Dear editor, please consider our submission, “Regular expressions and reshaping using data tables and the nc package,” for publication in the R Journal.

The main research idea contributed by this article is a novel syntax/front-end for defining wide-to-tall data reshape operations using regular expressions. These ideas are implemented in the new nc package, and are described in our paper in the subsection "New nc functions for wide-to-tall data reshaping." Our paper also describes the following new ideas:

- Support for uniform/standard output using any of three different regex engines: ICU, PCRE, RE2. The paper contains a subsection "Uniform interface to three regex engines" which explains how we added support for ICU, which was not possible to support in our previous namedCapture package.
- The uniform/standard output format from each function is a data table, which provides an efficient data query syntax. The paper contains a subsection "Data table integration and nc::field to avoid repetition" which explains how the new data table integration can simplify text processing (e.g. using by and joins).

The article aims to appeal to readers beyond nc package users, by showing detailed comparisons with R packages that provide similar functionality for data reshaping (stats, tidyr, cdata, reshape2, data.table, utils). These comparisons are in terms of functionality, syntax, output, and speed:

- Table 1 and Related Work section which explain what data reshaping features are supported by each R package/function.
- Section “Comparison with tidyr for a single output reshape column” which highlights similarities and differences in the syntax of the R code used to define data reshaping operations.
- Section “Comparing with data.table and stats::reshape for multiple reshape output columns” which highlights differences in terms of the output (which can sometimes be incorrect/unexpected using these other packages).
- Section “Comparing computation times of functions for wide-to-tall data reshaping” and Figures 2-5 which show empirical timings as a function of input data size (number of rows/columns).

I would like to highlight the broad relevance of the bug/inefficiency in base R (utils::reshape) and some other R packages (tidyr::pivot_longer, cdata::rowrecs_to_blocks) that was found as a result of this research. Our empirical timings revealed a surprising new result: the computation time of these other packages (cdata, tidyr, utils) is quadratic in the number of input reshape columns, when the desired output has multiple reshape columns (Figure 5). In contrast, the nc and data.table packages are linear, as expected. This means that for large inputs (>=10^4 reshape columns), nc/data.table are orders of magnitude faster than the other packages. These results suggest that the other packages could probably be improved by adopting the linear time algorithm used in data.table::melt, so we plan to submit issues / bug reports to the developers of the other packages. If the other package developers fix these speed issues, then we will update the timing figures. So as a result of this work, there should hopefully be speedups for the users of these other packages.

For reproducibility of figures I have included a Makefile --- typing “rm *.rds” will remove the R data files that store the timings I computed. Then typing “make” will re-do the timings, re-make the figures, and re-do texi2pdf to make the final PDF.

Here is a list of suggested reviewers:
- Jan Gorecki j.gorecki@wit.edu.pl
- Michael Chirico MichaelChirico4@gmail.com
- Kun Ren mail@renkun.me
- Karl Broman kbroman@gmail.com
- John Mount jmount@win-vector.com
- Nina Zumel nzumel@win-vector.com
- Lionel Henry lionel@rstudio.com
- G. Grothendieck ggrothendieck@gmail.com

Thanks for your careful consideration of our manuscript,
Sincerely,
Toby Dylan Hocking