The Journal of Open Source Software

oolong: An R package for validating automated content analysis tools

Chung-hong Chan[1] **and Marius Sältzer**[1]

1 Mannheimer Zentrum für Europäische Sozialforschung, Universität Mannheim

DOI: 10.21105/joss.02461

Software

- Review
- Repository
- Archive

Editor: Kakia Chatsiou

Reviewers:

- @pdwaggoner
- @mbod
- @Kudusch

Submitted: 04 July 2020
Published: 09 November 2020

Statement of need

Oolong is an R package providing functions for semantic validation of topic modeling and dictionary-based methods, two main tools for doing automated content analysis (Boumans & Trilling, 2016; Günther & Quandt, 2016).

While the validation of statistical properties of topics models is well established, the substantive meaning of categories uncovered is often less clear and their interpretation reliant on "intuition" or "eyeballing". As Chang et al. (2009, p. 1) put it: "qualitative evaluation of the latent space" or figuratively, reading tea leaves.

The story for dictionary-based methods is not better. Researchers usually assume these dictionaries have built-in validity and use them directly in their research. However, multiple validation studies (Boukes et al., 2020; González-Bailón & Paltoglou, 2015; Ribeiro et al., 2016) demonstrate these dictionaries have very limited criterion validity.

Oolong provides a set of tools to objectively judge substantive interpretability to applied users in disciplines such as political science and communication science. It allows standardized content based testing of topic models as well as dictionary-based methods with clear numeric indicators of semantic validity. Oolong makes it easy to generate standard validation tests suggested by Chang et al. (2009) and Song et al. (2020).

Validation of automated content analysis

Validity is a requirement of content analysis (Krippendorff, 2018; Neuendorf, 2016). Validation of automated methods has been called for by many scholars, e.g. Grimmer & Stewart (2013); Ribeiro et al. (2016); Van Atteveldt & Peng (2018). But how to validate these methods? The paper by DiMaggio et al. (2013) conceptualizes validation of automated methods as three different operations and the three operations supplement each other. These three operations are: 1) *statistical* validation –to see if the model results agree with the assumptions of the model. Examples of statistical validation are calculation of pointwise mutual information, perplexity or semantic coherence of a topic model; 2) *semantic* validation –to see if the model results are semantically making sense. This procedure involves comparing model results with human judgment (Grimmer & King, 2011); 3) *predictive* validation –to see if the model results can predict external events (Quinn et al., 2010). For example, one can study whether external events can explain surges in attention to a topic extracted by a topic model.

This package focuses on semantic validation for three reasons: First, there is existing architecture for conducting statistical validation and predictive validation. Topic modeling packages such as `text2vec` (Selivanov et al., 2020), `topicmodels` (Grün & Hornik, 2011), and `textmineR` (Jones, 2019) provide functions to calculate metrics such as perplexity and semantic coherence. Packages such as `stminsights` (Schwemmer, 2018) and `LDAvis` (Sievert

& Shirley, 2015) offer additional qualitative methods for predictive validation. As of writing, `tosca` (Koppers et al., 2020) is the only package dealing with semantic validation. But the text-based interface might pose challenges to human annotators and it can only support topic models from the `lda` package (Chang, 2015).

Second, results from statistical validation do not always agree with those from semantic validation. For example, a topic model with a lower perplexity does not have a better interpretability (Chang et al., 2009). Of course, there are also metrics from statistical validation that are shown to be correlated with semantic validity, e.g. semantic coherence (Mimno et al., 2011). But this correlation is also dependent on the text material. For example, Fan et al. (2019) show that semantic coherence is weakly correlated at best with human assessment, when the text material used for training a topic model has some frequent terms. But still, calculation of semantic coherence is recommended in the best practice paper by Maier et al. (2018). Nonetheless, conducting only statistical validation is not adequate because these three validation operations supplement each other.

Finally, predictive validation is dependent on research questions and thus it is difficult to be generalized as a reusable software framework. Additionally, the relationship between external (sociopolitical) events and the results from automated content analysis tools is usually what social scientists are eager to study, cf. using topic models for information retrieval (Yi & Allan, 2008). We do not believe social scientists would ignore conducting any form of predictive validation.

Oolong focuses on semantic validation. The package provides the "human-in-the-loop" semantic validation procedures suggested by Chang et al. (2009) and Song et al. (2020). The procedure proposed by Chang et al. (2009) has been adopted in subsequent social science studies as the gold standard to validate topic models, e.g. Bohr (2020), Chuang et al. (2015), and Miller (2017). The procedure proposed by Song et al. (2020) emphasizes both criterion validity and interrater reliability.

Semantic validation of topic models

Topic models can be validated by word intrusion test and topic intrusion test (Chang et al., 2009). In these tests, a human rater is asked to pick an odd word from a bunch of words (word intrusion test) or pick an odd topic from a bunch of topics for a document (topic intrusion test). Oolong provides an easy-to-use Shiny interface for these tests (Figure 1).

Currently, oolong supports a variety of topic models, e.g. structural topic models / correlated topic models from `stm` (Roberts et al., 2019), warp-LDA models from `text2vec` (Selivanov et al., 2020), latent dirichlet allocation / correlated-topic models from `topicmodels` (Grün & Hornik, 2011), biterm topic models from `BTM` (Wijffels, 2020) and keyword-assisted topic models from `keyATM` (Eshima et al., 2020).

For instance, `abstracts_stm` is a structural topic model trained with the text data from `abstracts$text` (Chan & Grill, 2020).

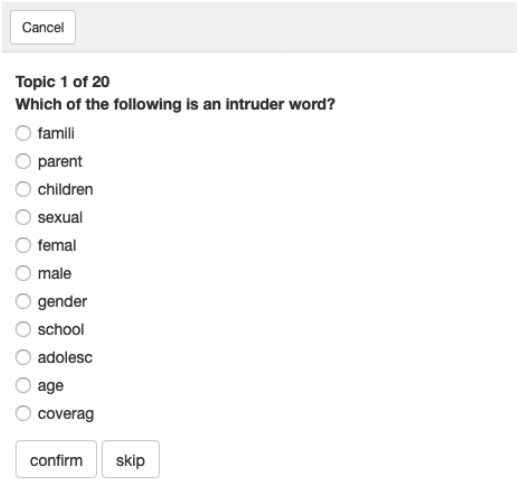

Figure 1: A screenshot of word intrusion test

```
library(stm)
library(tibble)
library(dplyr)
library(quanteda)
library(oolong)

abstracts_stm

## A topic model with 20 topics, 2500 documents and a 3998 word dictionary.
```

The function `create_oolong` creates a test object with both word intrusion test and topic intrusion test.

```
oolong_test <- create_oolong(input_model = abstracts_stm,
                             input_corpus = abstracts$text)
oolong_test

## An oolong test object with k = 20, 0 coded.
## Use the method $do_word_intrusion_test() to do word intrusion test.
## With 25 cases of topic intrusion test. 0 coded.
## Use the method $do_topic_intrusion_test() to do topic intrusion test.
## Use the method $lock() to finalize this object and see the results.
```

The tests can be administered with methods `do_word_intrusion_test` and `do_topic_intrusion_test`.

```
oolong_test$do_word_intrusion_test()
oolong_test$do_topic_intrusion_test()
```

After both tests has been done by a human rater, the test object must be locked and then accuracy metrics such as model precision (MP) and TLO (topic log odd) are displayed.

```
oolong_test$lock()
oolong_test
```

```
## An oolong test object with k = 20, 20 coded.
## 95%  precision
## With 25 cases of topic intrusion test. 25 coded.
## TLO: -0.135
```

The suggested workflow is to have at least two human raters to do the same set of tests. Test object can be cloned to allow multiple raters to do the test. More than one test object can be studied together using the function `summarize_oolong()`.

```
oolong_test_rater1 <- create_oolong(abstracts_stm, abstracts$text)
oolong_test_rater2 <- clone_oolong(oolong_test_rater1)

## Let rater 1 do the test.
oolong_test_rater1$do_word_intrusion_test()
oolong_test_rater1$do_topic_intrusion_test()
oolong_test_rater1$lock()

## Let rater 2 do the test.
oolong_test_rater2$do_word_intrusion_test()
oolong_test_rater2$do_topic_intrusion_test()
oolong_test_rater2$lock()
```

Get a summary of the two objects.

```
summarize_oolong(oolong_test_rater1, oolong_test_rater2)
```

```
## Mean model precision: 0.3
## Quantiles of model precision: 0.25, 0.275, 0.3, 0.325, 0.35
## P-value of the model precision
##  (H0: Model precision is not better than random guess): 0.0494
## Krippendorff's alpha: 0.071
## K Precision:
## 0, 0, 0, 0, 0, 0.5, 1, 0, 0.5, 0, 0.5, 0, 0, 0.5, 0.5, 0, 0.5, 0.5, 0.5, 1
## Mean TLO: -1.9
## Median TLO: -1.54
## Quantiles of TLO: -6.05, -3.56, -1.54, 0, 0
## P-Value of the median TLO
## (H0: Median TLO is not better than random guess): 0.014
```

Two key indicators of semantic validity are mean model precision and median TLO. Please interpret the magnitude of the two values (see Chang et al., 2009) rather than the two statisical tests. The two statistical tests are testing whether the raters did better than random guess. Therefore, rejection of the null hypothesis is just the bare minimum of topic interpretability, *not* an indicator of adquate semantic validity of the topic model. Besides, please a very conservative significant level, e.g. alpha < 0.001.

Semantic validation of dictionary-based methods

Dictionary-based methods such as AFINN (Nielsen, 2011) can be validated by creating a gold standard dataset (Song et al., 2020). Oolong provides a workflow for generating such gold standard dataset.

For example, you are interested in studying the sentiment of tweets from Donald Trump. `trump2k` is a random subset of 2,000 tweets from Donald Trump. And you would like to use AFINN to extract sentiment from these tweets. In this analysis, AFINN sentiment score is the *target value*.

A test object can be generated also with `create_oolong`. The argument `construct` should be an adjective, e.g. "positive" or "liberal".

```
trump <- create_oolong(input_corpus = trump2k,
                       construct = "positive",
                       exact_n = 20)
trump
```

```
## An oolong test object (gold standard generation) with 20 cases, 0 coded.
## Use the method $do_gold_standard_test() to generate gold standard.
## Use the method $lock() to finalize this object and see the results.
```

Similarly, we suggest to have at least two human coders to do the same set of tests.

```
trump2 <- clone_oolong(trump)
```

Instruct two coders to code the tweets and lock the objects.

```
trump$do_gold_standard_test()
trump2$do_gold_standard_test()
trump$lock()
trump2$lock()
```

The method `turn_gold` converts a test object into a quanteda corpus (Benoit et al., 2018).

```
gold_standard <- trump$turn_gold()
gold_standard
```

```
## Corpus consisting of 20 documents and 1 docvar.
## text1 :
## "Thank you Eau Claire, Wisconsin.  #VoteTrump on Tuesday, Apr..."
##
## text2 :
## ""@bobby990r_1: @realDonaldTrump would lead polls the second ..."
##
## text3 :
## ""@KdanielsK: @misstcassidy @AllAboutTheTea_ @realDonaldTrump..."
##
## text4 :
## "Thank you for a great afternoon Birmingham, Alabama! #Trump2..."
##
## text5 :
## ""@THETAINTEDT: @foxandfriends @realDonaldTrump Trump 2016 ht..."
##
## text6 :
## "People believe CNN these days almost as little as they belie..."
##
## [ reached max_ndoc ... 14 more documents ]
## Access the answer from the coding with quanteda::docvars(obj, 'answer')
```

This corpus can be used to calculate the target value, e.g. AFINN.

```
dfm(gold_standard, remove_punct = TRUE) %>% dfm_lookup(afinn) %>%
    quanteda::convert(to = "data.frame") %>%
    mutate(matching_word_valence = (neg5 * -5) + (neg4 * -4) +
               (neg3 * -3) + (neg2 * -2) + (neg1 * -1) +
               (zero * 0) + (pos1 * 1) + (pos2 * 2) + (pos3 * 3) +
               (pos4 * 4) + (pos5 * 5),
           base = ntoken(gold_standard, remove_punct = TRUE),
           afinn_score = matching_word_valence / base) %>%
           pull(afinn_score) -> afinn_score
```

Summarize all oolong objects with the target value.

```
res <- summarize_oolong(trump, trump2, target_value = afinn_score)
```

Printing the summary shows Krippendorff's Alpha, an indicator of interrater reliability. The validity metrics of a text analytic method can be tinted by poor interrater reliability of manual annotations (Song et al., 2020). It is important to ensure high interrater reliability first.

```
res
```

```
## Krippendorff's Alpha: 0.931
## Correlation: 0.744 (p = 2e-04)
## Effect of content length: -0.323 (p = 0.1643)
```

Additional diagnostic plots can also be displayed (Figure 2).

```
plot(res)
```

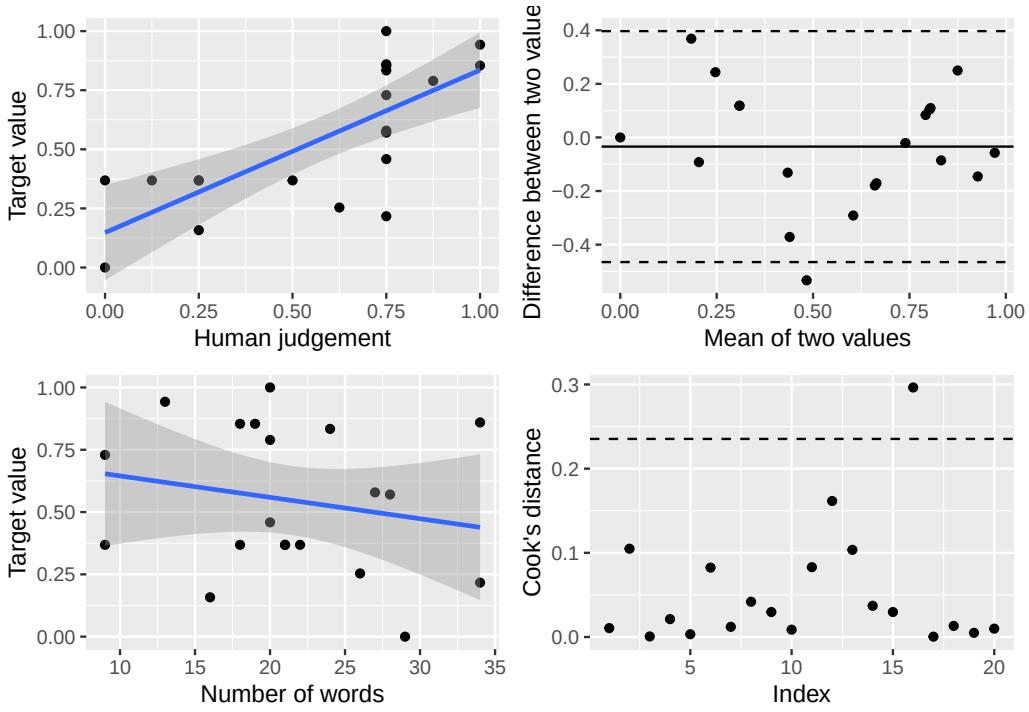

Figure 2: Diagnostic plots generated by oolong

The 4 subplots from left to right, top to bottom are:

1. Correlation between human judgement and target value - A strong correlation between the two is an indicator of criterion validity of the target value.
2. Bland-Altman plot - If the dots are randomly scattering around the mean value (solid line), it is an indicator of good agreement between human judgement and the target value.
3. Correlation between target value and content length - If there is no strong correlation between the target value and content length, it is an indicator of robustness against the influence of content length (see Chan et al., 2020).
4. Cook's distance of all data points - if there are only a few dots above the threshold (dotted line), it is an indicator of robustness against the influence of outliers.

Acknowledgements

The development of oolong is partially supported by SAGE Concept Grant.

References

Benoit, K., Watanabe, K., Wang, H., Nulty, P., Obeng, A., Müller, S., & Matsuo, A. (2018). Quanteda: An R package for the quantitative analysis of textual data. *Journal of Open Source Software*, *3*(30), 774. https://doi.org/10.21105/joss.00774

Bohr, J. (2020). Reporting on climate change: A computational analysis of us newspapers and sources of bias, 1997–2017. *Global Environmental Change*, *61*, 102038. https://doi.org/10.1016/j.gloenvcha.2020.102038

Boukes, M., Velde, B. van de, Araujo, T., & Vliegenthart, R. (2020). What's the tone? Easy doesn't do it: Analyzing performance and agreement between off-the-shelf sentiment analysis tools. *Communication Methods and Measures*, *14*(2), 83–104. https://doi.org/10.1080/19312458.2019.1671966

Boumans, J. W., & Trilling, D. (2016). Taking stock of the toolkit: An overview of relevant automated content analysis approaches and techniques for digital journalism scholars. *Digital Journalism*, *4*(1), 8–23. https://doi.org/10.1080/21670811.2015.1096598

Chan, C.-h., Bajjalieh, J., Auvil, L., Wessler, H., Althaus, S., Welbers, K., Atteveldt, W. van, & Jungblut, M. (2020). Four best practices for measuring news sentiment using "off-the-shelf" dictionaries: A large-scale p-hacking experiment. *Computational Communication Research*. https://doi.org/10.31235/osf.io/np5wa

Chan, C.-h., & Grill, C. (2020). The highs in communication research: Research topics with high supply, high popularity and high prestige in high-impact journals. *Communication Research*. https://doi.org/10.1177/0093650220944790

Chang, J. (2015). *Lda: Collapsed gibbs sampling methods for topic models*. https://CRAN.R-project.org/package=lda

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. *Advances in Neural Information Processing Systems*, 288–296. https://papers.nips.cc/paper/3700-reading-tea-leaves-how-humans-interpret-topic-model

Chuang, J., Roberts, M. E., Stewart, B. M., Weiss, R., Tingley, D., Grimmer, J., & Heer, J. (2015). TopicCheck: Interactive alignment for assessing topic model stability. *Proceedings of the 2015 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, 175–184. https://doi.org/10.3115/v1/N15-1018

DiMaggio, P., Nag, M., & Blei, D. (2013). Exploiting affinities between topic modeling and the sociological perspective on culture: Application to newspaper coverage of us government arts funding. *Poetics*, *41*(6), 570–606. https://doi.org/10.1016/j.poetic.2013.08.004

Eshima, S., Imai, K., & Sasaki, T. (2020). Keyword assisted topic models. *arXiv Preprint arXiv:2004.05964*.

Fan, A., Doshi-Velez, F., & Miratrix, L. (2019). Assessing topic model relevance: Evaluation and informative priors. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *12*(3), 210–222. https://doi.org/10.1002/sam.11415

González-Bailón, S., & Paltoglou, G. (2015). Signals of public opinion in online communication: A comparison of methods and data sources. *The ANNALS of the American Academy of Political and Social Science*, *659*(1), 95–107. https://doi.org/10.1177/0002716215569192

Grimmer, J., & King, G. (2011). General purpose computer-assisted clustering and conceptualization. *Proceedings of the National Academy of Sciences*, *108*(7), 2643–2650. https://doi.org/10.1073/pnas.1018067108

Grimmer, J., & Stewart, B. M. (2013). Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, *21*(3), 267–297. https://doi.org/10.1093/pan/mps028

Grün, B., & Hornik, K. (2011). topicmodels: An R package for fitting topic models. *Journal of Statistical Software*, *40*(13), 1–30. https://doi.org/10.18637/jss.v040.i13

Günther, E., & Quandt, T. (2016). Word counts and topic models: Automated text analysis methods for digital journalism research. *Digital Journalism*, *4*(1), 75–88. https://doi.org/10.1080/21670811.2015.1093270

Jones, T. (2019). *TextmineR: Functions for text mining and topic modeling*. https://CRAN.R-project.org/package=textmineR

Koppers, L., Rieger, J., Boczek, K., & von Nordheim, G. (2020). *Tosca: Tools for statistical content analysis*. https://doi.org/10.5281/zenodo.3591068

Krippendorff, K. (2018). *Content analysis: An introduction to its methodology*. SAGE.

Maier, D., Waldherr, A., Miltner, P., Wiedemann, G., Niekler, A., Keinert, A., Pfetsch, B., Heyer, G., Reber, U., Häussler, T., & others. (2018). Applying lda topic modeling in communication research: Toward a valid and reliable methodology. *Communication Methods and Measures*, *12*(2-3), 93–118. https://doi.org/10.1080/19312458.2018.1430754

Miller, C. (2017). Australia's anti-islam right in their own words. Text as data analysis of social media content. *Australian Journal of Political Science*, *52*(3), 383–401. https://doi.org/10.1080/10361146.2017.1324561

Mimno, D., Wallach, H., Talley, E., Leenders, M., & McCallum, A. (2011). Optimizing semantic coherence in topic models. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 262–272.

Neuendorf, K. A. (2016). *The content analysis guidebook*. SAGE.

Nielsen, F. Å. (2011). A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv Preprint arXiv:1103.2903*.

Quinn, K. M., Monroe, B. L., Colaresi, M., Crespin, M. H., & Radev, D. R. (2010). How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science*, *54*(1), 209–228. https://doi.org/10.1111/j.1540-5907.2009.00427.x

Ribeiro, F. N., Araújo, M., Gonçalves, P., Gonçalves, M. A., & Benevenuto, F. (2016). Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, *5*(1), 1–29. https://doi.org/10.1140/epjds/s13688-016-0085-1

Roberts, M. E., Stewart, B. M., & Tingley, D. (2019). stm: An R package for structural topic models. *Journal of Statistical Software*, *91*(2), 1–40. https://doi.org/10.18637/jss.v091.i02

Schwemmer, C. (2018). *Stminsights: A 'shiny' application for inspecting structural topic models*. https://CRAN.R-project.org/package=stminsights

Selivanov, D., Bickel, M., & Wang, Q. (2020). *Text2vec: Modern text mining framework for R*. https://CRAN.R-project.org/package=text2vec

Sievert, C., & Shirley, K. (2015). *LDAvis: Interactive visualization of topic models*. https://CRAN.R-project.org/package=LDAvis

Song, H., Tolochko, P., Eberl, J.-M., Eisele, O., Greussing, E., Heidenreich, T., Lind, F., Galyga, S., & Boomgaarden, H. G. (2020). In validations we trust? The impact of imperfect human annotations as a gold standard on the quality of validation of automated content analysis. *Political Communication*, 1–23. https://doi.org/10.1080/10584609.2020.1723752

Van Atteveldt, W., & Peng, T.-Q. (2018). When communication meets computation: Opportunities, challenges, and pitfalls in computational communication science. *Communication Methods and Measures*, *12*(2-3), 81–92. https://doi.org/10.1080/19312458.2018.1458084

Wijffels, J. (2020). *BTM: Biterm topic models for short text*. https://CRAN.R-project.org/package=BTM

Yi, X., & Allan, J. (2008). Evaluating topic models for information retrieval. *Proceedings of the 17th Acm Conference on Information and Knowledge Management*, 1431–1432. https://doi.org/10.1145/1458082.1458317