

For exercise 10_2 an additional PDF had to be created to properly analyse the entirety of the program's functionality. First off, since it wasn't found to be clear what exactly constitutes a temporal expression, all sorts of relative expressions of time and duration were included in the output which ended up matching 256 words and phrases. That said, a number of words accompanying temporal expressions have been omitted on purpose.

“about”/“around”: There are three instances of “about” and one of “around” where temporal expressions are concerned. They were considered redundant since they simply added uncertainty when it came to the expressions they preceded.

“within”: It shows up twice in the phrase “within 5 minutes”, and it was decided that “5 minutes” can stand alone as a temporal expression with no need for “within”.

“late”/“the latest”: They were considered too vague to include despite them occurring relatively frequently.

“between”: This was a problem case. It appears five times and its behavior follows a pattern of “between X and Y”, where X and Y share a type – whether it be a date or the time – in all cases except one. Because of how the regexes have been implemented for this problem it was decided that it was more trouble than it was worth, and the temporal expressions it bridges are printed out anyway. Nevertheless, the code to match two dates with “between” has been added in comment form on line 177.

There are some special cases that need to be brought up as well.

Fixed false positives

Is there any cinema discount at 181 Broad Street in Birmingham?

==> at 1

What is name of the movie starring Adam Campbell Mary Castro Jayma Mays showing on 15th of October?

==> May

Can you tell me the date of the last performance of nights at the circus at Warwick Arts Centre?

==> nights

Is Nights at the Circus on in the West Midlands this year?

==> Nights

All four of these matches are no longer part of the output thanks to the introduction of negative lookahead. In the first case, negative lookahead didn't allow for any more than 2 digits when it comes to describing time, while in the second months were no longer matched if followed by an “s”. The last two saw the addition of negative lookahead equal to “at the circus”, with an optional “s” added right before that for good measure to account for the fact that the “s” in “nights” is also optional. These will be mentioned again in the Python PDF in regex #1, #8, #9 and #11.

Remaining false negatives

Are there any programmes for the Birmingham Hippodrome from the 21st to the 26th of May 2007?

==> the 26th of May 2007

One could say that “from the 21st to” should also be part of this temporal expression, but a similar problem that of “between” was found and the expression was left “incomplete”.

What is the name of the event that is on at the Grand Theatre on the 26th of May till the 30th of May 2007?

==> the 26th of May

till the 30th of May 2007?

==> the 30th of May 2007

Again, it would be better if this was merged into a single temporal expression but it was an extremely specific occurrence, so the loss was considered minimal.

I'd like to know whether there is a movie in Dudley for about 1 half 1 and half hours.

==> 1 and half hours

This notation was found to be too bizarre to accommodate for, therefore only the latter part of the expression was included.

Are there any cinemas near Wolverhampton that do weekday daytime discounts?

==> weekday

Clearly, “daytime” could be included as a more specific indicator for this temporal expression. However it was a unique instance that felt like it was better left alone.

Do any Birmingham cinemas do evening discounts during the week?

==> evening

“During the week” was simply not included because “during” was only combined with seasons at a very late stage of this solution.

Are there any local cinemas showing German films over the next few weeks?

“Over the next few weeks” was considered to be both too specific a case (one instance) and also too vague to include.

I would like to go to a place which reminds me of the nineteenth century.

Extremely specific fringe case that would require a lot of effort, since there would be need for a list of words that count order. If I'd gone down this path, I would also have to exclude “Twentieth Century Fox”.

What follows next is the Python PDF along with a tiny sample of the end of this program's output. The lines are so many that the results are partially overwritten on Spyder (buffer left at default 500).