human annotators about what changed in the type system.
2. Ask human annotators to update their documents to reflect the changes in the type system. For example, if you added new entity types or relation types, they must review their documents and annotate them appropriately.

# Managing advanced rules projects and extractors

In the advanced rules editor, use projects to organize your extractors and sample documents.

## Before you begin

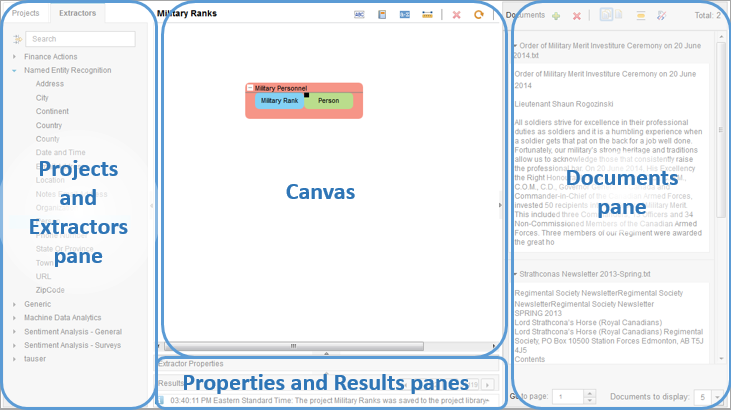
Advanced rules workspaces are available only in Knowledge Studio instances hosted in the **Dallas** or **Frankfurt** locations.

1. [Create an instance of Knowledge Studio](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-wks_tutintro#instance) in the **Dallas** or **Frankfurt** location.
2. From the **Manage** page of your Knowledge Studio service instance, click **Launch tool**.
3. Create an advanced rules workspace.
   * If you already have other workspaces, click **Create workspace**.
   * Click **Create advanced rules workspace**. Enter a name for your workspace, then click **Create**. The advanced rules workspace will load after a few seconds.

## Workspace layout

The workspace consists of the following areas:

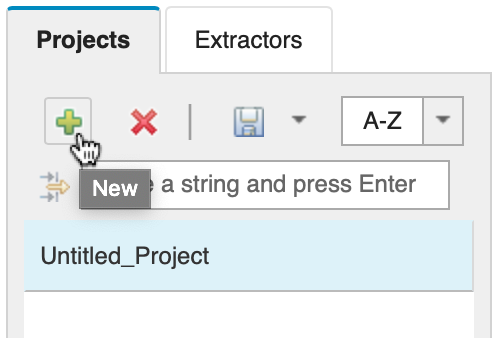
* The **Projects and Extractors pane** includes two tabs. Use the **Projects** tab to create, delete, and rename projects. The **Extractors** tab lists the provided extractors and any custom extractors that you create.
* Use the **Canvas** to design and customize extractors.
* Use the **Extractor Properties** and **Results** panes to set extractor properties and to view the results of running an extractor on sample text
* Use the **Documents** pane to manage sample documents that you can use to test the extractors.



## Managing projects

Use projects to develop and test extractors that support a specific solution or category of documents. Projects are saved automatically, at five minute intervals, as you work on them. You can save a project manually by clicking **Save** .

* To create a project, click **New**  on the Projects tab.
* To delete a project, select the project and click **Delete**  on the Projects tab.
* To rename a project, double-click it and edit the project name.



## Managing extractors

An extractor is a software component that performs specific analysis tasks and provides the analysis logic. Sample extractors that you can use as they are or copy and edit are provided. You can also define extractors to match specific terms, character patterns, and sequences and save them to the extractor library. Use the workspace to add, modify, combine, share, and delete extractors.

### Adding an extractor to the canvas

1. Click the **Extractors** tab in the **Projects and Extractors** pane.
   * Expand the appropriate folder and select the name of the desired extractor. If you want to see more information about an extractor, select it and then see the description information under **Extractor Properties**. You can also hover over the **Examples** link in the **Extractor Properties** pane to see examples of the results returned by the extractor.
   * Alternatively, search for the extractor by name.
2. Drag and drop the selected extractor to the canvas. When an extractor is added multiple times, all instances are treated as one. For example, a changed property in one instance is reflected in all instances. To prevent changing all instances, rename each instance when dragging it to the canvas. If you have renamed two or more instances, you can right-click and select **Unlink** to create a copy.

You can refine the extractor by modifying the **Settings** and **Output** values in the **Extractor Properties** pane as needed. See [Creating a dictionary](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-dictionary) and [Running an extractor](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#running-an-extractor) for more information about how to configure these properties. Changing any extractor property automatically creates a copy for use in your project.

### Running a category of extractors

You can also add a category of extractors from the extractor library at one time. Any extractors in the category or folder with matches in the input documents are added automatically to the canvas.

1. Click the **Extractors** tab in the **Projects and Extractors** pane.
2. Right-click the folder for the category.
3. Click **Run Category.**

### Saving extractors to the extractor library

Until an extractor is saved to the extractor library, you can access the extractor only in the project where it is defined.

To make an extractor available to all projects, click **Save Extractor** on the toolbar for the canvas. Custom extractors are saved on the **Extractors** tab in the tauser folder by default. As a best practice, save modified versions of a provided extractor in the same folder as the provided extractor.

## Creating an extractor

You can build an extractor by combining one or more of the following elements.

* [Literal](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-literal): matches a single term such as white-tailed deer or Congress
* [Dictionary](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-dictionary): matches a group of terms that belong to the same category
* [Sequence](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-sequence-pattern): a combination of extractors that matches a sequential pattern of terms
* [Union](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-union): a grouping of extractors that represent a single concept
* [Regular expressions](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-regular-expression): syntax-based pattern matching

### Creating extractors with linguistic patterns

Defining a new extractor using natural language processing (NLP) concepts such as tokenization and part-of-speech requires developing an extractor using the Annotation Query Language (AQL).

For more information about how to write AQL and available functions, see [Annotation Query Language reference](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-annotation-query-language-reference)

### Creating a dictionary

A dictionary is a list of terms that relate to a concept. Use a dictionary as a convenient substitute for individual literals when analyzing text for multiple terms that fit into the same category.

You can define a dictionary as a simple list of terms or as a mapping table that maps a value or term to a preferred term. Mapping tables are typically used to resolve synonyms, abbreviations, and encoded values.

Use of a dictionary can sometimes result in overlapping matches. For example, a dictionary used to extract information about people with a military rank such as Officer, Warrant Officer, or Chief Warrant Officer might result in three overlapping matches for every occurrence of Chief Warrant Officer. See [Eliminate duplicate and overlapping results](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#eliminate-duplicate-and-overlapping-results) for information needed to address this issue.

#### Defining a list

When using standardized terms such as names of departments in your organization, sales priorities, congressional committees, or military ranks to extract values, you can identify the needed terms in a list.

1. Click  on the canvas toolbar.
2. Specify a name for the dictionary. You can extend the name with a description. Lengthy text might be truncated in the display but you can hover over the dictionary to read the full description.
3. On the **Settings** tab under **Extractor Properties**, enter each term manually or load a list from a file. The maximum length for each entry is 1024 characters.
   * To specify the terms manually, click , enter the term, and press **Enter**. Enter one term per row.
   * To load terms from a UTF-8 text file with each term on a separate line, . If an error is displayed during the import, ensure that the file does not contain unprintable characters such as Line Feed or Backspace.

#### Defining a mapping table

When analyzing text that contains synonyms, alternate spellings for a term, or encoded values, you can define a mapping table to map terms in the text to a desired term.

A mapping table is analogous to a lookup table or a key-value pair table. Examples of instances in which you might define a mapping table include matching:

* Nicknames like Maggy and Peggy for Margaret
* Abbreviations and alternate forms of an organization name such as I.B.M., IBM Deutschland and International Business Machines for the commonly used name IBM
* 404 to File Not Found when translating HTML error codes in a log file
* The area code 408 to California

For best performance, use long or complex mapping tables in a second step, after the text extraction step.

1. Click New Dictionary on the canvas toolbar.
2. Specify a name for the dictionary. You can extend the name with a description. Lengthy text is truncated but is displayed when you hover over the dictionary.
3. On the **Settings** tab under **Extractor Properties**, click **Map Terms**.
4. Complete the two-column table manually or by loading from a file. The first column contains the terms to search for in the text and the second column lists the values to generate in the results. The maximum length for an entry is 1024 characters. If a term is repeated in the first column, only the first mapped value is used.
   * To specify the terms manually, click , enter the term and its mapped value, and press Enter.
   * To load terms from an ASCII or UTF-8 text file with each term and its mapped value, separated by a single comma, on a separate line, click . If an error is displayed during the import, ensure that the file does not contain unprintable characters such as Line Feed or Backspace.

### Creating a literal

For exact matches to a single term or phrase such as white-tailed deer or Congress use a literal.

1. Click the **New Literal**  icon on the canvas toolbar. This adds a new literal to the canvas.
2. In the new literal on the canvas, enter the string of text that you want to match. For example, white-tailed deer.

### Creating a regular expression

To extract information on the basis of the format of text, you must use a regular expression. The syntax for a regular expression is the same as that used to define regular expressions in Java.

To create an extractor that uses a regular expression:

1. From the canvas toolbar, click New Regular Expression.
2. Enter a name for the regular expression. You can also add some descriptive information. Note that the information that you enter at the prompt is the name for the expression, not the expression itself.
3. Select the regular expression on the canvas to open the **Extractor Properties** to the **Settings** tab. Specify a regular expression. For more information about the syntax, see the [Class Pattern](https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html) Javadocc.
4. Specify options for case sensitivity, token range, and special character handling. If, while designing a regular expression, you see false positives, provide more context by extending the pattern in the regular expression or adding context through a sequence pattern. For example, if your postal code expression matches additional terms, incorporate rules for which letters are permitted in specific positions.

#### Regular expression examples

The following are examples of regular Java expressions that might be used in specific instances. In the pattern descriptions, A represents a character and 9, a digit. For more information about Java syntax, see [Class Pattern](https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html) Javadoc.

* To select text that includes any ordinal number of one or more digits followed by th, st, nd, or rd (21st, 2nd, 3rd, and so forth), specify:
* \d+(st|nd|rd|th)
* To select text that includes US Social Security numbers formatted as 999-99-9999, specify:
* \d{3}\-\d{2}\-\d{4}
* To select all text that includes a Canadian postal code formatted as A9A-9A9, A9A 9A9 or A9A9A9 (for example, K1G 3K9, V5g-4X3, and x2H3m5), specify:
* [a-zA-Z][0-9][a-zA-Z](-|)[0-9][a-zA-Z][0-9]
* To select all text that includes a United Kingdom postal code formatted as A9 9AA, A99 9AA, AA99 9AA, A9A 9AA, or a9d AA9A 9AA (for example, M1 1AE, B33 8TH, DN55 1PT, W1A 0AX, CR2 6XH, and SW1A 2AA), specify:
* [A-Z]{1,2}[1-9][0-9]?[A-Z]?\s[0-9][A-Z]{2,}|GIR 0AA
* To select all text that includes number plates for private and commercial vehicles in India formatted as AA-99-AA-9999, AA-99-AAA-9999 and, for states where the 0 is optional, AA-9-AA-9999 with separators that may be hyphens, spaces or not present (for example, TN-86-AF-1199, WB 06 F 5971, and DL4CAF4943), specify:
* [A-Z]{2}(-| |)\d{1,2}(-| |)[A-Z]{1,3}(-| |)\d{4} (M1 1AE, B33 8TH, DN55 1PT, W1A 0AX, CR2 6XH, SW1A 2AA)

### Creating a sequence pattern

Often, you must assess the text for patterns that provide context for the terms of interest. For example, you might want to distinguish references to military or law enforcement personnel from references to civilians. When designing an extractor, several methods are available to identify and extract information on the basis of patterns in text. These methods can be used in a single extractor, or you can combine extractors.

Before defining your own pattern to meet your needs, check to see if a provided extractor can give you the desired results.

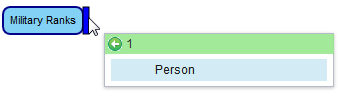
To create a sequence pattern:

1. Create individual extractors for all needed terms by extending provided extractors, or creating [dictionaries](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-dictionary), [regular expressions](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-regular-expression), [literals](https://cloud.ibm.com/docs/watson-knowledge-studio?topic=watson-knowledge-studio-managing-projects-and-extractors#creating-a-literal).
2. Drag and drop one extractor to another extractor on the canvas, aligning your cursor to reflect the order in which the term appears in the text pattern. A dark, bold blue line to the left or right of the extractor on which you are dropping the new extractor indicates the relative positions of the extractors. After you drop the new extractor, a box surrounds the two extractors to indicate the sequence. The box has a temporary title, Sequence n.
3. Optional: Select the sequence on the canvas and rename it in **Extractor Properties** under **General**.
4. Optional: If needed, repeat steps 1 and 2 to add additional elements to the pattern.

#### Sequence pattern example

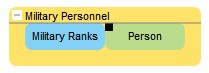
To select references to military personnel:

1. Create a dictionary called Military Ranks that includes terms such as Warrant Officer, Sergeant, and Lieutenant.
2. Drag the **Person** extractor onto the canvas following the **Military Ranks** dictionary to indicate that the new sequence finds ranks then names. The dark blue line after the **Military Ranks** extractor indicates placement of the **Person** extractor in the text pattern.



When you release the mouse button, a sequence object is displayed on the canvas.

1. Click the sequence object on the canvas and, under **Extractor Properties**, open the **General** tab. Rename the sequence object as Military Personnel.
2. The final extractor rule is displayed on the canvas.



#### Adding proximity rules

Proximity rules to specify the maximum number of tokens that might occur between the desired terms. In text, each word or character is generally referred to as a token. The sentence "The CEO announced the earnings last week, missing analyst estimates by 5%." contains 15 tokens: 11 words separated by spaces, 3 special characters, and 1 number.

You might want to locate references to red brick houses, knowing that phrases such as house made of red brick, red house of brick and the brick on the house reflected red might appear in the text you are analyzing. Proximity rules can ensure that these phrases are extracted while excluding sentences or phrases that coincidentally include the target terms, for example, "The red car passed a house near the brick yard".

1. Right-click an extractor on the canvas and under one of the **Add** submenus, click **Proximity Rule**.
2. In the format minimum-maximum, specify the minimum and maximum number of words, special characters, or values to allow between terms. By default, the proximity pattern is set to match the number of tokens between the two numbers you specify in your range. If the desired proximity does not vary, you can change the setting. For example, enter 0-2.
3. Select the new proximity rule. The **Extractor Properties** opens to the **Settings** tab where you can change the number range or convert to an exact number of tokens.