2025-03-03

R Translation Glossary Package

Saranjeet Kaur Bhogal
Imperial College London

Signatories

Project team

Saranjeet Kaur Bhogal will be the lead developer for this project. She is a Research Software Engineer at Imperial College London and has experience in developing R packages and contributing to the R community. She will be responsible for developing the R package and coordinating with the RCWG to ensure that the glossary is consistent across languages. Saranjeet is a member of the RCWG and has experience in translation projects. She has also contributed to the R community through the R Dev Days.

Contributors

Consulted

The Problem

An example in-text citation (Wickham 2016).

The problem being addressed is the inconsistency in the translations glossary across languages in R. This inconsistency affects users who rely on translations to use R. The translations glossary is used to translate messages in R and is currently managed using Weblate. However, the glossary is not consistent across languages, which makes it difficult for users to translate messages accurately. This can lead to errors in translations and make it difficult for users to contribute to the translations process.

Previous attempts to resolve this problem have been made by the R Contribution Working Group (RCWG) through the use of Weblate. However, the translations glossary is still not consistent across languages. This proposal aims to address this issue by creating an R package that will allow users to access the glossary and use it to translate messages in R. This will help in creating a consistent glossary across languages and improve the translation process in R.

This problem should be tackled because it affects the user experience of R for users who do not speak English. By creating a consistent glossary across languages, we can improve the quality of translations and make it easier for users to contribute to the translations process. This will help in making R more accessible to users who do not speak English and improve the overall user experience of R.

The proposal

The R Contribution Working Group (RCWG) has been involved in a number of projects to improve the R ecosystem and involves active contribution by the community, especially through the R Dev Days. This proposal is specifically related to one such initiatives that aims to improve the translations process in R to have a consistent glossary across languages.

Overview

The RCWG uses Weblate to track and translate messages in R. However, the translations glossary is not consistent across languages. This proposal aims to create an R package that will allow users to access the glossary and use it to translate messages in R. This will help in creating a consistent glossary across languages and improve the translation process in R. It will also help in improving the quality of translations and make it easier for users to contribute to the translations process.

Benefit to the R community

The proposed package will help in creating a consistent glossary across languages and improve the translation process in R. It will help in improving the quality of translations and make it easier for users to contribute to the translations process. This will help in making R more accessible to users who do not speak English and improve the overall user experience of R.

Detail

Minimum Viable Product

The minimum viable product will be an R package that will allow users to access the glossary and use it to translate messages in R. The package will have the following high-level functions:

1. `get_glossary()`: This function will return the glossary in a data frame format.
2. `translate_message()`: This function will take a message in English and return the translated message using the glossary.
3. `update_glossary()`: This function will allow users to update the glossary.
4. `save_glossary()`: This function will allow users to save the glossary to a file.
5. `load_glossary()`: This function will allow users to load the glossary from a file.
6. `reset_glossary()`: This function will allow users to reset the glossary to the original version.
7. `get_languages()`: This function will return the list of languages available in the glossary.

Architecture

The package will use the Weblate API to access the glossary and translate messages. The package will have a data frame that will store the glossary and will be used to translate messages. The package will also have a function that will allow users to update the glossary.

Assumptions

The assumptions for this project are:

1. The Weblate API will be available and accessible.
2. The glossary will be consistent across languages.
3. The package will be able to handle different languages and translations.
4. The package will be able to handle different types of messages and translations.

Project plan

Start-up phase

The project will be setup as a GitHub repository and the various steps for the development will be opened as issues on GitHub. Development will take place and be tracked by creating pull requests associated to the issues. Appropriate license will be chosen for the package. The contributors to the project will be acknowledged using the all contributors bot on GitHub.

Technical delivery

The work on the package development will begin around July 2025 until October 2025. The development work will be associated with milestones and deliverables which will be tracked through GitHub. A final blog post will be published to share the work done along with social media announcements. If possible, the work will also be publised on relevant events and R community events (including R-Ladies and RUG meetups).

Other aspects

The project will be promoted through the RCWG to ensure that the R community is aware of the project. Regular updates will be provided to the R community on the progress of the project. Feedback will be sought from the community to ensure that the package meets the needs of the users. The updates will be provided through regular blog posts and announcements on social media platforms.

Requirements

The project requires the following to make it happen:

1. **People**: The project requires a lead developer who has experience in developing R packages and has contributed to the R community. The lead developer will be responsible for developing the R package and coordinating with the RCWG to ensure that the glossary is consistent across languages.

2. **Processes**: The project requires processes to be put in place to ensure that the package is developed, tested, documented, and released in a timely manner. The project will follow the

best practices for R package development and will seek feedback from the R community to ensure that the package meets the needs of the users.

3. **Tools & Tech**: The project requires tools and technologies to be put in place to support the development of the R package. The project will use R, devtools, testthat, roxygen2, pkgdown, GitHub, and the Weblate API to develop the package. The package will eventually be released on CRAN so that it is easily accessible to users.

4. **Funding**: The project requires funding to support the development of the R package. The funding will be used to cover the costs associated with the development, testing, documentation, and release of the package by the lead developer.

The project will require coordination between the lead developer, the RCWG, and the R community to ensure that the package meets the needs of the users and is easy to use.

People

Saranjeet Kaur Bhogal, who is a Research Software Engineer at Imperial College London, will be the lead developer for this project. She has experience in developing R packages and has contributed to the R community. She will be responsible for developing the R package and coordinating with the RCWG to ensure that the glossary is consistent across languages.

The RCWG will be involved in this project to provide guidance and support. The RCWG is a group of volunteers who are actively involved in improving the R ecosystem and have experience in translation projects.

Feedback on the work will be sought from the R community at large, especially from users who are involved in the translations process. This will help in ensuring that the package meets the needs of the community and is easy to use.

Processes

The project will follow the following processes:

1. **Development**: The R package will be developed using the `devtools` package. The package will be hosted on GitHub and will follow the best practices for R package development.

2. **Testing**: The package will be tested using the `testthat` package. Unit tests will be written to ensure that the package functions as expected.

3. **Documentation**: The package will be documented using `roxygen2` and `pkgdown`. The documentation will be available on the package website.

4. **Translation**: Feedback will be sought from the R community to ensure that the translations are accurate and consistent across languages.

5. **Release**: The plan is to eventually release the package on CRAN so that it is easily accessible to users.

6. **Community engagement**: Regular updates will be provided to the R community on the progress of the project. Feedback will be sought from the community to ensure that the package meets the needs of the users.

7. **Handover**: The package will be handed over to the R community at large so that they can contribute to its development and maintenance.
8. **Code of conduct**: A code of conduct will be put in place to ensure that the project is a safe and welcoming space for all contributors.
9. **Governance**: The project will be governed by the RCWG to ensure that it aligns with the goals of the R community.
10. **Sustainability**: The project will be designed to be sustainable in the long term.
11. **Feedback**: Feedback will be sought from the R community to ensure that the package meets the needs of the users.

Tools & Tech

The project will use the following tools and technologies:

1. **R**: The project will be developed using the R programming language.
2. **devtools**: The package will be developed using the `devtools` package.
3. **testthat**: The package will be tested using the `testthat` package.
4. **roxygen2**: The package will be documented using the `roxygen2` package.
5. **pkgdown**: The documentation will be available on the package website.
6. **GitHub**: The package will be hosted on GitHub.
7. **Weblate API**: The package will use the Weblate API to access the glossary and translate messages.
8. **CRAN**: The plan is to eventually release the package on CRAN.

Most of the tools stated above are open-source and widely used in the R community. The Weblate API is used by the RCWG to track and translate messages in R.

Funding

The project seeks funding from the R Consortium Infrastructure Steering Committee (ISC) to support the development of the R package. The funding will be used to cover the costs associated with the development, testing, documentation, and release of the package by the lead developer.

The total funding expected for the project is $5,000. The breakdown of the costs is as follows:

Summary

The project requires funding to support the development of the R package that will allow users to access the glossary and use it to translate messages in R. The package will help in creating a consistent glossary across languages and improve the translation process in R. The funding will be used to cover the costs associated with the development, testing, documentation, and release of the package by the lead developer.

Success

The project will be considered a success if the following criteria are met:

1. The R package is developed and tested.
2. The package is documented and available on the package website.
3. Eventually release the package on CRAN. (This may not happen within the proposed timeline but is a long-term goal. However, the package will be available on GitHub for users to access.)
4. Feedback will be sought from the R community to ensure that the package meets the needs of the users.

Definition of done

The project will be considered done when the following criteria are met:

1. The R package is developed and tested.
2. The package is documented and available on the package website.
3. Feedback will be sought from the R community to ensure that the package meets the needs of the users.
4. Long-term goal: The package is released on CRAN.

Measuring success

The success of the project will be measured by the following metrics:

1. Closing the issues related to the package development on GitHub.
2. Pull requests are reviewed and merged.
3. The package is documented and available on the package website.

Future work

The project can be extended in the following ways:

1. Add more functions to the package to make it more user-friendly.
2. Improve the translation process by adding more languages to the glossary.
3. Add more features to the package to make it more useful for users.
4. Improve the documentation and make it more accessible to users.
5. Release the package on CRAN to make it more accessible to users.

Key risks

The are not any major risks associated with this project. The idea for the project is based on an actual need by the R community and has the support of the RCWG. The lead developer has experience in developing R packages and has contributed to the R community. The project will follow the best practices for R package development and will seek feedback from the R community to ensure that the package meets the needs of the users.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org.