INCA: Infrastructure for content analysis

Trilling, D.; van de Velde, B.; Kroon, A.C.; Löcherbach, F.; Araujo, T.; Strycharz, J.; Raats, T.; de Klerk, L.; Jonkman, J.G.F.

DOI
10.1109/eScience.2018.00078

Publication date
2018

Document Version
Final published version

Published in
IEEE 14th International Conference on eScience

2018 IEEE 14th International Conference on e-Science

INCA: Infrastructure for Content Analysis

Damian Trilling*, Bob van de Velde†, Anne C. Kroon*, Felicia Löcherbach‡, Theo Araujo*, Joanna Strycharz*, Tamara Raats‡, Lisa de Klerk‡ and Jeroen G.F. Jonkman*
*Dep. of Communication Science, Amsterdam School of Communication Research, University of Amsterdam, Netherlands
Email: {d.c.trilling, a.c.kroon, t.b.araujo, j.strycharz, j.g.f.jonkman}@uva.nl
†Informatics Institute, University of Amsterdam, Netherlands
Email: r.n.vandevelde@uva.nl
‡Dep. of Communication Science, Graduate School of Communication, University of Amsterdam, Netherlands
Email: {felicia.locherbach, tamara.raats}@student.uva.nl; lisadk93@gmail.com

***Abstract*—We present INCA (short for INfrastructure for Content Analysis), a Python module for collecting, storing, processing, and analyzing a wide variety of media content, including but not limited to news, political debates, social media, forums, and customer reviews. Using Elasticsearch as a database backend and Celery for task management, it makes automated content analysis scalable. INCA's main objective is to enable and promote an integrated workflow. INCA focuses on re-usability of data, processors, and analyses; making all steps of automated content analysis (ACA) accessible to social scientists, without requiring advanced programming skills. Here, we present the aim, implementation, and recommended workflow for INCA.**

***Index Terms*—automated content analysis, Python module, social science, communication science**

I. Introduction

INCA aims to solve a major challenge in social science, and in particular communication science: the analysis of increasingly large, increasingly dynamic, media content corpora, which contain not only text, but also various other features (e.g., metadata, social signals, ...). Gradually, the manual analysis of (physical) newspaper copies stored in archives is being replaced by automated approaches that differ in terms of (a) collection, (b) storage, and (c) analysis from the traditional approaches. Digital outlets often require continuous data collection using web scraping and API clients, database technologies, natural language processing as well as machine learning. In earlier work [1], we have outlined how an integrated framework for these tasks could look like; now, we present its implementation.

II. Related work

Several successful attempts have been made to develop a framework for the collection, annotation, and analysis of media content (e.g., [2], [3]). While the underlying techniques are very similar (see also [4]), INCA sets different priorities. Instead of developing an own GUI or WUI, or aiming at users who want to perform analyses in R using an API to interact with the system (e.g., as suggested by [2], [5]), we want to provide a framework for usage within a Python environment. INCA can be run either locally or on a remote server, and it is particular designed to make extension by the user easy, who can easily add new scrapers, processors, or analysis functions as they see fit. In particular, we aimed at integrating the capability of conducting diverse forms of automated content analysis (ACA) as described in [6] within our framework.

This work was carried out on the Dutch national e-infrastructure with the support of SURF Cooperation. We thank all student assistants who contributed additional code: Eoin Hennessey, Marieke van Hoof, Konstantinos Lampridis, Payal Mitra, Arno Polegato, Jara Ruth Strebel, and Chamoetal Zeidler.

III. INCA

INCA consists of the following components:

- scrapers and API clients;
- database management, importers, exporters;
- text processors (NLP, NER, POS-tagging);
- analysis (e.g., classification, topic modeling, time series).

INCA provides flexible (NoSQL based) data storage with rich text-search support through an underlying integration with an (optional) Elasticsearch database. This allows INCA to handle the wildly differing content across news articles, press releases, customer reviews, tweets, Youtube comments etc. without additional configuration. For horizontal scale-ability, all tasks can (also optionally) be run within the Celery framework implementation of a distributed task queue.

INCA provides a consistent way of accessing all functionality. As the following listing illustrates, just a couple of lines are necessary to scrape a news site, apply some preprocessing, and conduct an analysis.

```
from inca import Inca
myinca = Inca()
myinca.rssscrapers.nu()
myinca.processors.lower('nu','text')
myinca.analysis.VAR(<parameters>)
```

While INCA can be used locally, we suggest running it on a server, e.g. on a cloud computing platform. By using JupyterHub, users can access INCA with user-friendly Jupyter Notebooks. In addition, running Kibana on the same system can provide dashboards for visual exploration of the data and for monitoring ongoing data collections.

IV. Limitations and Future Work

Not all functionality described in [6] is already available within INCA. Future work also includes the analysis of non-textual content (e.g., images) and improving interoperability with related frameworks, in particular AMCAT [2].

978-1-5386-9156-4/18/$31.00 ©2018 IEEE
DOI 10.1109/eScience.2018.00078

REFERENCES

[1] D. Trilling and J. G. F. Jonkman, "Scaling up Content Analysis," *Communication Methods and Measures*, vol. online first, 2018.

[2] W. van Atteveldt, *Semantic Network Analysis: Techniques for Extracting, Representing, and Querying Media Content*. Charleston, SC: BookSurge, 2008.

[3] A. Niekler, G. Wiedemann, and G. Heyer, "Leipzig Corpus Miner – a text mining infrastructure for qualitative data analysis," in *Terminology and Knowledge Engineering 2014*, Berlin.

[4] O. de Rooij, A. Vishneuski, and M. de Rijke, "xTAS: Text analysis in a timely manner," *Proceedings of the 12th Dutch-Belgian Information Retrieval Workshop (DIR 2012)*, pp. 89–90, 2012.

[5] K. Welbers, W. Van Atteveldt, and K. Benoit, "Text Analysis in R," *Communication Methods and Measures*, vol. 11, no. 4, pp. 245–265, 2017.

[6] J. W. Boumans and D. Trilling, "Taking stock of the toolkit: An overview of relevant automated content analysis approaches and techniques for digital journalism scholars," *Digital Journalism*, vol. 4, no. 1, pp. 8–23, 2016.