University of Colorado Denver

CSCI 4580/5580 DATA SCIENCE
Spring 2022

Lab 1: Supplementary Information (Regular Expressions)

Basic Expressions

The simplest regular expressions are alphanumeric strings. For example, /pattern/ will match the substring "pattern".

Wild Cards and Quantifiers

The character . matches any character. For example, the pattern /.ed/ will match "sed", "bed", and "led".

Quantifiers match the preceding expression multiple times. There are several quantifier expressions:

- * matches the preceding expression 0 or more times.
- ? matches the preceding expression 0 or 1 times.
- \{n\} matches the preceding expression n times, where n is an integer.
- \{m, n\} matches the preceding expression m to n times, where m and n are integers.

Wild cards are combined with quantifiers to match arbitrary strings, for example, /.*ed/ matches "sed", "ed", and "foo bar baz ed".

Grouping

A regular expression can be treated as a unit for quantifiers by surrounding it with \(...\). For example, /\(foo\)\{3\}/ matches 'foofoofoo'.

Character Sets

Sometimes . is too powerful a wildcard, and what you really want is to match some characters but not others. Character sets let you do that. A character set is composed of square brackets around the characters that you want to match. For example, [AEIOU] matches any uppercase vowel. You can also invert the character set with a ^, i.e., [^AEIOU] matches anything BUT an uppercase vowel. For convenience, you may also list character ranges, for example, [A-Z] or [0-9].

Beginning and End of a String

You can match the start and end of a string with ^ and $, respectively. As we will see with the sed utility, regular expressions are often evaluated line by line over a file, and in these instances ^ and $ refer to the start and end of the line.

Alternation

You can match one or more alternatives with \|. For example, /foo|bar/ matches both foo and bar. If you only want part of the regular expression to have alternatives, you can use \(...\). For example, /\(Li\|U\)nix/ matches both Linux and Unix.

Utilities other than sed

The above guide describes POSIX Basic Regular Expressions which are used by the sed utility. Many regular expression tools, such as egrep and Python's re module, use a syntax based on extended regular expressions. In this style of regular expressions, there is no backslash before |, (,), {, or }, so one must write /(Li|U)nix/ instead of /\(Li\|U\)nix/. Putting a backslash before a nonalphanumeric character always matches that character only.

Although there are standards for regular expressions, in practice, there is a lot of variations in the features and syntaxes that regular expression libraries support, so you should try to consult documentation for the tool you are using.

- Regular Expressions Operations in Python: refer to Python's official documentation https://docs.python.org/3/library/re.html.
- Regular Expressions with grep and egrep commands: refer to the official documentation of the grep and egrep commands https://www.gnu.org/software/grep/manual/grep.html.