SHORT TERM SCIENTIFIC MISSION (STSM)
SCIENTIFIC REPORT

This report is submitted for approval by the STSM applicant to the STSM coordinator

Action number: CA16105
STSM title: Making sense of crowdsourced language data: Tools for efficient processing and visualization
STSM start and end date: 15/02/2020 to 22/02/2020
Grantee name: Valeria Caruso

PURPOSE OF THE STSM:

The STSM offered the opportunity to exchange experiences and methods for accessing crowdsourced data effectively and efficiently by automatic importing, cleaning and analysis. User-inputted information is paramount in the design and development process of mobile applications, where new tools for language learning and linguistic support are needed and should be the object of ad-hoc scientific investigation. Crowdsourcing can be beneficial to this aim by enriching mobile applications with users' contents and specific desiderata from the communities. Additionally, users' feedbacks could be of aid in validating the usefulness of any app content.
Efficient ways to process crowdsourced data are thus key for app design.

DESCRIPTION OF WORK CARRIED OUT DURING THE STSMS

The main activities carried out during the STSM were addressed to trial state-of-the-art packages and methods available for the R computing environment (R Core Team 2020), which is the leading tool in data analysis.
Different datasets were prepared for this purpose by the Host and another STSM Grantee, such as:

1. Different toy datasets from R packages
2. Sets of System Usability Survey feedback from different user studies
3. User log files for the English Wikipedia, harvested from Wikimedia servers along with related lexical data (e.g. lexical prevalence data collected via crowdsourcing)
4. Data from the European Survey of Dictionary Use and Culture, collected as part of a previous COST action (Kosem et al. 2019)
5. Academic journals citation data taken from different sources
6. Language data from corpora

Using these datasets different actions were undertaken:

1. Testing the use of GIT desktop and GitHub in conjunction with RStudio to streamline collaboration.
2. Trialing out different Exploratory Data Analysis packages to asses their efficiency with crowdsourced datasets (autoEDA, summary tools).
3. Labeling a large number of words/lemmas in plot visualizations while assuring readability.
4. Testing data import from badly formatted Excel files (readr, xlsx, XL Connect, fread).
5. Test-driving several data cleaning packages and functions (e.g. daff, janitor).
6. Testing data imputation with the simputation package.
7. Visualizing Likert-type data (from surveys) efficiently.

8. Harnessing the RShiny package for interactive web-based presentations of crowdsourced data.
9. Asses effective tools for data storage and visualization in web-based and mobile applications.

DESCRIPTION OF THE MAIN RESULTS OBTAINED

Main results of the STSM can be summarized as follows:

1. Using GIT and GitHub in the context of RStudio offers no remarkable advantage over cloud-based collaboration (such as OneDrive) in terms of file synchronization.
2. To the aim of automatically processing crowdsourced data, autoEDA (Exploratory Data Analysis) package proved to be not particularly suited because it is addressed to numerical data having many correlations between the variables involved. Therefore, it seems more reasonable to use a more informed and targeted approach that takes into account the nature and structure of the data to process them more efficiently.
3. To address the problem of labeling a large number of words/lemmas in visualizations and maintain good readability, the repel package (base R plotting) or the ggrepel proved to be effective.
4. For importing Excel files that are badly formatted the manual check remains the best option and importers are confused mainly by blank lines in the data. It is worth noting that the efficient fread function from the data.table approach uses, by default, the first 1000 rows to guess the data structure; if changes or errors occur further down the file, they will often lead to problems down the line.
5. As far as data cleaning packages and functions are concerned, the existing packages proved to be of little help for more experienced R coders, whilst they may be useful to the less versed in data analysis.
6. Testing data imputation with the simputation package proved to be helpful since it is easy to use, though some errors were discovered (e.g. in the Vignette where the complete dataset was fed to imputing) and it is still unclear whether data should be imputed in the first place.
7. For visualizing data (from surveys) efficiently the Likert package offers attractive solutions (e.g. horizontal bars and density plots) but needs to be implemented to deal with some deficiencies like forcing alphabetical ordering and having no provisions for items with reverse polarity.
8. Interactive web-based presentations of crowdsourced data can be made by using the RShiny package, though this requires a lot of additional proficiency in R and is not advisable for less proficient coders.
9. Additional activities were carried out to asses data storage and visualization for web and mobile applications using XML and Java files which might be beneficial for collecting language data and provide lexicographical descriptions also in gamification applications. To this aim, Oxygen XML Editor offer easy to use facilities and guarantees high compatibility with different database types (e.g. IBM DB2, Oracle Berkeley DB, Microsoft SQLServer).

FUTURE COLLABORATIONS (if applicable)

Collaboration with members of the Action and future inquiries in all of the above topics is welcome.