Fuzzy Matcher

V4.0 Documentation

Significans
AUTOMATION

Table of Contents

About Significans Automation

Significans Automation is a software integrator specializing in delivering next-generation automation to the Printing and Packaging industry.

We offer programming and expertise in custom workflow development, deployment of communication and project management systems, color management, and end-to-end business integration. While upholding software neutrality, Significans Automation advises and tailors best in class software to optimally fit the environment.

The level of sophistication that is provided increases profitability, improved quality control, and enhanced production efficiency, enabling Artificial Intelligence and Robotics, while also facilitating new revenue opportunities in e-commerce. We are driven by the conviction that customized automation is the only path forward.

Overview

Do you consistently have inconsistent data? Tried using complex Regex statements to find matches? “Fuzzy Matching”, or approximate string matching, is the technique of finding strings that match a pattern approximately (rather than exactly). The **Fuzzy Matcher** app was designed to take this principle and create an easy user interface for a popular algorithm.

If there is functionality you believe is missing or have an idea on how to expand the current functionality, please reach out to us by sending us an email with the subject prefix “*Fuzzy Matcher Functionality*”.

Example Use-Cases

- Compare filenames to find *like* filenames from a repository/directory.
- Compare PDF Spot colors/separations from a list of pre-defined colours in your personal library to determine if a color already exists.

Compatibility

Switch 25.11 and higher. Windows and Mac OSX

Incoming Connections

At least one incoming connection required.

Outgoing Connections

At least one outgoing traffic light connection is required.

Flow Element Properties

- Search by: Specify whether to utilize search terms by a file or multi-line parameter.
  - File: A path to a text file that will be read in, each line considered its own item. **(File)**
  - Encoding: Encoding type to assist in reading the file. **(File)**
  - Multiline: One-or-more items separated by newlines that would be compared to the "List" property. Each line is treated as a separated search. **(Multiline)**
- List by: Specify whether to utilize search terms by a file or multi-line parameter.
  - Search path: Specify an absolute path to a folder to start scanning nested files and directories. Note: Prioritize using UNC paths instead of mapped drives on Windows where possible. **(Directory)**
  - Recursive: If Recursion is enabled, the app will look in sub-directories of the path specified. **(Directory)**
    - Depth: Enter a numerical depth value for recursion or leave it as "All Folders" to let the system go through all possible directories available.
  - Types: Choose to include Files and/or Directories within the search path. **(Directory)**

- Search field (1-3): Defines the scope of the search **(Directory)**:

  - Name: The file or directory name itself.

  - Folder: The immediate parent directory.

  - Path: The complete system path to the item.

  Note: Duplicate selections are ignored.

- File: A path to a text file that will be read in, each line considered its own item. **(File)**
- Encoding: Encoding type to assist in reading the file. **(File)**
- Multiline: One-or-more items that would be compared to the "Search" property. Each line is checked and validated to each item in the corresponding "Search" list of items. **(Multiline)**

- Result: Specify the desired output result. Either inject the resulting JSON as a new job, or store the JSON as a dataset to the incoming job on output.
  - Filename: The outgoing file name. **(When "Inject as new job" is selected).**
  - Dataset name: The outgoing dataset name. **(When "Store as dataset" is selected).**
- Advanced options: Enable to have finer control over how the Fuzzy Match app operates. (Yes/No)
  - Case sensitive: Indicates whether comparisons should be case sensitive. (Yes/No)
  - Include score: Whether the score should be included in the result set. A score of 0 indicates a perfect match, while a score of 1 indicates a complete mismatch. (Yes/No)
  - Include matches: Whether the matches should be included in the result set. When true, each record in the result set will include the indices of the matched characters. These can consequently be used for highlighting purposes. (Yes/No)
  - Minimum character length: Only the matches whose length exceeds this value will be returned. (For instance, if you want to ignore single character matches in the result, set it to 2). (Numerical – default is "1")
  - Should sort: Whether to sort the result list, by score. (Yes/No)
  - Find all matches: When true, the matching function will continue to the end of a search pattern even if a perfect match has already been located in the string. (Yes/No)
  - Location: Determines approximately where in the text is the pattern expected to be found. (Numerical – default is "0")
  - Threshold: At what point does the match algorithm give up. A threshold of 0.0 requires a perfect match (of both letters and location), a threshold of 1.0 would match anything. (Decimal between 0 and 1 – default is "0.6").
  - Distance: Determines how close the match must be to the fuzzy location (specified by location). An exact letter match which is distance characters away from the fuzzy location would score as a complete mismatch. A distance of 0 requires the match be at the exact location specified. A distance of 1000 would require a perfect match to be within 800 characters of the location to be found using a threshold of 0.8. (Numerical, default is "100").
  - Ignore location: When true, search will ignore location and distance, so it won't matter where in the string the pattern appears. The default options only search the first 60 characters. This should suffice if it is reasonably expected that the match is within this

range. To modify this behavior, set the appropriate combination of location, threshold, distance (or Ignore location). (Yes/No)

- Extended search: This form of advanced searching allows you to fine-tune results. When true, it enables the use of unix-like search commands. White space acts as an AND operator, while a single pipe (|) character acts as an OR operator. To escape white space, use double quote ex. ="scheme language" for exact match. (Yes/No)
  - Exact: " = "
  - Include: " ' "
  - Inverse Exact: " ! "
  - Prefix Exact: " ^ "
  - Inverse Prefix: " !^ "
  - Suffix Exact: " $ "
  - Inverse Suffix: " !...$ "

    E.g. 'Man 'Old | Artist$ = Search for items that include "Man" and "Old", OR end with "Artist"
- Ignore field norm: When true, the calculation for the relevance score (used for sorting) will ignore the field-length norm. The only time it makes sense to set *Ignore field norm* to true is when it does not matter how many terms there are, but only that the query term exists. (Yes/No)
- Field norm weight: Determines how much the field-length norm affects scoring. A value of 0 is equivalent to ignoring the field-length norm. A value of 0.5 will greatly reduce the effect of field-length norm, while a value of 2.0 will greatly increase it. (Decimal – default is "01").