

## The Validity of Sentiment Analysis: Comparing Manual Annotation, Crowd-Coding, Dictionary Approaches, and Machine Learning Algorithms

Wouter van Atteveldt, Mariken A. C. G. van der Velden & Mark Boukes

**To cite this article:** Wouter van Atteveldt, Mariken A. C. G. van der Velden & Mark Boukes (2021) The Validity of Sentiment Analysis: Comparing Manual Annotation, Crowd-Coding, Dictionary Approaches, and Machine Learning Algorithms, *Communication Methods and Measures*, 15:2, 121-140, DOI: [10.1080/19312458.2020.1869198](https://doi.org/10.1080/19312458.2020.1869198)

**To link to this article:** <https://doi.org/10.1080/19312458.2020.1869198>



© 2021 The Author(s). Published with license by Taylor & Francis Group, LLC.



Published online: 28 Jan 2021.



[Submit your article to this journal](#)



Article views: 23882



[View related articles](#)






[View Crossmark data](#)



Citing articles: 21 [View citing articles](#)

# The Validity of Sentiment Analysis: Comparing Manual Annotation, Crowd-Coding, Dictionary Approaches, and Machine Learning Algorithms

Wouter van Atteveldt <sup>a</sup>, Mariken A. C. G. van der Velden <sup>a</sup>, and Mark Boukes <sup>b</sup>

<sup>a</sup>Department of Communication Science, Vrije Universiteit Amsterdam; <sup>b</sup>Department of Communication Science, Amsterdam School of Communications Research (ASCoR), University of Amsterdam

## ABSTRACT

Sentiment is central to many studies of communication science, from negativity and polarization in political communication to analyzing product reviews and social media comments in other sub-fields. This study provides an exhaustive comparison of sentiment analysis methods, using a validation set of Dutch economic headlines to compare the performance of manual annotation, crowd coding, numerous dictionaries and machine learning using both traditional and deep learning algorithms. The three main conclusions of this article are that: (1) The best performance is still attained with trained human or crowd coding; (2) None of the used dictionaries come close to acceptable levels of validity; and (3) machine learning, especially deep learning, substantially outperforms dictionary-based methods but falls short of human performance. From these findings, we stress the importance of always validating automatic text analysis methods before usage. Moreover, we provide a recommended step-by-step approach for (automated) text analysis projects to ensure both efficiency and validity.

## Key words

Sentiment Analysis; Manual Annotation; Automated Approaches; Measurement; Validity; Evaluation

*Sentiment* (or tone) of communication is a central topic for scholars of communication (Lengauer et al., 2012). Sentiment has been studied in news coverage of politicians (Dunaway et al., 2015; Hopmann et al., 2011; Vargo et al., 2014), news coverage of political elections (Kleinnijenhuis et al., 2007, 2019; McCombes et al., 2000), political campaigns (Cho, 2013; Haselmayer, 2019; Nai & Martínez i Coma, 2019; Ridout & Searles, 2011; Shah et al., 2007), political referendums (Elenbaas & De Vreese, 2008), political debates (Connaughton & Jarvis, 2004; Hopmann et al., 2011; Nagel et al., 2012), and to analyze the rhetoric of political elites in parliament and manifesto's (Kosmidis et al., 2019; Rheault et al., 2016; Rhodes & Vayo, 2019) – to name a few topics in the last two decades. Beyond the domain of politics, a wide variety of communication scholars used sentiment to study objects, such as the quality of mediated inter-group contact (Wojcieszak & Azrout, 2016), the hostile media effect (Shin & Thorson, 2017), news coverage of wars (Aday, 2010), news coverage of Asia in Asian and Western TV stations (Natarajan & Xiaoming, 2003), coverage about companies (Jonkman et al., 2020), media violence and aggression (Martins et al., 2013), health news coverage (Kim, 2015), the content of news websites (Valenzuela et al., 2017) and the user-comments under their articles (Muddiman & Stroud, 2017), gender differences in news reporting (Rodgers &

Thorson, 2003), news frames on biotechnology (Matthes & Kohring, 2008), or criticisms of news media (Domke et al., 1999).

While omnipresent as a concept, measuring sentiment – or any of the often co-occurring concepts, such as emotionality, negativity, polarity, subjectivity, tone, or valence – is not straightforward. Sentiment is generally expressed with ambiguous and creative language (Liu, 2012; Pang & Lee, 2008; Wiebe et al., 2004). In addition, sentiment analysis in the social sciences suffers from a lack of agreed-upon conceptualization and operationalization (Kleinnijenhuis, 2008; Lengauer et al., 2012).

Computational approaches to sentiment analysis have the potential to remedy the problems of scalability and replicability inherent in manual coding (For overviews, see for example, Boumans & Trilling, 2016; Van Atteveldt et al., 2019; Welbers et al., 2017). While these methods can be very cost-efficient, their application is not without pitfalls (Grimmer & Stewart, 2013; Hilbert et al., 2019; Margolin, 2019; Van Atteveldt & Peng, 2018; Wilkerson & Casas, 2017). Off-the-shelf dictionaries are developed for, and typically validated on, a specific task and domain, and often do not perform well on other tasks. For example, Young and Soroka (2012a) found that different dictionaries for measuring sentiment “show stunningly little overlap” (p. 211) and do not correlate well with each other or with expert annotations. Similarly, machine learning models that are optimized for a certain task (such as distinguishing positive film reviews from negative ones) can give misleading results for social science research by identifying spurious patterns in the data that was used to train these algorithms (Thelwall et al., 2012). Nevertheless, the current proliferation of available dictionaries and other off-the-shelf tools tends to overshadow the low measurement validity (González-Bailón & Paltoglou, 2015; Soroka et al., 2015), and in many cases these tools are not validated before being used on new tasks. On top of these challenges, (semi-)automated tools for non-English languages are rare, hampering comparative research of communication (Haselmayer & Jenny, 2017).

This paper’s contribution lies in demonstrating which, if any, of the sentiment analysis methods actually work well for determining the tone of media coverage. We use a triple-coded gold standard validation set to compare different manual and automatic approaches to sentiment analysis of Dutch economic news headlines. Using a Dutch-language case study gives an impression of the state of the art in non-English language tools as well as allowing us to test the efficacy of machine translation for comparative research. Moreover, we presume that journalistic descriptions of whether the economy is doing well or not are relatively factual and unambiguous. Therefore, the economy seems to be an “easy” test for automated methods compared to many aspects of political news, where opinions and sentiment are often expressed in more nuanced or creative ways.

Headlines are often relatively simple and explicit compared to the more nuanced arguments made in the article body. Thereby, focusing only on headlines avoids the issue of articles citing multiple sources that can each express a different sentiment, which makes it difficult to judge the “overall” sentiment of an article. Nevertheless, the limited amount of words in a headline could put automatic methods at a disadvantage since it is less likely that any given dictionary or machine learning model will contain a matching word. The sentiment of headlines, however, is interesting in its own right as the headline is the first (and sometimes the only) part of an article that people read, and headlines shape the context in which people read the article. In social science research, headlines are seen as an important framing device (e.g., Liu et al., 2019; Tankard et al., 2001), a signal of news values (e.g., Ng & Zhao, 2020) and as an important predictor of effects on polarization, political attitudes, and consumer confidence (e.g., Blood & Phillips, 1995; Munger et al., 2020; Narayan & Narayan, 2017). Moreover, many other forms of communication that are studied in our field (and with automated content analysis), such as tweets, chat messages, and online comments, are also relatively short. Thus, evaluating methods that determine sentiment of headlines is a relevant task for communication science. In addition, it is a suitable case for sentiment analysis in general, even though results might differ for other forms of communication such as full articles.

For this study, we used the existing manual annotations presented in Boukes et al. (2020). Besides comparing the gold standard to these manual codings and to crowd coding data that were collected specifically for this article, we used the manual codings as training data for classical and “deep learning” machine learning models. Finally, we apply many of the different dictionaries that have recently become available, including English-language dictionaries using machine translation of the headlines (e.g., as suggested by De Vries et al., 2018) and a dictionary specifically customized to the domain of interest (suggested by Muddiman et al., 2019). Thereby, our paper presents a comprehensive review of existing methods to measure sentiment. We apply an open science, open materials approach. All our analytical code, data, and results are published in the online compendium for this paper.<sup>1</sup>

Generally, results from this case study do not warrant optimism regarding the possibilities of automatic sentiment analysis, especially dictionary based analyses. In line with e.g., Soroka et al. (2015) and Boukes et al. (2020), we find that off-the-shelf dictionaries perform poorly on this task. Machine learning approaches, especially deep learning, performs considerably better than the off-the-shelf sentiment analysis tools, but still do not reach the level of validity generally required for text analysis methods. On a more positive note, the results of crowd coding can compete with the quality delivered by trained coders, providing a cheaper and especially more transparent and replicable alternative to expensive student coders. The main take-home message is, however, that a human eye is still required to guarantee the validity of measuring sentiment in content analysis. As detailed in the step-by-step process given at the end of this article, we strongly recommend that every automatic text analysis project should start with coding a validation set.

## Existing methods of sentiment analysis

Traditionally, sentiment is measured using manual annotators and a codebook (for examples, see Aday, 2010; Cho, 2013; Dunaway et al., 2015; Elenbaas & De Vreese, 2008; Kleinnijenhuis et al., 2007, 2019; Martins et al., 2013; McCombes et al., 2000; Meijer & Kleinnijenhuis, 2006; Muddiman & Stroud, 2017; Nagel et al., 2012; Natarajan & Xiaoming, 2003; Rodgers & Thorson, 2003; Shah et al., 2007). Manual coding is expensive, however, and even when using extensive training programs high levels of reliability are not always achieved (Weber et al., 2018).

One possible solution is to use crowd coding platforms instead of traditional expert (or undergraduate) coders (Benoit et al., 2016; Haselmayer & Jenny, 2017; Lind et al., 2017). Sentiment analysis, in particular, can be relatively easily reduced to simple questions suitable for crowd coding. In addition, the lower costs of crowd coding allow for each document to be coded by multiple coders. This not only increases reliability, but also gives an indication of the ambiguity of sentiment. Finally, the crowd only sees the material as presented in the crowd coding platform, which can be easily shared between researchers. This means that crowd coding is more transparent and replicable than traditional manual coding.

Another option is to use automatic sentiment analysis methods: Using a computer program to classify each document as being positive, neutral, or negative (for overviews of these methods, see Boumans & Trilling, 2016; Grimmer & Stewart, 2013; Hopkins & King, 2010; Nunez-Mir et al., 2016; Van Atteveldt et al., 2019; Welbers et al., 2017; Wilkerson & Casas, 2017). There are broadly two types of methods that can be used for this: dictionaries and supervised machine learning.

One could (naively) think that measuring sentiment using a dictionary should be easy: Make a long list of positive and negative words and count how many words of each category occur. Indeed, in the last decade we have seen a proliferation of available *off-the-shelf* dictionaries for sentiment analysis (González-Bailón & Paltoglou, 2015; Soroka et al., 2015). Studies within computational linguistics, however, suggest that coming up with the right set of keywords might be less trivial than one initially thinks (for an overview, see Pang & Lee, 2008). Applying one or another dictionary to

investigate a specific research question could lead to widely divergent conclusions (Boukes et al., 2020; González-Bailón & Paltoglou, 2015; Soroka et al., 2015; Young & Soroka, 2012a).

In general, it turns out that the validity of sentiment analysis highly depends on the domain, genre and language to which it is applied (González-Bailón & Paltoglou, 2015; Pang & Lee, 2008; Soroka et al., 2015; Thelwall et al., 2012). Therefore, scholars have started to apply (supervised) machine learning methods to classify texts. In contrast to the off-the-shelf dictionaries, machine learning can account for the peculiarities of the texts under study. In machine learning, rather than having the researcher specify this link explicitly using a dictionary, the computer uses manually coded *training data* to learn the link between input features (e.g., words as independent variables) and the desired output label (e.g., sentiment as dependent variable). Technically, a machine learning algorithm is used to create a statistical model based on the training data which is then used to predict the sentiment of unlabeled texts. Machine learning generally outperforms dictionary-based methods, and most state-of-the-art sentiment analysis systems developed in computational linguistics are now based on machine learning (Rosenthal et al., 2017). These models generally have a very high number of parameters to be estimated, for example, a score for every unique word in the training data. Since this can easily lead to overfitting of the model, its performance should always be judged on separate validation data (out of sample prediction).

Initially, most machine learning applications used word frequencies – also called “bag-of-words” – as input features. This approach ignores all aspects of word order and grammar. More recently, so-called “deep learning” models have been developed that use neural networks with many hidden layers to overcome this limitation (Goldberg, 2017). This is generally used in combination with word embedding models (Mikolov et al., 2013; Rudkowsky et al., 2018). These models used very large quantities of unlabeled texts (i.e. without manual codings) to learn the overall meaning of a word. This reduces the number of distinct input features and allows words that were not present in the initial data to be captured in the model.

Each of the methods surveyed here has its own advantages and disadvantages. Although machine learning methods generally outperform dictionaries in state-of-the-art systems, their performance strongly depends on the availability of sufficient training material. Dictionaries are more transparent and easier to apply. All automated methods, however, will decrease in performance when applied to a task or domain that is different from the one it was developed for. This makes it difficult to estimate beforehand which method will be the most effective or what its performance will be. The difficulty of ex-ante estimation stresses the importance of validating each method for its specific task.

## Data and methods

By comparing various methods of sentiment analysis, this paper aims to help researchers choose the best method for approaching a sentiment analysis project. The starting point of this analysis are the data reported in Boukes et al. (2020). These authors collected news from a total of ten newspapers and five websites using an extensive search string that covers a wide variety of economic and financial issues published between February 1 and July 7, 2015. It included three quality newspapers (*NRC Handelsblad*, *Trouw*, *de Volkskrant*), a financial newspaper (*Financieel Dagblad*), three popular newspapers (*Algemeen Dagblad*, *Metro*, *De Telegraaf*) and three regional outlets (*Dagblad van het Noorden*, *de Gelderlander*, *Noordhollands Dagblad*). For the automated text analyses, all headlines were preprocessed by lemmatizing them using the Dutch lemmatizer Frog (Van den Bosch et al., 2007). In short, lemmatizing means that we reduce a verb like *dacht* (thought) to its lemma *denken* (think).

### **Gold standard**

To guarantee the quality of our comparisons, we have created a gold standard against which we evaluate the existing methods by manually annotating a selection of the headlines from the manually annotated data of Boukes et al. (2020). We randomly selected headlines that were annotated multiple times and that were annotated as being about the economy. Krippendorff (2012, p. 240) recommends at least 143 documents for determining intercoder reliability for this case ( $P_c = .33; \alpha \geq .8; \text{sig.} \leq .005$ ), but to be conservative we decided to use 300 headlines. These headlines were annotated by the three authors using the instructions from the original article. The initial inter-coder agreement of this coding was Krippendorff's  $\alpha = 0.75$ , with agreement between coder pairs ranging from 0.75 to 0.82, which reflects the subjectivity of the task at hand. All disagreements were then discussed between the authors and resolved where possible. In most cases, these disagreements were caused by simple clerical errors or by a misinterpretation of the sentence or coding rules. Eleven headlines were removed after discussion as they were deemed to be inherently ambiguous (i.e., not ideal for a gold standard). If anything, removing the hardest headlines should make the task slightly easier for the different forms of automatic sentiment analysis.

### **Manual coding**

As reported in Boukes et al. (2020), the sentiment of newspaper and website headlines was manually annotated by a team of 22 student coders. The coders were trained by means of two training sessions of three hours, and three homework assignments. In addition, they could send their questions or doubts by e-mail and receive almost immediate feedback.

To verify whether articles really dealt with economic news, the first question in the code-book was whether the headline or first paragraph of the item referred to the economy, economic developments or an economic topic (e.g., inflation, unemployment, interest rates, or the housing market). Subsequently, coders were asked to evaluate the sentiment of a headline with regards to the economy on a three point-scale: (−1) negative; (0) neutral, ambiguous, or mixed; and (+1) positive. Coders were explicitly instructed to only evaluate the headline and to not consider any information that they already had seen from the full text. Inter-coder reliability was assessed on sample of 148 randomly selected news items that were analyzed by at least three of the coders; on average this were 5.63 coders. This sentiment measurement proved to have a satisfactory inter-coder reliability (Krippendorff's  $\alpha = .80$ ).

### **Crowd coding**

The headlines from the gold standard were all coded by five crowd coders using the Figure 8 platform.<sup>2</sup> Coders received short instructions with a limited amount of examples (see online appendix for the task definition). Besides these 300 sentences, 70 (relatively straightforward) test sentences were included for which we provided the correct answer. These were used first in an initial quiz to ensure coders understood the instructions, and after that one test question was included in every page of five target sentences to ensure that coders remained concentrated during the task. Coders that missed test questions were informed of this, and coders missing more than 70% of test questions were excluded.

### **Sentiment dictionaries**

In this paper, we partly replicate the dictionaries used in Boukes et al. (2020) and add specific dictionaries for *hope* and *fear* in the economic context. Specifically, we applied the Dutch ANEW (Bradley & Lang, 1999), Pattern (Smedt & Daelemans, 2012) and Polyglot (Richardson et al., 2018)



dictionaries using the Python code published with that paper. In addition, we used the R package *Quanteda* (Benoit et al., 2018) to add (a) the dictionary developed by Damstra and Boukes (2018) to measure the sentiment of economic news; (b) the NRC Emotion Lexicon (Mohammad & Turney, 2013) for both measuring positive and negative words as well as trust and fear words; and (c) a customized dictionary based on the approach suggested by Muddiman et al. (2019). The upper part of Table 1 displays the number of words per category in the Dutch dictionaries. For the dictionaries applied using *Quanteda* (D2, D3, and D4), the table also lists the categories that were used for positive and negative words. For these dictionaries, a headline was counted as positive if there were more positive than negative words, and similarly for negative. Headlines without any sentiment words or with equal positive and negative words were treated as neutral.

### English dictionaries using machine translation

The number of Dutch dictionaries available is rather limited. There are many more dictionaries for sentiment analysis available in English, including also domain-specific dictionaries for sentiment in finance. Following the suggestion by De Vries et al. (2018), we used machine translation to translate the gold standard texts. For this, we used both Google Translate and DeepL to translate the gold standard texts.<sup>3</sup> This allowed us to make use of the plenitude of English sentiment dictionaries available using the *Quanteda* R packages (Benoit et al., 2018): First, we used the Affective Norms for English Words (AFINN) (Bradley & Lang, 1999), a publicly available list of English words rated for valence with values between  $-5$  (negative) and  $+5$  (positive). The version implemented in the *Quanteda* package uses a binary classification. Second, we used the Augmented General Inquirer<sup>4</sup> *Positiv* and *Negativ* dictionary. Third, we used the dictionary from Hu and Liu (2004; Liu et al., 2005). Fourth, we applied the 2014 version of the Loughran and McDonald Sentiment Word Lists

**Table 1.** Information on off-the-shelf dictionaries.

| Dictionary                  | Category             | # Words |
|-----------------------------|----------------------|---------|
| <i>Dutch Dictionaries</i>   |                      |         |
| D1: DANEW                   | (continuous scale)   | 4299    |
| D2: DamstraBoukes           | Hope                 | 30      |
|                             | Fear                 | 33      |
| D3: Muddiman                | Positive             | 32      |
|                             | Negative             | 38      |
| D4: NRC:                    | Positive + Trust     | 3162    |
|                             | Negative + Fear      | 4037    |
| D5: Pattern                 | Positive (valence>0) | 1554    |
|                             | Negative (valence<0) | 2364    |
| D6: Polyglot                | Positive             | 1502    |
|                             | Negative             | 2474    |
| <i>English Dictionaries</i> |                      |         |
| E1: AFINN                   | Positive (valence>0) | 878     |
|                             | Negative (valence<0) | 1599    |
| E2: DamstraBoukes           | Hope                 | 30      |
|                             | Fear                 | 33      |
| E3: GenInq                  | Positiv              | 1653    |
|                             | Negativ              | 2010    |
| E4: HuLiu                   | Positive             | 2006    |
|                             | Negative             | 4783    |
| E5: LoughranMcDonald        | Positive             | 354     |
|                             | Negative             | 2355    |
| E6: LSD                     | Positive             | 1709    |
|                             | Negative             | 2858    |
| E7: Muddiman                | Positive             | 32      |
|                             | Negative             | 38      |
| E8: NRC                     | Positive + Trust     | 3543    |
|                             | Negative + Fear      | 4800    |
| E9: RID                     | Positive Affect      | 70      |
|                             | Anxiety              | 49      |

(Loughran & McDonald, 2011). Fifth, we employed the Martindale's Regressive Imagery Dictionary (RID) (Martindale, 1975, 1990). Sixth, we used the Lexicoder Sentiment Dictionary (LSD) (Young & Soroka, 2012a, 2012b). The seventh and eighth dictionaries are the NRC Emotion Lexicon, but in English this time, and the translated Damstra and Boukes (2018) dictionary similar to the one applied in Dutch. The lower-part of Table 1 displays the used categories and number of words per category in the English dictionaries.

## Machine learning

This paper uses two types of machine learning: “classical” machine learning with a Naive Bayes (NB) and Support Vector Machine (SVM) classifier, and “deep” learning using a Convolutional Neural Network (CNN). The setup and training procedure for both models are given below. For all models, we used 6,038 manually coded headlines from Boukes et al. (2020) as training data. The final models were validated against the 300 headlines gold standard.<sup>5</sup>

For the NB and SVM model, we used *scikit-learn* (Pedregosa et al., 2011) to create a document-term matrix with normalized td-idf weights, and train and test the model. To determine the hyperparameters for the SVM model (regularization parameter and kernel type and coefficient) we performed a grid search using 5-fold crossvalidation.<sup>6</sup> The best performing model was then trained on all training data and tested on the validation set.

For the deep learning model, we chose to use a Convolutional Neural Network (CNN), which allows for local interactions between word meanings. That is, scores are computed for windows of adjacent words, so the model can treat word combinations differently from the underlying words. Given the relatively small length of newspaper headlines, we decided not to use more complicated models (such as Long-Short Term Memory or LSTM models) that also allow non-local interactions (Goldberg, 2017). Specifically, we used *keras* with *tensorflow* back-end (Abadi et al., 2016) to train and test the CNN consisting of the following layers:

- (1) An *embeddings* layer using the Amsterdam Embeddings Model trained on Dutch news

(Kroon et al., 2019). This layer looks up each word in the input and replaces it by its 320-dimensional embedding vector representing its position in a latent semantic space.

- (2) A *convolution* layer that concatenates the embeddings for each 3-word window and transforms them into a lower-dimensional representation using a dense neural layer, effectively allowing for features to be created spanning at most 3 words.
- (3) A *max-pooling* layer that maximizes the value for each feature for each document.
- (4) A regular *dense network* that predicts the sentiment from the pooled input features.

The architecture above is a relatively standard architecture for document classification.

However, there are still many choices (hyperparameters) left, from the depth and size of the dense network to the learning rate and loss function. Since again there are no strong theoretical grounds to determine these parameters, we conducted a grid search using cross-validation within the training to find the optimal parameter values similar to the procedure for SVM. These settings were then used to train a new model on all training data, which was validated on the gold standard data.

## Results

Table 2 shows the overall performance of all tested methods, listed as percentage agreement (acc), Krippendorff's  $\alpha$  for ordinal measures, as well as the precision, recall, and F1 score for both positive, neutral, and negative sentiment. These latter scores are given since they are the standard for



**Table 2.** Overall performance of the tested sentiment analysis approaches.

| section              | name             | Acc. | alpha | Positive |      |      | Neutral |      |      | Negative |      |      |
|----------------------|------------------|------|-------|----------|------|------|---------|------|------|----------|------|------|
|                      |                  |      |       | Pr.      | Re.  | F1   | Pr.     | Re.  | F1   | Pr.      | Re.  | F1   |
| Manual Coding        | Single Coder     | 0.82 | 0.82  | 0.88     | 0.86 | 0.87 | 0.76    | 0.81 | 0.78 | 0.84     | 0.80 | 0.82 |
| Manual Coding        | Vote (3 Coders)  | 0.88 | 0.90  | 0.97     | 0.91 | 0.94 | 0.82    | 0.88 | 0.85 | 0.87     | 0.84 | 0.86 |
| Crowd-Coding         | Single Coder     | 0.72 | 0.75  | 0.69     | 0.84 | 0.76 | 0.69    | 0.58 | 0.63 | 0.78     | 0.78 | 0.78 |
| Crowd-Coding         | Vote (3 Coders)  | 0.77 | 0.81  | 0.73     | 0.89 | 0.80 | 0.74    | 0.65 | 0.69 | 0.83     | 0.81 | 0.82 |
| Crowd-Coding         | Vote (5 Coders)  | 0.77 | 0.81  | 0.73     | 0.90 | 0.81 | 0.73    | 0.65 | 0.69 | 0.84     | 0.80 | 0.82 |
| Machine Learning     | CNN              | 0.63 | 0.50  | 0.68     | 0.49 | 0.56 | 0.58    | 0.78 | 0.66 | 0.72     | 0.57 | 0.63 |
| Machine Learning     | NB               | 0.58 | 0.39  | 0.74     | 0.34 | 0.47 | 0.52    | 0.83 | 0.64 | 0.65     | 0.47 | 0.55 |
| Machine Learning     | SVM              | 0.57 | 0.41  | 0.69     | 0.37 | 0.48 | 0.52    | 0.79 | 0.62 | 0.64     | 0.48 | 0.55 |
| Dictionaries         | DANEW            | 0.42 | 0.10  | 0.75     | 0.08 | 0.15 | 0.40    | 0.97 | 0.57 | 0.80     | 0.04 | 0.08 |
| Dictionaries         | DamstraBoukes    | 0.41 | 0.05  | 0.83     | 0.07 | 0.13 | 0.40    | 0.99 | 0.57 | 0.00     | 0.00 | 0.00 |
| Dictionaries         | Muddiman         | 0.49 | 0.31  | 0.53     | 0.38 | 0.44 | 0.46    | 0.64 | 0.53 | 0.53     | 0.39 | 0.45 |
| Dictionaries         | NRC              | 0.47 | 0.32  | 0.39     | 0.53 | 0.45 | 0.46    | 0.44 | 0.45 | 0.59     | 0.46 | 0.52 |
| Dictionaries         | Pattern          | 0.39 | 0.07  | 0.43     | 0.08 | 0.14 | 0.39    | 0.90 | 0.54 | 0.38     | 0.03 | 0.06 |
| Dictionaries         | Polyglot         | 0.42 | 0.26  | 0.38     | 0.32 | 0.34 | 0.39    | 0.55 | 0.45 | 0.53     | 0.33 | 0.41 |
| English Dictionaries | AFINN            | 0.43 | 0.27  | 0.35     | 0.38 | 0.37 | 0.40    | 0.50 | 0.45 | 0.58     | 0.38 | 0.46 |
| English Dictionaries | DamstraBoukes    | 0.42 | 0.07  | 0.67     | 0.08 | 0.15 | 0.40    | 0.98 | 0.57 | 1.00     | 0.02 | 0.04 |
| English Dictionaries | GenInq           | 0.41 | 0.26  | 0.31     | 0.37 | 0.34 | 0.38    | 0.38 | 0.38 | 0.54     | 0.47 | 0.51 |
| English Dictionaries | HuLiu            | 0.46 | 0.34  | 0.40     | 0.30 | 0.34 | 0.42    | 0.63 | 0.50 | 0.65     | 0.40 | 0.50 |
| English Dictionaries | LoughranMcDonald | 0.50 | 0.29  | 0.50     | 0.14 | 0.22 | 0.46    | 0.79 | 0.58 | 0.62     | 0.43 | 0.51 |
| English Dictionaries | LSD              | 0.46 | 0.33  | 0.39     | 0.40 | 0.39 | 0.42    | 0.54 | 0.48 | 0.62     | 0.41 | 0.50 |
| English Dictionaries | Muddiman         | 0.48 | 0.27  | 0.48     | 0.38 | 0.43 | 0.46    | 0.71 | 0.55 | 0.57     | 0.30 | 0.39 |
| English Dictionaries | NRC              | 0.42 | 0.23  | 0.34     | 0.62 | 0.44 | 0.43    | 0.32 | 0.37 | 0.57     | 0.39 | 0.46 |
| English Dictionaries | RID              | 0.42 | 0.06  | 0.00     | 0.00 | 0.00 | 0.41    | 0.97 | 0.57 | 0.82     | 0.09 | 0.16 |

Acc.: Accuracy expressed as percentage correct;  $\alpha$ : Krippendorff's alpha (ordinal), Pr.: Precision for the specified class (Positive/Neutral/Negative); Re.: Recall for that class; F1: F1-Score for that class. When more than one measurement or prediction (respectively for manual/crowd annotation and machine learning) was available, we averaged the score.

performance evaluation in machine learning, but they also give more insight into the type of error made by an algorithm. For example, a dictionary with only a few very clear sentiment words can have high precision but low recall, meaning that when it identifies a document as positive or negative it is generally correct (precision), but that it misses a lot of the documents that were actually positive or negative (recall) because these documents did not contain the words in the dictionary. The F1 score is the harmonic mean between precision and recall and will generally be closest to the lower of the two scores. Since Krippendorff's  $\alpha$  is the most commonly used measure for the validity of text analysis in communication science, we will mostly report performance using that measure, but overall accuracy,  $\alpha$ , and F1 scores show the same pattern for all cases.

Starting with the top rows of [Table 2](#), manual coding (using undergraduate students) yields the best results, with both accuracy and alpha above .8. Particularly, using three different students as manual annotators in combination with a majority vote to determine the final sentiment of the sentence achieves the highest performance scores, although this is of course expensive for large scale projects.

A good second place is taken by crowd coding – rows 3 to 5 in [Table 2](#). Using just one crowd coder already yields a reasonably good performance ( $\alpha \geq 0.75$ ). This can be further improved by applying a majority vote decision when using multiple crowd coders (an increase to  $\alpha \geq 0.8$ ).

Moving from manual annotation to the realm where “computers do the work”, [Table 2](#) demonstrates that machine learning performs worse than both students' manual coding and crowd coding. Reaching  $\alpha = 0.50$  for deep learning (CNN) and slightly worse for classical machine learning (SVM;  $\alpha = 0.41$ , NB;  $\alpha = 0.40$ ), machine learning still performs significantly better than chance. However, since these results are lower than generally accepted levels of inter-coder reliability, these models cannot directly be used for substantive analyses.

Finally, the bottom half of [Table 2](#) shows the performance of Dutch dictionaries and English dictionaries applied to machine translated texts. These methods perform worse than the machine learning results and much worse than manual annotation. Most overall performance scores of the dictionaries approximate chance agreement for three categories (i.e. positive, negative and neutral).

Ultimately, Hu and Liu (2004) scores best on inter-coder reliability ( $\alpha = 0.34$ ), although Loughran and McDonald (2011) does better when considering percentage agreement (50%), possibly because it is more prone to confuse positive and negative, which is penalized more than confusing with neutral in Krippendorff's ordinal measurement.

Some dictionaries seem reasonably able to measure both positive and negative sentiment looking at the precision scores – e.g., the custom made dictionary of Damstra and Boukes (2018) reaches a precision score of 0.83 for Martindale (1975, 1990)'s RID reach equally high levels of precision for negativity. The low levels of performance of dictionaries stems from extremely low scores on the recall measure. Thus, relying on these dictionaries yields many *false negative* cases, also referred to as type II errors.

### Trade-off between coverage and accuracy

As explained above, crowd coding using multiple coders achieved results that are very close to coding by trained students, presenting researchers with a more affordable option. This raises the question of how many crowd coders should code each article? To answer this question, Figure 1 shows the relation between performance in percentage accuracy and coverage, i.e. what percentage of all articles can be coded at a certain reliability. The highest point of the solid line (5/5) indicates that

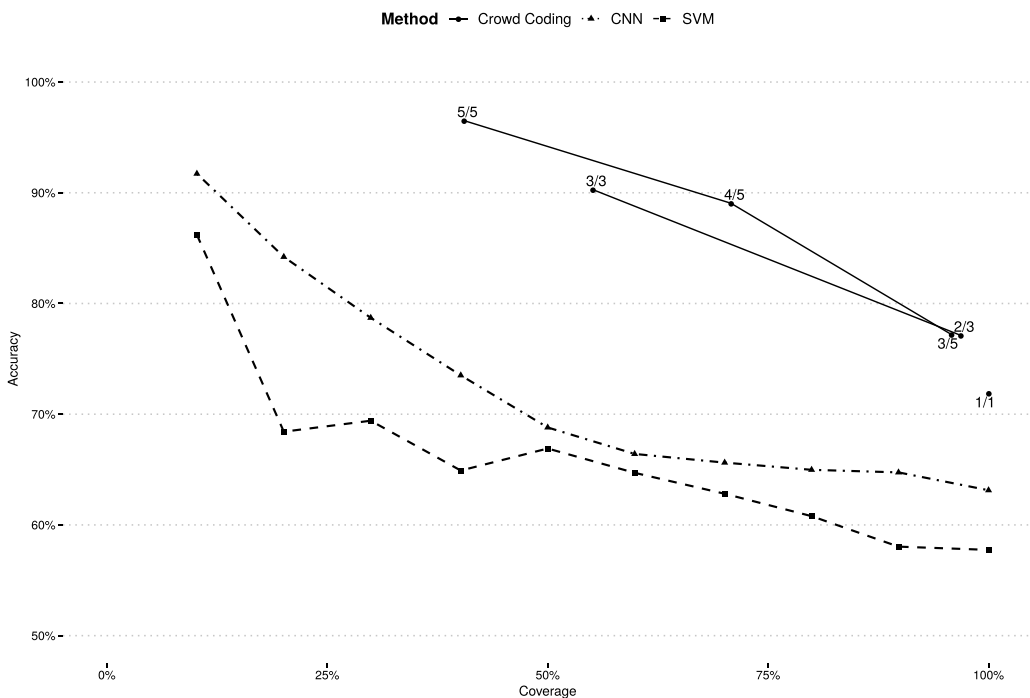


Figure 1. Coverage vs accuracy for crowd coding and machine learning.

Note: For Machine Learning (CNN and SVM), points show the percentage accuracy on the subset of documents on which the algorithm was most confident. For example, the second dash-dotted triangle shows that the CNN model was accurate in about 84% of cases for the 20% of documents on which it was most confident. For Crowd Coding, the lines indicate how many coders were used, and each point represents a majority vote of at least that size. For example, the point '4/5' shows that if at least 4 out of 5 coders agreed on a label, they were correct in almost 90% of cases (accuracy), but only just under 75% of documents were coded with this majority (coverage).

five out of five crowd coders agreed on approximately 40% of the sentences (i.e. coverage). In this case, the coders achieved an accuracy of  $> 97\%$ , or near-perfect coding. Utilizing three coders and having them all agree (3/3) leads to 93% accuracy for over half of the sentences – i.e. coverage  $> 50\%$ . Hence, when all three coders agree, the researcher can be very confident that the score is correct. Looking at the majority-vote scenario – i.e. 4/5, 3/5, or 2/3 – Figure 1 presents an increase in coverage: From 75% in a 4/5 scenario to almost 100% coverage using 3/5, or 2/3. The performance varies between just under 90% and 75% for the respective majority vote scenarios.

The results of Figure 1 suggest that the best strategy is to code all texts with two coders initially, and use a third coder on the  $< 30\%$  of sentences that they disagree on. Besides giving a high overall validity, this also gives some measure of which texts were more difficult to code, giving an indication of the ambiguity of the sentiment in these texts. Adding more coders (up to the five tested here) does not significantly increase overall performance, but does give a better idea of the spread or uncertainty of the annotations, as sentences on which 4 out of 5 agree score significantly better than the ones on which only 3 out of 5 agree, and sentences on which all 5 coders agree are almost certain to be correct.

The bottom lines of Figure 1 show a similar trade-off for machine learning. Where the size of the majority vote in crowd coding gives an indication of the uncertainty of the coded sentiment, machine learning models give a *confidence* score for each prediction. Thus, we can choose to use only the predictions on which the algorithm was sufficiently confident, and for example, use manual coding for the more difficult sentences. As shown in the figure, an accuracy of almost 80% can be achieved by CNN for the 30% of cases on which it was most confident, and on the 50% of cases it scored just under 70%.

### Correlation between various dictionaries

Table 3 demonstrates the correlations between the Dutch and English dictionaries applied in this paper. The cell shading in the table indicates the strength of the correlation. Most correlations between the tested Dutch dictionaries are weak to moderate, ranging from 0.00 (Pattern and NRC) and 0.40 (Muddiman approach and NRC). The dictionaries' correlation with the gold standard is even lower – varying between 0.12 for Pattern and 0.33 for NRC.

Table 3 also shows the correlations among the English dictionaries – indicated by *E* in Table 3 – and between the Dutch and English versions dictionaries. Again, the correlations with the gold standard are low, varying between 0.11 for RID (Martindale, 1975, 1990) and 0.34 for both Hu and Liu (2004) and the Lexicoder Sentiment Dictionary (LSD) of Young and Soroka (2012a, 2012b). Interestingly, correlations between the English dictionaries are substantially higher than for the Dutch dictionaries. It remains to be seen whether this is because the dictionaries we used are indeed more similar, or whether the translation introduced artifacts.

Overall, this table shows that the various dictionaries all measure something else, and none of them can be seen as a valid measurement of the sentiment as defined in our gold standard.

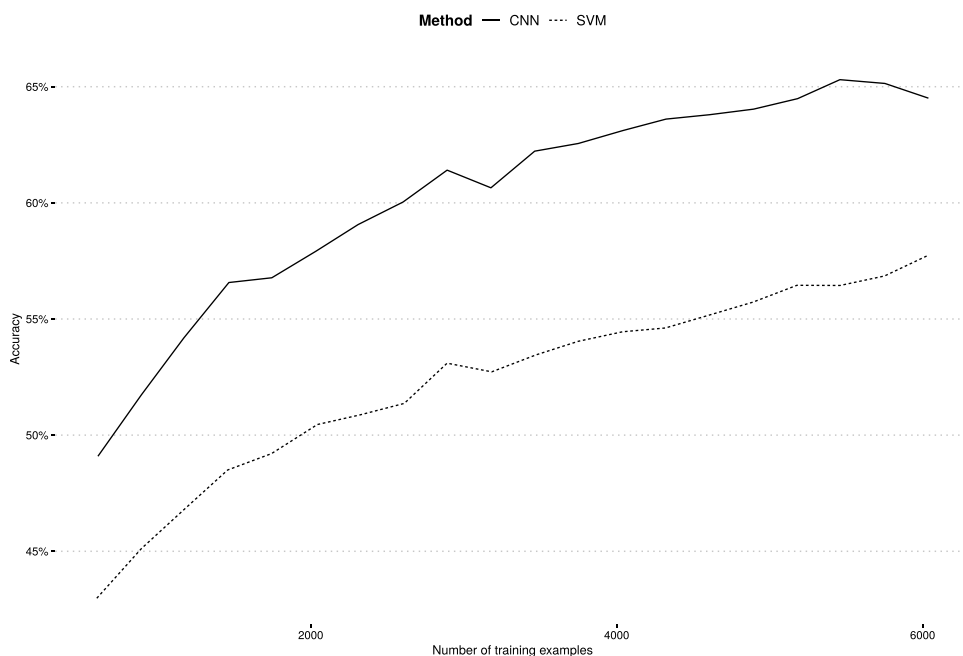
### Learning curve of machine learning

Finally, Figure 2 shows the so-called “learning curve” of both machine learning algorithms. This figure shows the increase of the algorithms' performance as the number of training examples (i.e. more new sentences) increases. The learning curve was estimated by training on random subsets of the training data and testing against the gold standard, and gives an indication of how many training documents are needed to achieve a given performance. This curve shows that, as expected, both methods improve most quickly at the beginning, signaled by a steep incline, before leveling off to a presumed asymptote, where adding more training examples no longer increases their performance. The climbing lines for both algorithms indicate that presumably neither method is saturated yet, assuming the slight downturn at the end of the CNN curve is an anomaly. Thus, one can expect that the performance of both methods will profit from more training data. However, given that both lines

Table 3. Bi-variate correlations between dictionaries and gold standard.

|   | Gold | D1   | D2    | D3   | D4   | D5    | D6   | E1   | E2   | E3   | E4   | E5   | E6   | E7   | E8   |
|---|------|------|-------|------|------|-------|------|------|------|------|------|------|------|------|------|
| <i>Dictionaries</i>                                 |      |      |       |      |      |       |      |      |      |      |      |      |      |      |      |
| D1: DANEW   | 0.22 |      |       |      |      |       |      |      |      |      |      |      |      |      |      |
| D2: DamstraBoukes                                   | 0.18 | 0.11 |       |      |      |       |      |      |      |      |      |      |      |      |      |
| D3: Muddiman  | 0.32 | 0.23 | 0.24  |      |      |       |      |      |      |      |      |      |      |      |      |
| D4: NRC   | 0.33 | 0.23 | 0.11  | 0.40 |      |       |      |      |      |      |      |      |      |      |      |
| D5: Pattern   | 0.12 | 0.17 | -0.01 | 0.07 | 0.17 |       |      |      |      |      |      |      |      |      |      |
| D6: Polyglot  | 0.27 | 0.18 | 0.11  | 0.20 | 0.35 | 0.21  |      |      |      |      |      |      |      |      |      |
| <i>English Dictionaries (translated using deep)</i> |      |      |       |      |      |       |      |      |      |      |      |      |      |      |      |
| E1: AFINN   | 0.28 | 0.20 | 0.06  | 0.40 | 0.31 | 0.17  | 0.26 |      |      |      |      |      |      |      |      |
| E2: DamstraBoukes                                   | 0.18 | 0.08 | 0.61  | 0.18 | 0.10 | -0.01 | 0.03 | 0.12 |      |      |      |      |      |      |      |
| E3: Genlmg  | 0.26 | 0.21 | 0.03  | 0.30 | 0.23 | 0.16  | 0.23 | 0.51 | 0.09 |      |      |      |      |      |      |
| E4: HuLiu   | 0.34 | 0.18 | 0.20  | 0.32 | 0.34 | 0.20  | 0.37 | 0.50 | 0.23 | 0.49 |      |      |      |      |      |
| E5: LoughranMcDonald                                | 0.30 | 0.20 | 0.05  | 0.35 | 0.35 | 0.17  | 0.27 | 0.54 | 0.07 | 0.39 | 0.48 |      |      |      |      |
| E6: LSD   | 0.34 | 0.23 | 0.20  | 0.33 | 0.36 | 0.25  | 0.33 | 0.61 | 0.17 | 0.53 | 0.61 | 0.47 |      |      |      |
| E7: Muddiman  | 0.28 | 0.18 | 0.07  | 0.56 | 0.35 | 0.10  | 0.19 | 0.38 | 0.03 | 0.38 | 0.35 | 0.27 | 0.38 |      |      |
| E8: NRC   | 0.28 | 0.21 | 0.08  | 0.25 | 0.44 | 0.10  | 0.18 | 0.40 | 0.14 | 0.46 | 0.43 | 0.37 | 0.45 | 0.33 |      |
| E9: RID   | 0.11 | 0.08 | 0.02  | 0.06 | 0.19 | 0.17  | 0.21 | 0.24 | 0.09 | 0.20 | 0.19 | 0.31 | 0.29 | 0.15 | 0.14 |
| Gold  |      | D1   | D2    | D3   | D4   | D5    | D6   | E1   | E2   | E3   | E4   | E5   | E6   | E7   | E8   |

E2 and E7 are machine translated dictionaries, performed on the machine translated texts.



**Figure 2.** Learning curve of machine learning algorithms.

*Note:* Each line shows how quickly more training examples allowed the algorithm to improve performance on the gold standard. For example, with 2000 training documents from the original data of Boukes et al. (2020), the SVM model achieved an accuracy of about 50%, which increased to almost 55% with 4000 examples.

are mostly parallel, there is no indication that the performance of the SVM will converge to the performance of the CNN with an equal amount of training data.

### Error analysis

We conducted an error analysis to improve our understanding of the mistakes made by the various automatic methods.<sup>7</sup> For the error analysis on the off-the-shelf dictionaries, we picked the NRC dictionary, the best performing Dutch dictionary (by alpha) that also has an English translation. Many of the mistakes made in the Dutch NRC dictionary are missed negations. For example positively classifying the word *groei* (growth) in sentences like *afnemende groei hypotheken* (reduced growth in mortgages) or *matige groei* (tepid growth) or negatively classifying the word *crisis* in *Cyprus heft laatste restricties op na crisis* (Cyprus lifts last restrictions after the crisis). In addition, the error analysis revealed clear mistakes, such as misclassifying the word *beurs* (stock exchange) as positive.

The English NRC dictionary applied to headlines translated by deepl shows a similar pattern, with the words *job* and *savings* being misclassified as positive in sentences such as *Best Buy deletes 1500 jobs* and *300 million in savings gone in one day*. The same happened to negative words like *crisis* and *inflation*. To better understand the role of translation, we then looked at the sentences that were correctly classified in Dutch but missed by the translated version. An inspection of these ( $n=54$ ) sentences suggests that, although some are translated a bit clumsily, this does not seem to be the cause of the errors as the translation errors are more concerned with function words rather than the sentiment carrying words (for example, *300 million in savings gone in one day* was actually translated as *300 million in savings in one day away*). An interesting detail is that the word *interest* caused most of the translated misclassifications as the English word *interest* is ambiguous between the supposedly

neutral meaning of interest rate and the positive concept of being interested or interesting, while the Dutch translation (*rente*) is not ambiguous.

For the error analysis of our machine learning approaches, we first look at Naive Bayes as that method has the most interpretable feature weights. Interestingly, the words *faillissement* (bankruptcy) and *werkloosheid* (unemployment) were positive features, presumably because these words are often negated in the training documents. Being based on word frequencies (bag of words), Naive Bayes also suffers from the same lack of context as the dictionaries, for example classifying the sentences *minder woningen onder water* (fewer underwater mortgages) as negative based on negative values for both *fewer* and *underwater*, even though the result of the former is actually to negate the latter.

The convolutional neural network can take word context into account, but unfortunately the more complex parameters are not easy to inspect manually. When inspecting the misclassified sentences manually, however, it turns out to make some of the same mistakes as the other methods, for example, classifying *more bankruptcies* as positive. For many other sentences it is less clear why they are misclassified, although they do seem to have a large amount of rare or new words such as *Werkgevers torpederen caos* (employers torpedo collective-bargaining-agreements), *dubbelfout bij crisistaks* (double-error with crisis-tax) and *Grieken zijn weer platzak* (Greeks are stone-broke again). Then again, all these words were in the word embeddings and most had sensible synonyms, for example, listing *werkgeversheffing* (employer charge) and *graaitax* (grabber tax) as synonyms for *crisistaks*, although *caos* was misclassified (presumably due to the failed lemmatization to the singular form *CAO*) and *dubbelfout* (double error) was related mostly to tennis terms. Still, since the close synonyms of these rare words are mostly rare words themselves, it is possible that even though the embeddings vectors does words not in the training data to be used for classification, if the words occur in too few contexts in the documents used for creating the embeddings they will still cause difficulties for the algorithm.

## Discussion

Determining the tone or valence of statements is an important task for analyzing communication. Sentiment analysis is not an easy task, however. Sentiment is conceptually not trivial: statements such as “Libya’s Moammar Gadhafi killed”, ‘Brexit was postponed (again)’, or “House prices skyrocketing” can be positive or negative (or even neutral), depending on your perspective and definition of sentiment. Subjective language is also typically more ambiguous and more creative than factual statements, and even trained expert coders can have serious difficulty agreeing on the sentiment of statements.

This paper investigated the relatively straightforward (but substantively important) case of differentiating good from bad economic news headlines. We compared a large number of different methods for measuring this sentiment: trained student coders; crowd coding; classical machine learning and deep learning; and a large variety of dictionaries. For the latter, we used both original (generic) Dutch dictionaries and generic and domain-specific English dictionaries applied to automatic translation of the text. The results of all these methods were compared to a gold standard created by coding every unit multiple times and resolving any disagreements.<sup>8</sup>

The main finding is that human coding still carries the day for sentiment coding, with only trained students and crowd coding achieving levels of agreement with the gold standard that would generally be accepted as valid measurements. Of these, student coding is clearly still better than crowd coding at replicating the gold standard coding, achieving a Krippendorff’s alpha of over 0.9 when using three coders per text. Crowd coding, however, also has relatively good accuracy, achieving an alpha of over 0.8 when each text is coded multiple times, equal to the performance of single student coding. With this performance, crowd coding has a number of substantial advantages over regular student coding. First, crowd coding will often be significantly cheaper



than student coding. For our experiments, we paid coders 0.02 USD per sentence. Including test questions, total costs were less than 50 USD for 1500 annotations. This makes it affordable to code all units multiple times, a practice often considered too expensive for manual coding, which not only improves the overall point estimate of sentiment, but also gives a measure of spread (cf. Benoit et al., 2016; Lind et al., 2017). A second advantage is that after setting up the job, almost no researcher effort is required, as the system takes care of recruiting, testing, and monitoring the coders.

Please note that this does not mean we endorse the economics or business models of current crowd coding providers. In our view, the main benefit of crowd coding is that the process is inherently transparent. Since coders are interchangeable and all training and selection happens within the system, we can be sure of the exact training and material that the coders received. With expert (or student) coding, the presumed standard practice is to have training events where groups of coders are instructed in the codebook and the coding routines. Although ideally the training procedure would be published together with the codebook, it is difficult to avoid coders talking with each other and with the instructors, meaning that it is possible that a shared understanding can arise that is not captured in the codebook. This potentially makes it hard to create truly replicable coding and implies that the published intercoder agreement can be overly optimistic. For crowd coding, in contrast, there is no reason to assume why the outcome will be structurally different if another researcher launches the same job, guaranteeing replicability and validity of the reliability measure. This is in line with the findings and recommendations of Weber et al. (2018), who advocate crowd coding after having difficulty replicating their own previous codings of moral claims.

Dictionary coding is probably the most used automatic sentiment analysis method in the social sciences due to its transparency and simplicity. Unfortunately, but not surprisingly given previous results (Boukes et al., 2020; González-Bailón & Paltoglou, 2015; Soroka et al., 2015), performance of dictionaries was not satisfactory. Agreement was generally close to chance agreement, and correlations between dictionaries were also low. Error analysis showed that this was mostly due to the missing context of words. This holds for both Dutch dictionaries and for English dictionaries applied to the automatically translated texts.

In line with findings from the computational linguistics community, machine learning significantly outperformed dictionary coding, with a “deep learning” convolutional neural network scoring around 20 percentage points higher on both agreement and intercoder reliability measures (Liu, 2012; Rosenthal et al., 2017; Rudkowsky et al., 2018). This performance is notwithstanding the fact that a relatively low amount of training material was available, implying that better results might still be achievable with more training data. It is also possible that the machine learning models had particular difficulties with the low number of words in each headline, which could have exacerbated this data scarcity problem. This could also explain the higher performance of the CNN model. Because this model used word embeddings vectors rather than the word counts (as in the NB and SVM models), the model could utilize the words in the gold standard headlines even if these words did not occur in the training data. The disadvantage of machine learning methods are, however, their relative lack of transparency and the need for coded training data. The fact that a model is based on specific training data also means that a different task or domain requires new training data, so for smaller tasks it might be better to just use manual coding on a sufficiently large sample.

## Recommendations for text classification

There are many different techniques for sentiment analysis and classifications of texts more generally. Unfortunately, the results of this article show that automatic techniques do not always perform sufficiently. This even holds for dictionaries that have been developed and validated independently: Since tasks and domains are almost never interchangeable in social science, validated

performance on the task a tool was developed for does not guarantee sufficient performance on a new task.

On the basis of these findings, we recommend that text classification projects should follow the following steps to guarantee validity:

- (1) Formalize the conceptualization and operationalization for manual annotation of the quantity of interest. This step is extremely important and often requires pilot coding of material and discussion between researchers.
- (2) Annotate a sufficiently large gold standard for validation. This needs to be coded by at least two annotators to calculate inter-coder reliability (or three in the case of disagreements). The size of the sample depends on the number and distribution of categories, but a good indication is Table 11.2 in Krippendorff (2012, p. 240) which shows that often between 100 and 300 units are required. If insufficient reliability is achieved, go back to step one and repeat; otherwise, finish the validation set by discussing and resolving all disagreements.
- (3) Apply any applicable off-the-shelf dictionaries. If any of these is sufficiently valid as determined by comparison with the gold standard, we recommend using this for the text analysis as dictionaries give very good transparency and replicability for a low cost.
- (4) If no sufficiently valid off-the-shelf dictionaries exist, consider customizing a dictionary or creating one. In this case, it is paramount that the gold standard is not referenced when creating a dictionary as that would bias the validity estimate: Any person involved in annotating the gold standard cannot contribute to creating the dictionary. It can be very beneficial to use corpus statistics of the texts under study, for example, by listing the most frequent or surprising words in the corpus or by listing words that are similar to words in an existing dictionary (Amsler, 2020). For example, similar to the results of the error analysis presented above this would have shown that a word like *beurs* (stock exchange) is relatively frequent in this corpus and occurs in the NRC sentiment dictionary, but in the corpus of this study has no clear positive or negative valence and should probably be dropped from the dictionary. As always, make sure the corpus analysis excludes any documents in the gold standard to avoid biasing the performance estimate. As above, test the validity of the created dictionary against the gold standard, and use it if it is sufficiently valid. If many variations need to be tested or threshold scores need to be determined, a second set of documents should be annotated for this. Only the final dictionary should be tested against the validation set to ensure an unbiased estimate of validity.
- (5) If off-the-shelf or custom dictionaries do not achieve sufficient validity, the remaining options are human coding or machine learning. For this, code a relatively large set of articles (a thousand or more) using crowd coding or expert coding. We would recommend to use the validation set created in step 2 to continually test the validity of the manual coding by including a small percentage of validation sentences in the jobs.
- (6) Train a machine learning model using the coded documents. Use *cross-validation* to perform a first validation and/or test multiple models and select (hyper)parameters as needed. If the model is sufficiently valid, train again using the whole coded data set and validate against the gold standard.
- (7) If needed, repeat from step 5 until the model is sufficiently valid or enough units have been coded to perform the substantive analysis.

Following these guidelines ensures that the results are valid and replicable with the minimum amount of manual coding needed, which can range from only coding the gold standard to having to do a fully manual analysis.

We would like to close with a final recommendation for the field. In our view, the biggest problem facing the analysis of sentiment or tone in communication science is the lack of a shared conceptualization. “Sentiment” can mean many things in different theoretical contexts, as is clear

from the fact that, for instance, a hotel review, a political policy preference, and a statement about the economy can all be seen as expressions of sentiment. To remedy this situation, we should categorize the different theoretical claims related to tone or sentiment, and formalize a set of definitions that can be used to create gold standards for each type of sentiment measure. Specifically, attention should be paid to the unit of measurement and to disentangle the value of the sentiment from its source (who is expressing the sentiment) and target (who or what is being evaluated). As a field, we could then collaborate on creating training material and building, comparing, and improving shared tools for one or more concrete sentiment analysis tasks as defined in the first step. This approach of creating *shared tasks* has yielded very good results in computational linguistics and related fields and have the potential to dramatically increase the quality, comparability, and transparency of automatic sentiment analysis in the social sciences.

## Notes

1. <https://github.com/vanatteveldt/ecosent>
2. <http://figure-eight.com>, formerly CrowdFlower
3. Accessed at <https://translate.google.com/and> deepl.com, respectively. Since DeepL provided slightly better performance, we only report these scores here, but see the online compendium for a full overview.
4. <http://www.wjh.harvard.edu/~inquirer/homecat.htm>
5. Since the models were trained on the headlines coded by the student coders as reported by Boukes et al. (2020), it is possible that these models would perform better when validated against the student codings rather than against our gold standard data, even though we used the same coding procedure. To make sure the ML models were not disadvantaged, we also validated the models against the student codings of the gold standard data, which yielded almost identical results. See the online appendix for these outcomes.
6. See the online appendix for the grid search procedure and results
7. See the online appendix for the full analysis and results
8. See the online compendium at <https://github.com/vanatteveldt/ecosent> for all code and materials used in this article.

## Disclosure Statement

No potential conflict of interest was reported by the authors.

## Funding

This work was supported by the Nederlandse Organisatie voor Wetenschappelijk Onderzoek [VI.Veni.191R.006]. The manually annotated data used in this article was collected with support by the Netherlands Organisation for Scientific Research (NWO) with a VIDI grant under project number: 016.145.369.

## ORCID

Wouter van Atteveldt  <http://orcid.org/0000-0003-1237-538X>

Mariken A. C. G. van der Velden  <http://orcid.org/0000-0003-0227-9183>

Mark Boukes  <http://orcid.org/0000-0002-3377-6281>

## References

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., & Zheng, X. (2016). Tensorflow: A system for large-scale machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)* (pp. 265–283). USENIX association. <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>
- Aday, S. (2010). Chasing the bad news: An analysis of 2005 iraq and afghanistan war coverage on nbc and fox news channel. *Journal of Communication*, 60(1), 144–164. <https://doi.org/10.1111/j.1460-2466.2009.01472.x>
- Amsler, M. (2020). *Using lexical-semantic concepts for fine-grained classification in the embedding space* [Unpublished doctoral dissertation]. University of Zurich.

- Benoit, K., Conway, D., Lauderdale, B. E., Laver, M., & Mikhaylov, S. (2016). Crowd-sourced text analysis: Reproducible and agile production of political data. *American Political Science Review*, 110(2), 278–295. <https://doi.org/10.1017/S0003055416000058>
- Benoit, K., Watanabe, K., Wang, H., Nulty, P., Obeng, A., Müller, S., & Matsuo, A. (2018). quantda: 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>
- Blood, D. J., & Phillips, P. C. (1995). Recession headline news, consumer sentiment, the state of the economy and presidential popularity: A time series analysis 1989–1993. *International Journal of Public Opinion Research*, 7(1), 2–22. <https://doi.org/10.1093/ijpor/7.1.2>
- Boukes, M., Van de Velde, B., Araujo, T., & Vliegthart, R. (2020). What's the tone? easy doesn't do it: Analyzing performance and agreement between off-the-shelf sentiment analysis tools. *Communication Methods & 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>
- Bradley, M. M., & Lang, P. J. (1999). *Affective norms for english words (ANEW): Stimuli, instruction manual and affective ratings*. (Technical report C-1). Gainesville, FL: The Center for Research in Psychophysiology, University of Florida.
- Cho, J. (2013). Campaign tone, political affect, and communicative engagement. *Journal of Communication*, 63(6), 1130–1152. <https://doi.org/10.1111/jcom.12064>
- Connaughton, S. L., & Jarvis, S. E. (2004). Invitations for partisan identification: Attempts to court latino voters through televised latino-oriented political advertisements, 1984–2000. *Journal of Communication*, 54(1), 38–54. <https://doi.org/10.1111/j.1460-2466.2004.tb02612.x>
- Damstra, A., & Boukes, M. (2018). The economy, the news, and the public: A longitudinal study of the impact of economic news on economic evaluations and expectations. *Communication Research*, 48(1), 26–50. <https://doi.org/10.1177/0093650217750971>
- De Vries, E., Schoonvelde, M., & Schumacher, G. (2018). No longer lost in translation: Evidence that google translate works for comparative bag-of-words text applications. *Political Analysis*, 26(4), 417–430. <https://doi.org/10.1017/pan.2018.26>
- Domke, D., Watts, M. D., Shah, D. V., & Fan, D. P. (1999). The politics of conservative elites and the “liberal media” argument. *Journal of Communication*, 49(4), 35–58. <https://doi.org/10.1111/j.1460-2466.1999.tb02816.x>
- Dunaway, J. L., Davis, N. T., Padgett, J., & Scholl, R. M. (2015). Objectivity and information bias in campaign news. *Journal of Communication*, 65(5), 770–792. <https://doi.org/10.1111/jcom.12172>
- Elenbaas, M., & De Vreese, C. H. (2008). The effects of strategic news on political cynicism and vote choice among young voters. *Journal of Communication*, 58(3), 550–567. <https://doi.org/10.1111/j.1460-2466.2008.00399.x>
- Goldberg, Y. (2017). Neural network methods for natural language processing. *Synthesis Lectures on Human Language Technologies*, 10(1), 1–309. <https://doi.org/10.2200/S00762ED1V01Y201703HLT037>
- 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., & 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>
- Haselmayer, M. (2019). Negative campaigning and its consequences: A review and a look ahead. *French Politics*, 17, 355–372. <https://doi.org/10.1057/s41253-019-00084-8>
- Haselmayer, M., & Jenny, M. (2017). Sentiment analysis of political communication: Combining a dictionary approach with crowdcoding. *Quality & Quantity*, 51(6), 2623–2646. <https://doi.org/10.1007/s1135-016-0412-4>
- Hilbert, M., Barnett, G., Blumenstock, J., Contractor, N., Diesner, J., Frey, S., & others. (2019). Computational communication science| Computational communication science: A methodological catalyzer for a maturing discipline. *International Journal of Communication*, 13, 3912–3934. <https://ijoc.org/index.php/ijoc/article/view/10675>
- Hopkins, D. J., & King, G. (2010). A method of automated nonparametric content analysis for social science. *American Journal of Political Science*, 54(1), 229–247. <https://doi.org/10.1111/j.1540-5907.2009.00428.x>
- Hopmann, D. N., de Vreese, C. H., & Albæk, E. (2011). Incumbency bonus in election news coverage explained: The logics of political power and the media market. *Journal of Communication*, 61(2), 264–282. <https://doi.org/10.1111/j.1460-2466.2011.01540.x>
- Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. *Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 168–177). Association for Computing Machinery. <https://doi.org/10.1145/1014052.1014073>
- Jonkman, J. G., Boukes, M., Vliegthart, R., & Verhoeven, P. (2020). Buffering negative news: Individual-level effects of company visibility, tone, and pre-existing attitudes on corporate reputation. *Mass Communication and Society*, 23(2), 272–296. <https://doi.org/10.1080/15205436.2019.1694155>

- Kim, H. S. (2015). Attracting views and going viral: How message features and news-sharing channels affect health news diffusion. *Journal of Communication*, 65(3), 512–534. <https://doi.org/10.1111/jcom.12160>
- Kleinnijenhuis, J. (2008). Negativity. In W. Donsbach (ed.), *The International Encyclopedia of Communication* (pp. 1–5). John Wiley & Sons, Ltd. <https://doi.org/10.1002/9781405186407.wbiecn005>
- Kleinnijenhuis, J., Van Hoof, A. M., Oegema, D., & De Ridder, J. A. (2007). A test of rivaling approaches to explain news effects: News on issue positions of parties, real-world developments, support and criticism, and success and failure. *Journal of Communication*, 57(2), 366–384. <https://doi.org/10.1111/j.1460-2466.2007.00347.x>
- Kleinnijenhuis, J., Van Hoof, A. M., & Van Atteveldt, W. (2019). The combined effects of mass media and social media on political perceptions and preferences. *Journal of Communication*, 69(6), 650–673. <https://doi.org/10.1093/joc/jqz038>
- Kosmidis, S., Hobolt, S. B., Molloy, E., & Whitefield, S. (2019). Party Competition and Emotive Rhetoric. *Comparative Political Studies*, 52(6), 811–837. <https://doi.org/10.1177/0010414018797942>
- Krippendorff, K. (2012). *Content analysis: An introduction to its methodology*. Sage.
- Kroon, A. C., Fokkens, A., Trilling, D., Loecherbach, F., Moeller, J., Van der Velden, M. A. C. G., & Van Atteveldt, W. (2019). *The Amsterdam word embedding model*. Proceedings of the 69th annual conference of the International Communication Association (ICA), Washington, DC.
- Lengauer, G., Esser, F., & Berganza, R. (2012). Negativity in political news: A review of concepts, operationalizations and key findings. *Journalism: Theory, Practice & Criticism*, 13(2), 179–202. <https://doi.org/10.1177/1464884911427800>
- Lind, F., Gruber, M., & Boomgaarden, H. G. (2017). Content analysis by the crowd: Assessing the usability of crowdsourcing for coding latent constructs. *Communication Methods and Measures*, 11(3), 191–209. <https://doi.org/10.1080/19312458.2017.1317338>
- Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1), 1–167. <https://doi.org/10.2200/S00416ED1V01Y201204HLT016>
- Liu, B., Hu, M., & Cheng, J. (2005). Opinion observer: Analyzing and comparing opinions on the web. *Proceedings of the 14th international conference on world wide web* (pp. 342–351). Association for Computing Machinery. <https://doi.org/10.1145/1060745.1060797>
- Liu, J., Lee, B., McLeod, D. M., & Choung, H. (2019). Effects of frame repetition through cues in the online environment. *Mass Communication and Society*, 22(4), 447–465. <https://doi.org/10.1080/15205436.2018.1560475>
- Loughran, T., & McDonald, B. (2011). When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *The Journal of Finance*, 66(1), 35–65. <https://doi.org/10.1111/j.1540-6261.2010.01625.x>
- Margolin, D. B. (2019). Computational contributions: A symbiotic approach to integrating big, observational data studies into the communication field. *Communication Methods and Measures*, 13(4), 229–247. <https://doi.org/10.1080/19312458.2019.1639144>
- Martindale, C. (1975). *The romantic progression: The psychology of literary history*. Halsted Press.
- Martindale, C. (1990). *The clockwork muse: The predictability of artistic change*. Basic Books.
- Martins, N., Weaver, A. J., Yeshua-Katz, D., Lewis, N. H., Tyree, N. E., & Jensen, J. D. (2013). A content analysis of print news coverage of media violence and aggression research. *Journal of Communication*, 63(6), 1070–1087. <https://doi.org/10.1111/jcom.12052>
- Matthes, J., & Kohring, M. (2008). The content analysis of media frames: Toward improving reliability and validity. *Journal of Communication*, 58(2), 258–279. <https://doi.org/10.1111/j.1460-2466.2008.00384.x>
- McCombes, M., Lopez-Escobar, E., & Llamas, J. P. (2000). Setting the agenda of attributes in the 1996 spanish general election. *Journal of Communication*, 50(2), 77–92. <https://doi.org/10.1111/j.1460-2466.2000.tb02842.x>
- Meijer, -M.-M., & Kleinnijenhuis, J. (2006). Issue news and corporate reputation: Applying the theories of agenda setting and issue ownership in the field of business communication. *Journal of Communication*, 56(3), 543–559. <https://doi.org/10.1111/j.1460-2466.2006.00300.x>
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems* 26 (pp. 3111–3119). Curran Associates, Inc.
- Mohammad, S. M., & Turney, P. D. (2013). Crowdsourcing a Word-Emotion Association Lexicon. *Computational Intelligence*, 29(3), 436–465. <https://doi.org/10.1111/j.1467-8640.2012.00460.x>
- Muddiman, A., McGregor, S. C., & Stroud, N. J. (2019). (Re)Claiming our expertise: Parsing large text corpora with manually validated and organic dictionaries. *Political Communication*, 36(2), 214–226. <https://doi.org/10.1080/10584609.2018.1517843>
- Muddiman, A., & Stroud, N. J. (2017). News values, cognitive biases, and partisan incivility in comment sections. *Journal of Communication*, 67(4), 586–609. <https://doi.org/10.1111/jcom.12312>
- Munger, K., Luca, M., Nagler, J., & Tucker, J. (2020). The (null) effects of clickbait headlines on polarization, trust, and learning. *Public Opinion Quarterly*, 84(1), 49–73. <https://doi.org/10.1093/poq/nfaa008>



- Nagel, F., Maurer, M., & Reinemann, C. (2012). Is there a visual dominance in political communication? how verbal, visual, and vocal communication shape viewers' impressions of political candidates. *Journal of Communication*, 62(5), 833–850. <https://doi.org/10.1111/j.1460-2466.2012.01670.x>
- Nai, A., & Martínez i Coma, F. (2019, June). Losing in the polls, time pressure, and the decision to go negative in referendum campaigns. *Politics and Governance*, 7(2), 278. <https://doi.org/10.17645/pag.v7i2.1940>
- Narayan, S., & Narayan, P. K. (2017). Are oil price news headlines statistically and economically significant for investors? *Journal of Behavioral Finance*, 18(3), 258–270. <https://doi.org/10.1080/15427560.2017.1308942>
- Natarajan, K., & Xiaoming, H. (2003). An asian voice? a comparative study of channel news asia and cnn. *Journal of Communication*, 53(2), 300–314. <https://doi.org/10.1111/j.1460-2466.2003.tb02592.x>
- Ng, Y.-L., & Zhao, X. (2020). The human alarm system for sensational news, online news headlines, and associated generic digital footprints: A uses and gratifications approach. *Communication Research*, 47(2), 251–275. <https://doi.org/10.1177/0093650218793739>
- Nunez-Mir, G. C., Iannone, B. V., III, Pijanowski, B. C., Kong, N., & Fei, S. (2016). Automated content analysis: Addressing the big literature challenge in ecology and evolution. *Methods in Ecology and Evolution*, 7(11), 1262–1272. <https://doi.org/10.1111/2041-210X.1260>
- Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends R in Information Retrieval*, 2(1–2), 1–135. <https://doi.org/10.1561/15000000011>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Rheault, L., Beelen, K., Cochrane, C., & Hirst, G. (2016). Measuring emotion in parliamentary debates with automated textual analysis. *PLoS ONE*, 11(12), e0168843. <https://github.com/lrheault/emotion>
- Rhodes, J. H., & Vayo, A. B. (2019, January). The historical presidency: Fear and loathing in presidential candidate rhetoric, 1952–2016. *Presidential Studies Quarterly*, 49(4), 909–931. <https://doi.org/https://doi.org/10.1111/psq.12512>
- Richardson, K., Berant, J., & Kuhn, J. (2018). Polyglot semantic parsing in APIs. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), Human Language Technologies*, Volume 1 (Long Papers) (pp. 720–730). New Orleans, Louisiana: Association for Computational Linguistics. <https://doi.org/10.18653/v1/N18-1066>
- Ridout, T. N., & Searles, K. (2011, June). It's my campaign i'll cry if i want to: How and when campaigns use emotional appeals. *Political Psychology*, 32(3), 439–458. <https://doi.org/10.1111/j.1467-9221.2010.00819.x>
- Rodgers, S., & Thorson, E. (2003). A socialization perspective on male and female reporting. *Journal of Communication*, 53(4), 658–675. <https://doi.org/10.1111/j.1460-2466.2003.tb02916.x>
- Rosenthal, S., Farra, N., & Nakov, P. (2017, August). SemEval-2017 task 4: Sentiment analysis in twitter. In S. Bethard, M. Carpuat, M. Apidianaki, S. M. Mohammad, D. Cer, D. Jurgens (eds.) *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)* (pp. 502–518). Association for Computational Linguistics. <https://www.aclweb.org/anthology/S17-2088>
- Rudkowsky, E., Haselmayer, M., Wastian, M., Jenny, M., Emrich, Š., & Sedlmair, M. (2018). More than bags of words: Sentiment analysis with word embeddings. *Communication Methods and Measures*, 12(2–3), 140–157. <https://doi.org/10.1080/19312458.2018.1455817>
- Shah, D. V., Cho, J., Nah, S., Gotlieb, M. R., Hwang, H., Lee, N.-J., and McLeod, D. M. (2007). Campaign ads, online messaging, and participation: Extending the communication mediation model. *Journal of Communication*, 57(4), 676–703. <https://doi.org/10.1111/j.1460-2466.2007.00363.x>
- Shin, J., & Thorson, K. (2017). Partisan selective sharing: The biased diffusion of fact-checking messages on social media. *Journal of Communication*, 67(2), 233–255. <https://doi.org/10.1111/jcom.12284>
- Smedt, T. D., & Daelemans, W. (2012). Pattern for python. *Journal of Machine Learning Research*, 13(66), 2063–2067. <http://jmlr.org/papers/v13/desmedt12a.html>
- Soroka, S., Young, L., & Balmas, M. (2015). Bad news or mad news? sentiment scoring of negativity, fear, and anger in news content. *The ANNALS of the American Academy of Political and Social Science*, 659(1), 108–121. <https://doi.org/10.1177/0002716215569217>
- Tankard, J. W., et al. (2001). The empirical approach to the study of media framing. In Reese, S. D., Gandy, O. H., & Grant, A. E. (Eds.), *Framing Public Life: Perspectives on Media and Our Understanding of the Social World* (pp. 95–106). New York: Routledge. <https://doi.org/10.4324/9781410605689>
- Thelwall, M., Buckley, K., & Paltoglou, G. (2012). Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology*, 63(1), 163–173. <https://doi.org/10.1002/asi.21662>
- Valenzuela, S., Piña, M., & Ramírez, J. (2017). Behavioral effects of framing on social media users: How conflict, economic, human interest, and morality frames drive news sharing. *Journal of Communication*, 67(5), 803–826. <https://doi.org/10.1111/jcom.12325>
- 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>



- Van Atteveldt, W., Welbers, K., & Van der Velden, M. A. C. G. (2019). Studying political decision-making with automatic text analysis. *Oxford Research Encyclopedia of Politics*. Oxford University Press. <https://doi.org/10.1093/acrefore/9780190228637.013.957>
- Van den Bosch, A., Busser, G., Daelemans, W., & Canisius, S. (2007). An efficient memory-based morphosyntactic tagger and parser for Dutch. In F. van Eynde, P. Dirix, I. Schuurman, & V. Vandeghinste (Eds.), *Selected papers of the 17th computational linguistics in the Netherlands meeting, Leuven, Belgium* (pp. 99–114).
- Vargo, C. J., Guo, L., McCombs, M., & Shaw, D. L. (2014). Network issue agendas on twitter during the 2012 us presidential election. *Journal of Communication*, 64(2), 296–316. <https://doi.org/10.1111/jcom.12089>
- Weber, R., Mangus, J. M., Huskey, R., Hopp, F. R., Amir, O., Swanson, R., Gordon, A., Khooshabeh, P., Hahn, L., & Tamborini, R. (2018). Extracting latent moral information from text narratives: Relevance, challenges, and solutions. *Communication Methods and Measures*, 12(2–3), 119–139. <https://doi.org/10.1080/19312458.2018.1447656>
- Welbers, K., Van Atteveldt, W., & Benoit, K. (2017). Text analysis in r. *Communication Methods and Measures*, 11(4), 245–265. <https://doi.org/10.1080/19312458.2017.1387238>
- Wiebe, J., Wilson, T., Bruce, R. F., Bell, M., & Martin, M. (2004). Learning subjective language. *Computational Linguistics*, 30(3), 277–308. <https://doi.org/10.1162/0891201041850885>
- Wilkerson, J., & Casas, A. (2017). Large-scale computerized text analysis in political science: Opportunities and challenges. *Annual Review of Political Science*, 20(1), 529–544. <https://doi.org/10.1146/annurev-polisci-052615-025542>
- Wojcieszak, M., & Azrout, R. (2016). I saw you in the news: Mediated and direct intergroup contact improve outgroup attitudes. *Journal of Communication*, 66(6), 1032–1060. <https://doi.org/10.1111/jcom.12266>
- Young, L., & Soroka, S. (2012a). Affective news: The automated coding of sentiment in political texts. *Political Communication*, 29(2), 205–231. <https://doi.org/10.1080/10584609.2012.671234>
- Young, L., & Soroka, S. (2012b). *Lexicoder sentiment dictionary*. McGill University.