Amsterdam University Press

COMPUTATIONAL COMMUNICATION RESEARCH 5.1 (2023) 85–108
HTTPS://DOI.ORG/10.5117/CCR2023.1.004.KATH

Automated Detection of Voice in News Text – Evaluating Tools for Reported Speech and Speaker Recognition

Ahrabhi Kathirgamalingam
Computational Communication Science Lab, University of Vienna

Fabienne Lind
Computational Communication Science Lab, University of Vienna

Hajo G. Boomgaarden
Computational Communication Science Lab, University of Vienna

Abstract
The automated content analysis of text has become integral to contemporary communication and journalism research. However, automated approaches are seldom utilized to analyze reported voice in text, while doing so would offer valuable insights into media and communication practices. Bridging the fields of communication science and computational linguistics, this study reviews and evaluates off-the-shelf tools for automated voice detection (of direct/indirect speech and of speakers) with respect to user experience and validity. Manually annotated English news articles and Twitter data served as baseline for evaluating the automated detection of voice. Findings indicate that the tools being assessed offer a satisfactory user experience and provide promising solutions for detecting direct speech automatically, encouraging fellow researchers to utilize automated detection for direct quotations. However, the recognition of indirect speech and speakers needs considerable improvement.

Keywords: automated voice detection, reported speech annotation, quote annotation, speaker annotation, quote attribution, automated content analysis

Automated Detection of Voice in News Text

The digitization of both research objects and tools has greatly affected research in communication science and journalism studies and the development of computational methods to meet the emerging possibilities and challenges is increasing (Bennett, 1990; Hepp, 2016; Löblich, 2010). While generally automated approaches to content analysis are nowadays quite

commonly applied (Van Atteveldt and Peng, 2018), the specific method of automated voice analysis is seldom addressed in communication and journalism research. By 'voice' we refer to the direct or indirect speech of actors as reported in text. Thus, voice detection includes the identification and extraction of any reported speech as well as the attribution of speech acts to certain actors or actor categories. Since the practice of incorporating actors (i.e., people, organizations) or other sources such as documents in text and the embedding of the quoted statements is prevalent and highly relevant in journalism and media communication (Berkowitz, 2009), automated voice analysis presents a potentially very important approach for communication and journalism scholars. Detecting actors' voices in text, for example, allows to shed light on journalists' work with sources (e.g., Reich, 2011) or to compare arguments or opinions of various mentioned actors. The empirical examination of reported speech in news content (e.g.,Bennett, 1990) has largely been based on manual content approaches (e.g.,Crawley et al., 2016; Fengler and Kreutler, 2020). Yet, there are also efforts to automate the detection of voice in text (Krestel et al., 2008; Lazaridou et al., 2017; Pouliquen et al., 2007). A considerable part of these tools can be attributed to developers from computational linguistics (e.g., Brunner, 2015; Elson and McKeown, 2010; He et al., 2013; Muzny et al., 2017; O'Keefe et al., 2012), with some important contributions also coming from communication scholars (e.g., Krestel et al., 2008; Pouliquen et al., 2007; Scheible et al., 2016; Welbers et al., 2021). Beyond the developers' reports, there is little to no literature on the performance of these automated voice detection tools and the tools are rarely used by other researchers than the developers themselves. Keeping this in mind and considering that methods critically shape the research questions we raise and the problems we solve (Waldherr, 2019), this study focuses on reviewing potential approaches to automate the detection of reported speech from persons and organizations in textual data. By evaluating off-the-shelf tools, we intend to raise their visibility and ultimately seek to contribute to their wider application.

More specifically, we reviewed and tested three natural language processing tools that offer the immediate recognition of voice in English: CoreNLP (C. Manning et al., 2014), QSample (Scheible et al., 2016), and rsyntax (Welbers et al., 2021). We evaluated the selected tools in respect to user experience and focused here on aspects such as required user skills, output data structure, and post-processing. The validity of the tools is assessed through a comparison of the tools' coding decisions to manual coding decisions. Our manually coded benchmark consists of 200 English news articles and 400

English tweets, yielding 1.851 instances of reported speech with corresponding speakers. This way, we test the tools jointly on a variety of text types. We explain which tool is more suitable for which sub task (direct speech, indirect speech, and speaker detection) and provide recommendations for their usage in automated content analysis tasks.

In what follows, we first discuss the theoretical need for voice detection, before moving on to a reflection on the automation of the detection of voice in text. Subsequently, we review available voice detection tools and discuss their quality assessment.

Examining Voices

Since methodological choices impact the validation of existing theories and the construction of new theories (Hepp, 2016; Mahrt, 2015; Strippel et al., 2018), the usability and particularly the validity of new methodological approaches cannot be examined isolated from theory. Several theoretical approaches reflecting on voice in media provide grounds for empirical investigations. To name a few, theories query on journalistic rationales of including voices, e.g., to cross-check (Reich, 2011), to increase credibility (Gans, 1979), to construct a certain objectivity (Tuchman, 1972), or on the aspects determining who is included and who is not such as newsworthiness (Reich, 2011), proximity to media production and elitism (Hall et al., 1978), or socio-demographics such as gender and ethnicity (Bennett, 1990; Benson, 2009; Hall et al., 1978). Other theories consider the promises and pitfalls of being included as a speaker (e.g., Berkowitz, 2009; P. Manning, 2001). To empirically address these theories conducting an analysis of voices is useful. Voice analysis focuses on detecting and analyzing direct and indirect speech from persons, organizations, or other sources such as documents incorporated in textual data. Thereby, the analysis allows insights into presence, absence and distributions of speakers, and on the content level of the reported speech itself.

Considering existing voice analyses by means of manual analysis (e.g., Crawley et al., 2016; Fengler and Kreutler, 2020; Thorbjørnsrud and Figenschou, 2016) a few observations are noteworthy. First, scholars detect both direct and/ or indirect quotations. Second, only a few studies explain the coding rules in more detail (e.g., Carpenter, 2008; Thorbjørnsrud and Figenschou, 2016). Third, for both kinds of speech, verbs of speaking - so-called *verba dicendi* (e.g., say) play a significant role. Fourth, in case of direct speech, quotation marks are crucial (Thorbjørnsrud and Figenschou, 2016). Fifth, the person or organization that is the source of the voice is coded

and some studies coded other characteristics of a speaker such as socio-demographics(e.g., Crawley et al., 2016). Sixth, many studies combined voice analysis with other content analyses, such as framing (e.g., Brown et al., 2017; Thorbjørnsrud and Figenschou, 2016).

The current state of manual investigation techniques in communication science presents challenges such as high costs, imprecision, and poor inter-coder reliability. These challenges, along with the richness of theoretical frameworks that deal with voices, justify the need to explore the potential capacities of automated voice analysis.

Automating Voice Detection

Efforts to apply new digital opportunities to voice detection have been initiated since the 2000s (e.g., Pouliquen et al., 2007). The main body of literature can be situated in computational linguistics, where voice analysis enables, for example, the investigation of literary texts (Brunner, 2015; Elson and McKeown, 2010; Muzny et al., 2017). Advancing interdisciplinary research, computational linguistics approaches are also considered for communication and journalism science purposes. The techniques underlying text-as-data approaches can be grouped in three umbrella categories. Some approaches to automating voice detection rely on rule-based techniques where punctuation, morphological, and syntactic patterns (e.g., Krestel et al., 2008) or dictionaries with the names of speakers and *verba dicendi* (e.g., Pouliquen et al., 2007) are used. Others rely on supervised machine learning models where annotation rules are learned from pre-coded samples (e.g., Scheible et al., 2016).Yet, unsupervised machine learning approaches offer also great potential as seen in related fields (Boumans and Trilling, 2016).

Voice detection, strictly speaking, consists of several different tasks,and these tasks can be assigned to two main categories: reported speech and voice attribution (Muzny et al., 2017). Defining the first category, reported speech detection refers to the annotation of speech. Researchers distinguish between direct and indirect speech (e.g., Pouliquen et al., 2007; Scheible et al., 2016). Some additionally consider mixed speech (e.g., Pareti et al., 2013). Direct speech is fully enclosed in quotation marks and is a verbatim reproduction of the original utterance. Indirect quotations paraphrase the original utterance without quotation marks. Mixed quotations contain both verbatim and paraphrased content (Pareti et al., 2013).*Verba dicendi* are central for all reported speech forms (Lazaridou et al., 2017; Pareti et al., 2013). Further, and complicating voice analysis, there are many variations to both direct and indirect speech (e.g., implicit or explicit quotes). There

are also many instances of phrases that are not reported speech but follow similar rules (e.g., use of quotation marks for titles) and *verba dicendi* that are used in other contexts (Scheible et al., 2016).

Moving on to the second category, voice attribution is the detection of the source of the reported speech (Muzny et al., 2017). Some researchers distinguish between extracting speaker entities and extracting mentions (Muzny et al., 2017). While a speaker entity is the concrete source of the quote, the mention is the attribution that is given closest to the quote (e.g., pronoun or reference) and later helps to identify the speaker entity. Here, too, there are different approaches to detecting the attribution. While computational linguists can often work with a predefined list of the possible actors in their text (e.g., He et al., 2013), this is rather challenging for communication and journalism scholars dealing with news. Another possibility is to incorporate databases with possible actors such as Wiki Data, but these might miss actors that are less or not prominent.

Tools for Voice Detection

Serving the two main tasks described above, a number of tools have evolved over the past two decades. We now review tools that were identified via a comprehensive online search, performed in 2021. As part of this search, we scanned open source platforms such as *GitHub* as well as studies and reports by the tool developers. All reviewed tools process English language data, but differ with respect to the specific voice detection task(s) they perform and regarding their methodological approach to voice detection.

Pouliquen et al. (2007) were among the early developer teams to contribute a voice detection solution for the communication science research community. They describe a rule-based tool that retrieves direct speech and speakers from a large set of news reports. The tool was developed for eleven languages within the framework of NewsExplorer. The logic behind their approach is to look for quotation markers in combination with *verba dicendi* and person names from a list of possible names (Pouliquen et al., 2007). Accordingly, only quotes that meet these three conditions would be found.

The Reported Speech Tagger (RST) by Krestel et al. (2008) is built for the GATE framework. GATE (General Architecture for Text Engineering) is a continuously complemented Open-Source toolkit providing text analysis and language processing solutions. RST detects direct and indirect speech and returns the attributed speaker with a set of six patterns with varying position of verba dicendi, speaker and reporting clause (Krestel et al., 2008).

In the realm of supervised machine learning, Elson and McKeown (2010) propose to treat speaker detection as a classification task. The approach first identifies potential speakers in a text and assigns to each reported speech the most likely speaker, unlike relying on the closest speaker to a quote span. This method was designed for literary texts and therefore uses a gold standard from this domain.

Based on the work of O'Keefe et al. (2012) who consider quote attribution as a sequence labeling task and who experiment with different classifiers and configurations, Pareti et al. (2013) introduce a model that is later extended in Pareti (2015). This pipeline detects cues (*verba dicendi*) with the help of a token-level k-NN classifier and then uses the cue in combination with linear-chain conditional random field (CRF) to locate the quote. This pipeline detects direct and indirect quotations.

Scheible et al. (2016) identified that the correct identification of the beginning and end of a quote is challenging for some existing tools and suggests improvements proceeding from the CRF setting from Pareti et al. (2013). Scheible et al. (2016) propose a semi-Markov sequence model that smooths the Markov assumption and therefore incorporates global features into the classification task. Other than only the presence of *verba dicendi*, Scheible et al. (2016) find the frequency of *verba dicendi* in each reported speech of importance. The semi-Markov model outperforms Pareti (2015). The resulting supervised machine learning based tool, called QSample, is able to detect direct and indirect speech for English data.

Stanford CoreNLP, a well-known pipeline framework for various NLP tasks such as tokenization, part-of-speech tagging and named entity recognition, also offers a tool for direct speech and speaker detection called Quote Extraction and Annotation (C. Manning et al., 2014).The annotator was provided by Muzny et al. (2017) who combined the rule-based approach to direct speech annotation from O'Keefe et al. (2012) with a two-stage sieve approach to detect not only the attributed mentions but also the speaker entities. The two stages of the algorithm consists of first linking the quote to the mention and then linking the mention to the speaker. For both steps different deterministic techniques such as rule-based patterns, dependency parsing, co-referencing, and others are used (Muzny et al., 2017). Due to missing records, it is rather challenging to state which techniques from O'Keefe et al. (2012) and Muzny et al. (2017) are included in the currently available CoreNLP pipeline.

Another approach to voice detection is an R package that allows querying syntactic dependency trees called rsyntax. This package was first presented

as part of a case study (Van Atteveldt et al., 2017) and later updated with more information on its solutions for voice detection (Welbers et al., 2021). Rsyntax provides functions for querying direct and indirect speech and also speakers from tokenized data. The querying relies on information on the token's relation to other tokens in a sentence due to a prerequisite step of dependency parsing. Since this package can be combined with different language models it can be potentially utilized for languages other than English.

Quality Assessment of Tools

Assessing the quality of automated voice detection tools, we concentrate on user experience and validity as primary criteria.

User experience focuses on the users' interaction with a service and describes their overall impression (Hassenzahl and Tractinsky, 2006). Tools need to be user-oriented and feedback from external users is significant for their evaluation and further development. Different suggestions exist on which aspects should be examined to assess user experience (Laugwitz et al., 2006). Here, user experience implies aspects that may be decisive for the use of automated tools for voice detection from communication and journalism scholars' perspective: accessibility, required technical infrastructure, required skills and expertise, required data preparations, needed post-processing of the outcomes, and overall strengths and weaknesses of the tools. Up to this point, there are no reports on the user experience of the tools reviewed for this study.

The second and crucial criterion for quality assessment is validity. Validity is ensured when the tool measures what it is supposed to measure (Krippendorff, 2004) and is described as a significant challenge for computational methods. Testing validity is necessary for developing a tool (Grimmer and Stewart, 2013) and measuring validity by comparing the computational output with a gold standard is a standard procedure. The developers of the tools mentioned above rely on and report parameters such as precision, recall, and F1 scores (Muzny et al., 2017; Pareti et al., 2013; Scheible et al., 2016). A direct comparison of these reported parameters is quite misleading, as the nature of the evaluation for each tool differs, especially as they are validated based on different data sources. Thus, to jointly assess the validity of the presented voice detection tools we propose the implementation of an identical evaluation protocol for all tools.

A quality assessment by external researchers comparing several tools with each other is not yet available for voice detection tools. While voices are

very relevant from a communication and journalism science perspective, and tools to automatically detect them are also continuously being developed, the application of these tools is often rather rare and mostly by the developers of the tools. The aim of this study is, thus, to provide guidance and a systematic comparison of different tools with the aim of leading to more frequent and better use of the tools.

Data & Methods

Data Selection

Since the tool review presented here primarily addresses communication science and journalism researchers, we selected a variety of news texts and tweets for the tool performance test. In line with the limited language capabilities of the tools, we chose English language news texts. The selected data contain news texts from two different media types: newspapers and social media. Analyzing voice in newspapers is a major application field in communication science and journalism studies. Newspaper content follows systematic punctuation rules and provides a certain text quality, which increases the chances of high quality computational annotation. To review the tool's performance on newspaper data with variation in reporting style, we utilize 100 newspaper articles from traditional and tabloid papers from UK and USA: *The Guardian*, *Daily Mirror*, *Washington Post*, and *USA Today*. Additionally, we have a more specialized dataset of newspaper articles on migration from the research project 'Role of European Mobility and its Impacts in Narratives, Debates and EU Reforms' (REMINDER) (Lind et al., 2020). The focus on migration texts is motivated by previous manual analyses that examine the representation of migrants' voices in news coverage (e.g., Brown et al., 2017; Thorbjørnsrud & Figenschou, 2016). The specialized dataset consists of 100 news articles from the tabloid outlet *Daily Mirror* and the traditional outlet *The Guardian* both from the UK. The second selected media type is social media communication, which is another major field of observation, but rarely examined in terms of reported speech. Thus, by manually coding and automatically annotating, we may infer the value of voice detection for shorter text with potentially different reporting style and syntactic structures. We make use of a Twitter dataset that consists of 50 randomly sampled tweets from each of the mentioned outlets (*Daily Mirror*, *The Guardian*, *Washington Post*, *USA Today*) and added 100 tweets from each, the Twitter account of UK right-wing alternative news blog *Guido Fawkes* (order-order.com) and more left-wing alternative news blog *AlterNet*

(alternet.org). These randomly sampled 400 tweets were posted between 2016 and 2018.

Tool Selection

Among the previously discussed tools (see page 6), we have selected all those that were off-the shelf and available. Thus, no longer available tools were discarded from the selection (e.g., Elson and McKeown, 2010; Krestel et al., 2008; Pareti, 2015; Pouliquen et al., 2007). Based on this requirement, the quotation detection tool of the CoreNLP pipeline (C. Manning et al., 2014), QSample (Scheible et al., 2016), and rsyntax (Welbers et al., 2021) were selected to evaluate user experience and validity. All three tools are available via *GitHub*. The pipeline CoreNLP includes a rule-based technology to detect direct speech and speaker, QSample relies on supervised machine learning to detect direct and indirect speech, and rsyntax queries dependency trees to extract both types of reported speech and the speakers.

Methodology

We evaluate the automated voice detection tools with a three-step process: First, we automatically annotate the text sample with the tools and provide a comprehensive documentation of the user experiences. Second, we annotate the same text sample manually to build a reference corpus. Third, the automatically and manually detected information is compared. In the following, each step is described in detail.

Automated Annotation

In the first step, the sampled data were automatically annotated with the selected tools, CoreNLP, QSample, and rsyntax. Meanwhile, to evaluate the user experience, a systematic documentation of the implementation was conducted by following a predefined set of categories (see Online Appendix A[1]). The categories cover the comprehensive process of automated annotation in terms of user experience: installation process, required technical infrastructure, required input data structure, required skills from user, other specific preparations, output data structure, and required post-processing.

Manual Annotation

Three trained human coders annotated direct and indirect speech and the speaker of the reported speech for each text of the presented sample based on a set of coding rules (see Online Appendix B). The coding rules resemble the rules of the tools evaluated here to ensure comparability. Any

[1]https://osf.io/sq5hj/

verbatim quote containing quotation marks is coded as direct speech. Mixed quotations that combine direct and indirect speech within one sentence are coded as direct speech. If a sentence contains more than one word or phrase in quotation marks, each word combination is coded as a separate quote. Multi-paragraph quotations, however, are coded as one comprehensive direct quote. Indirect quotations are coded by identifying *verba dicendi* combined with a reported clause and a speaker. In practice, it appeared to be challenging to distinguish between indirect quotations and interpretative statements by authors. In these cases, the full text as well as context knowledge about the presented story needed to be considered to decide. Additionally, we extracted the speakers of the quotes. The speaker entity can be a person, an organization, or another source that is quoted such as documents. In many cases, however, the speaker of the quote is mentioned by a pronoun or a reference (e.g., the president) in the close surroundings of the quote. Here, the context was considered to identify the correct speaker entity. We coded both, the speaker entity and the mention. If there was no speaker indicated, we coded 'Unknown' as the tools should also be able to recognize that there is no speaker.

To ensure the quality of the manually coded reference corpus, an intercoder reliability test was conducted with all three coders for 10 randomly selected articles and 20 randomly selected tweets. The coders were able to extract up to 47 cases of reported speech from the articles and 8 cases of reported speech from the tweets. In terms of newspaper annotating, we achieved a Krippendorff's alpha = .83 for the detection of direct speech. The identification of indirect speech yielded a Krippendorff's alpha = .74, confirming the mentioned difficulties in detecting indirect speech. For the speaker annotation, the Krippendorff's alpha was = .74. For the annotation of the Twitter data, we yielded a Krippendorff's alpha = 1 for direct speech, alpha = .93 for indirect speech, and alpha = .93 for speaker detection.

The manual annotation resulted in three reference datasets[2] providing information on article level (amount of direct and indirect quotations) and quotation level (type of reported speech, the extracted quote, and speaker). Table 1 summarizes the results of the manual annotation.

In total, manually annotating all three datasets resulted in 1.851 reported speech consisting of 1.198 direct and 653 indirect speech. Further, 1.414 speaker entities and 437 speaker mentions were annotated. Table 1 presents the distribution of the manually annotated quotes between the datasets.

[2]The annotations for the general news, specialized news and Twitter dataset are available here: https://osf.io/f6pt9/

Table 1: Amount of manually coded quotes and speakers per dataset

	Reported S.	Direct S.	Indirect S.	Speaker	Mention
General newspaper	958	558	400	756	202
Specialized newspaper	810	577	233	578	232
General Twitter	83	63	20	80	3
Total	1851	1198	653	1414	437

Note. S = Speech, Reported Speech = Direct and Indirect Speech. General news dataset n = 100 articles, Specialized news dataset n = 100 articles, General Twitter dataset n = 400 tweets.

The high amount of reported speech, especially direct speech, indicates its importance in journalistic reporting and encourages communication and journalism researchers to empirically assess this field. In the sampled tweets, however, reported speech occurs rather less frequently.

Comparison of Automated and Manual Annotation

The automated annotation results were then systematically compared with the manual results to assess the validity of the voice detection tools. A combination of automated and manual cross-checking enabled tracking true positives, false positives, and false negatives. The three categories were used to calculate precision, recall, and F1 scores (Goutte and Gaussier, 2005) for each tool by task and by dataset. For the speaker annotation, we calculated two recall, precision, and F1 scores per task and dataset. Both versions were only calculated for the reported speech that were correctly extracted by the tools. In the first version, automatically detected speaker entities and mentions were both considered correct. The second version only considers speaker entities (and not mentions such as pronouns) as sufficient.

As another method to identify possible shortcomings of the tools, we conducted a classification error analysis by manually checking the false positives and false negatives.

Results

In the following, we present how CoreNLP, QSample, and rsyntax performed with respect to user experience and in contrast to the manual coding decisions.

User Experience

CoreNLP's website delivers well-documented information on the installation of CoreNLP and the required files (C. Manning et al., 2014). Ideally, the input data is organized in plain text files. Although CoreNLP detects most different quotation marks in addition to regular quotation marks, embedded quotations need to be marked by differing punctuation (C. Manning et al., 2014). The documented command initiates other pipeline applications such as tokenizer and sentence splitter, that are required to successfully run the quote extraction and attribution.

As a result of fast processing, CoreNLP creates output files for each input file. The extracted quotes and speakers are enlisted with the assigned index per quote and information on the byte where the quote begins, presented as follows: *Speaker: "Quote" [index = X, charOffsetBegin = Y].*

Similar to CoreNLP, QSample can be downloaded from *GitHub* (Scheible et al., 2016) and provides well-documented installation steps. The implementation requires *Java* (>= 1.7) and *Maven* (>= 3.0.0) to be preinstalled and to run the command. QSample requires the input text files gathered in a specific folder. QSample's command then processes the input data and creates tokenized output files with words and punctuation as units. Each token is assigned to a label: C, B, E, I, or 0. If the token is assigned a zero, it is not part of a quote. Label 'C' identifies a token as a cue for a quote, while 'B' marks the beginning, 'E' the ending, and 'I' the span of the quotation. QSample does not categorize the detected quotes into direct and indirect speech. For most purposes, researchers need to reassemble the relevant tokens in a post-processing step. Even though only minimal coding knowledge was required to run the tool, expertise in coding and data management are required for post-processing.

Rsyntax is available either from the Comprehensive R Archive Network (CRAN) or as a developer version on *GitHub*. The input files are first processed using NLP applications such as *spaCy* or *UDpipe*, which are applied for dependency parsing. For the queries applied to the resulting dependency parsed data, users need to set a list of *verba dicendi* (see Online Appendix C for the list used here). This is an important step, as it decides which reported speech clauses are extracted by rsyntax. Based on the predefined list of *verba dicendi* and the dependencies of the tokens, rsyntax searches for specific patterns that are indicated by queries. This process yields labels per token (e.g., quote and source) that can be used in a post-processing step to extract reported speech and speaker. Using rsyntax with the published codes does not provide any information on the type of reported speech. However, the

queries can be easily customized and reformulated to suit various needs. As Welbers et al. (2021) describe, some basic understanding of R and especially data frames is necessary to utilize the package.

Comparison of Automated Annotations to Manual Baseline and Error Analysis

Regarding validity, Table 2 shows the number of automatically extracted annotations and the validity scores by task and dataset per tool. The results of the classification error analysis are addressed here to provide more insight into the contexts of the validity scores.

CoreNLP was used to tag direct speech and the related speakers and annotated – for all three datasets together – 1401 direct quotes and corresponding speakers. Assessing CoreNLP's annotations for the general newspaper dataset reveals sufficient precision values (.85) and very high recall (.99), leading to a high F1 score (.91). Results were similar for the specialized migration dataset (precision = .79, recall .90, F1 score = .84). For the Twitter data, precision was considerably lower (.60), while recall was still high (.90), which resulted in a passable F1 score (.72). A closer examination of the false positives revealed that most difficulties occurred for phrases written in quotation marks without being a quote (e.g., titles). Inspecting the few false negatives reveals that CoreNLP struggles with quotations when a space is missing before or after the quotation mark.

For the task of speaker detection and with permitting mentions, CoreNLP performed with passable precision (.72), recall (.72) and F1 score (.72) for the general newspaper dataset, while the values were lower for the specialized newspaper dataset with precision (.49), recall (.54) and F1 score (.52). The speaker annotation of the Twitter data, however, yielded a precision of .60, a recall of .54 and, therefore, an F1 score of .57. With restricting true positives to speaker entity recognition and excluding mention detections, CoreNLP performed poorer for all three dataset.[3] Examining the misclassification points to a potential problem of other entities and mentions in the close surroundings or within the reported speech.

QSample annotated 2.221 instances of reported speech (direct and indirect). Describing the results for the detection of reported speech in general newspaper dataset, we observed a precision of .75 and a high recall of .82, leading to a F1 score of .79. For the specialized newspaper dataset we

[3]General newspaper dataset: Precision = .58, recall = .57, F1 score = 57; Specialized newspaper dataset: Precision = .37, recall = .40, F1 score = .38; General Twitter dataset: Precision = .51, recall = .46, F1 score = .48

Table 2: Comparison of Automated Annotations to Manual Baseline

	Reported Speech		Direct Speech			Indirect Speech		Speaker	
Tool	QSample	rsyntax	CoreNLP	QSample	rsyntax	CoreNLP	QSample	CoreNLP	rsyntax
General newspaper dataset - 100 articles									
N	1172	827	651	492	370	680	457	551	661
P	.75	.80	.85	.90	.94	.64	.68	.72	.93
R	.82	.68	.99	.85	.62	.77	.72	.72	.76
F1	.79	.74	.91	.88	.75	.70	.70	.72	.83
Specialized newspaper dataset - 100 articles									
N	1011	747	655	464	380	547	367	634	552
P	.69	.74	.79	.96	.94	.46	.53	.49	.88
R	.80	.66	.90	.77	.64	.81	.72	.54	.60
F1	.74	.70	.84	.85	.76	.59	.60	.52	.72
General Twitter dataset - 400 tweets									
N	38	57	95	21	25	17	32	59	35
P	.66	.61	.60	.71	.95	.59	.44	.60	.89
R	.28	.42	.90	.24	.32	.50	.70	.54	.37
F1	.39	.50	.72	.36	.48	.54	.54	.57	.53

Note. *N* = Amount of detected speech or speaker, P = Precision, R = Recall, F1 = F1 score. Precision, Recall, and F1 score are calculated based on the manually annotated results reported in Table 1. Reported validity scores for speaker detection are calculated based on speaker entities and mentions.

achieved similar results (precision = .69, recall = .80, F1 score = .74), for Twitter, the scores were much lower. Here, the tool performed with a precision of .66, a recall of .28 and F1 score of .39. We further used quotation marks as indicators for direct speech, to see if we can find any differences in the tool's performance for direct and indirect speech annotation. In both newspaper datasets, the F1 scores were considerably higher for direct speech tagging (.88 and .85) than for indirect speech annotation (.70 and .59). In the case of Twitter data, however, the direct speech tagging performed poorer (F1 score= .36) than the indirect speech tagging (F1 score = .54) because precision (.71) and recall (.24) were much more imbalanced for direct speech than for indirect speech. The few false positives for the direct speech detection point to the same problems that were found with CoreNLP regarding quotation marks. A reason for the false negatives was that QSample was not able to detect multi-paragraph quotes. A closer look at the false positives

and false negatives for the indirect speech detection does not provide much insight. QSample splits many annotated quotes into two or three parts for an indistinct reason, which makes it difficult to find clear patterns. Possibly, this might be due to subordinate clauses (e.g., The person, who…) and, therefore, more complex sentence structures.

Moving on to the third tool in review, rsyntax was tested with respect to reported speech (direct and indirect) as well as speaker annotation. Rsyntax performed with high precision for both general (.80) and specialized news content (.74), while recall is lower (.68 and .66), leading to an F1 score above .70 for both. For Twitter, the tool annotated with a precision of .61 and a low recall of .42, resulting in an F1 score of .50. Here too, we distinguished between direct and indirect speech by considering quotation marks as indicators. This way, it was possible to find that precision for direct speech was much higher for all data types (above .94) than for indirect speech, while recall was higher for indirect speech annotation than for direct speech (.72 for both newspaper data and .70 for Twitter), leading to similar F1 scores between direct and indirect annotation across all three datasets. Taking a closer look at the false positives reveals that rsyntax extracted phrases including *verba dicendi* from the predefined list with a different meaning (e.g., Our politicians once told the truth). Also, subordinate clauses within sentences were misclassified as reported speech sometimes (e.g., The person, who did not reveal his surname). In terms of false negatives, rsyntax missed to annotate direct speech quotes that are entirely written in quotation marks without any dependencies to other tokens outside of the quotation marks. In a few cases, false negatives resulted from *verba dicendi* that were missing from the predefined list.

Lastly, rsyntax showed a good performance for speaker detection with an F1 score of .83 for general news and of .72 for specialized news, with precision being higher than recall. For annotating speakers for the tweets, the tool shows high precision (.89) and low recall (.3), resulting in an F1 score of .53. With restricting the true positives to correctly matched speaker entities, not mentions, the validity decreases.[4] Examining the false positives, we find that mostly tokens that introduce a subordinate clause and therefore show a more complex dependency structure are misclassified as speakers (e.g., who). Also, the singular and plural first-person pronouns (I and we) are often incorrectly tagged as speakers, especially in cases where a person reports themselves within a reported speech (e.g., I say we…).

[4]General newspaper dataset: Precision = .60, recall = . 56, F1 score = .58; Specialized newspaper dataset: Precision = .83, recall = .49, F1 score = .62; General Twitter dataset: Precision = .86, recall = .36, F1 score = .51

Discussion

A research field that can benefit a lot from computational methods is the investigation of reported speech in textual data. The existing impulses in automated methods for voice detection have been rarely used in applied research, although they promise relief for the labor-intense challenge of data collection and annotation. This study addressed the usability and validity of automated approaches for voice detection.

Overall, we can state that the user experience of the tools evaluated here was highly satisfactory, while validity differed considerably across tools, but also across tasks and data domains. This leads us to recommend different tools for different tasks and data types. To detect direct speech in newspaper and Twitter data, we recommend CoreNLP. Further, we suggest utilizing rsyntax for reported speech (i.e., direct and indirect speech) and speaker annotation in newspaper data. While QSample is worth considering for the annotation of reported speech in newspaper data, we find that flexibility and potential for improvements speak in favor of rsyntax.

In the following, we discuss user experience and validity in greater detail before heading to a detailed discussion on our overall evaluation and recommendations.

User-Experience

The here reviewed tools, CoreNLP, QSample, and rsyntax, provide a comparable user experience in most of the evaluated points. They provide a rather easy-to-implement download and installation process. The required technical infrastructures are well-documented. It may take some effort to set up all prerequisites. Further, the tools provide straightforward commands that reduce the requirements on coding skills considerably and provide results in only a few minutes.

We observed more significant differences in the structure of the output and the required post-processing for subsequent analyses. While CoreNLP assembles all extracted quotations in a specific order, QSample and rsyntax return the articles in tokenized manner with annotations. For post-processing, experience in coding and data structures come in very handy. As an advantage, rsyntax outputs can be post-processed without switching the environment, which can be an asset if users have some experience with coding in R.

In summary, the tools offer solid user experience and are more comprehensible and customizable for more experienced coders, especially in

regard to post-processing.

Validity

In comparison to the values that Muzny et al. (2017) report for CoreNLP (precision = .89, recall = .75, F1 score = .81), the F1 score of the presented annotation for direct speech does not differ remarkably, while precision and recall do. In our analysis, CoreNLP performs somewhat poorer yet still satisfactory in terms of precision and better in respect to recall. Taking some loss on recall into account, a possible solution for a better precision value that can be derived from the error analysis would be to exclude extracted direct speech with less than two words.

For the CoreNLP speaker detection, Muzny et al. (2017) report a precision of .9, a recall of .65, and an F1 score of .75. When accepting mentions as correct annotations in addition to speaker entities, CoreNLP performs similar to the report in the case of general news and even better in case of Twitter data, while the scores are poorer for the specialized newspaper data on migration. By using the two-stage sieve approach to track from quote to mention and from mention to speaker entity, CoreNLP promises to track the speaker entity correctly even if only a reference to it is present in the direct surrounding of a quote. Our results indicate that this promise is not fully realized.

In the case of QSample, the reported indicators (precision = .79, recall = .71, F1 score = .75) refer to the annotation of direct and indirect speech (Scheible et al., 2016). While the overall F1 score achieved in our automated annotation is similar to the reported outcome, we obtained a lower precision and a higher recall. As we distinguish between direct and indirect speech, a deeper insight is needed here. The scores of the direct speech are substantially better than those of the quotations overall, except for the Twitter data, where the poor recall for direct speech strongly lowers the F1 score. QSample flags only a few passages as direct quotes that are none in the view of our manual coding while also tracking down an acceptable proportion of manually identified quotes in the data set.

For the newspaper data, the validity measures for indirect speech of QSample are poorer than the overall outcome that Scheible et al. (2016) reported. For the Twitter data, however, the validity is better than for direct speech, albeit still not passable. A rather broad definition of indirect speech could lead to the disagreements with the manual baseline. Yet, the results are in line with the literature indicating that indirect speech is rather challenging to detect Pareti et al. (2013)

Welbers et al. (2021) do not report any specific validity values. However, they anticipate that the usage of rsyntax for reported speech provides a good precision, while the improvement of recall is dependent on setting queries that can extract the different shapes of reported speech. Using the queries provided on their *GitHub* repository, our results match these arguments, precision is higher than recall. The balanced F1 score, however, indicates that the tool is acceptable for newspaper data. For the Twitter data, rsyntax is just as unsuitable in terms of validity as QSample. Similar to QSample, rsyntax performs better for direct speech than for indirect speech. While both tools perform similarly for indirect speech, QSample returns better F1 scores for direct speech as recall is higher. The speaker detection of rsyntax yields good F1 scores for the newspaper datasets, especially for the general news data. Thus, the validity of speaker detection for Twitter is considerably poorer.

Since it is possible to fine-tune the rsyntax queries, the validity could be improved. We suggest extending the code so that it can extract reported speech with the entire quote between quotation marks and other more specific formulations of reported speech. Further, as our classification error analysis shows, the *verba dicendi* list offers potential to improve validity by finding a balance between more and less straightforward verbs. Optimizing the classification of the reported speech might also enhance the validity for speaker detection.

Overall Performance and Recommendations

The evaluation of both, user experience and validity, allows us to make more specific recommendations based on each data type and task, and moreover a few general ones.

In terms of data types, we evaluated CoreNLP, QSample, and rsyntax on newspaper articles and on tweets from news media. The evaluation of all three tools revealed that reported speech detection performs considerably better with text that follows a more sophisticated writing style and is lengthier such as newspaper. For shorter text data such as tweets and other social media posts, we limit our recommendation to direct speech detection with CoreNLP.

In terms of tasks, our results indicate that detecting direct speech works much better than detecting indirect speech. The decision to extract only direct speech is also justifiable as many manual voice analysis studies also limit to direct speech and speaker. For the detection of direct speech, we recommend using CoreNLP for both newspaper data and tweets. The imple-

mentation requires only little coding and data management knowledge, and all outputs can be further processed easily. In terms of validity, CoreNLP provides high values for direct speech annotations, while the speaker recognition performs only on a sufficient level for general newspaper data with permitting mentions (e.g., pronouns or references). In addition, other aspects highlight the benefits of CoreNLP: The possibility to combine voice detection with other NLP tasks and the potential to solve emerging issues by querying community platforms such as Stack Overflow as CoreNLP is widely used. The chance for a long-lasting availability and further development of CoreNLP is high.[5]

For the detection of direct and indirect speech as well as of speakers in newspaper data, rsyntax appears to be a good fit. Our recommendation here is especially based on the flexibility of the tool that allows optimizing the recall by adding further query rules and by fine-tuning the list of *verba dicendi*. Accordingly, the validity scores reported here can be understood more as an initial point. Welbers et al. (2021) also offer a *GitHub* repository[6] that is updated regularly. Lastly, we assume R to be a widely utilized coding language in communication sciences, which makes the tool even more approachable.

If there is no research interest in the speakers, QSample might be worth considering. While the post-processing is somewhat more difficult compared to the other tools, QSample is well usable, too. In terms of validity, it performs slightly better than rsyntax overall. However, the possibility to improve these values is not as easy as with rsyntax, which is designed more transparently.

By directly comparing three off-the-shelf tools considering user experience and validity issues, it can be said that the tools are not yet flawless and need further development. Nevertheless, the reviewed tools offer solutions that prove to be very suitable, at least partially. The performance for direct speech recognition is very satisfactory and highly recommended for research in this area, as the tools score in user experience and validity. For indirect speech and speaker detection, however, there is potential to improve. Nevertheless, the utilization of automated approaches for the detection of direct speech itself is already a major addition to scientific practice, and the

[5]We further recommend monitoring the functionalities of stanza, which is just as Core NLP, another open source general NLP library by the Stanford NLP group. https://stanfordnlp.github.io/stanza/index.html While it includes no specific operation for quote detection yet, the NLP package has received a lot of attention by the community in recent year and is likely to do so in the next years as well.

[6]rsyntax recipe repository: https://github.com/kasperwelbers/rsyntaxRecipes

outcomes of the here presented review may encourage fellow researchers to make use of the possibilities in this regard. Considering the effort and costs of human annotation, a software tool that is easily scalable can relieve researchers of some work.

Conclusion

First and foremost, we recommend considering automated approaches to voice detection in general. The amount of manually annotated quotes in our sample (see Table 1) indicates the relevance of detecting and analyzing voices. While our recommendations mainly refer to the use of direct speech detection in longer news articles, we strongly advise to always conduct a pretest with a small sample to detect easily solvable problems, e.g., regarding punctuation systems, or revalidating before applying the tools on larger datasets.

The presented findings here must be seen in the light of some limitations. Validating tools with a manually created baseline which is in itself not free from error is rightfully subject to criticism but still one of the most appropriate benchmarks available (Song et al., 2020). Further, the complexity in evaluating user experience is to assess whether errors of the automated tools hinge on the user's expertise or point out the error-proneness of the tools. Lastly, by concentrating on off-the shelf tools in our comparison, we naturally only dealt with other voice detection tools in passing.

The ongoing innovative and interdisciplinary drive in computational social science, however, allows assuming that approaches to voice detection will continue to evolve. Regarding future research, our recommendations tackle three fields: First, as the detection of indirect speech and speakers holds potential, further research should optimize existing techniques or investigate new approaches for voice detection. Secondly, research needs to evaluate the proposed software more and also from an external perspective. Methods, as emphasized, impact not only what we observe but also how and what conclusions we draw from our observation. Therefore, it is crucial to evaluate methods and draw new impulses for improvement. Especially transformer model based approaches are worth exploring next. Similarly, the promising and up-to-date applications for other languages such as STWR for German (Brunner et al., 2020) or multilingual solutions (e.g., Byszuk et al., 2020) also give much reason for closer examination.

This leads to our final and most significant recommendation: to apply existing tools in research on voices. Although the applications do not provide flawless results, they are useful, as presented here. It is evident that reported

speech is an extensively applied technique already and continues to be a relevant field of interest for communication and journalism studies.

Supplementary Materials

All supplementary materials can be found at: https://osf.io/2g3a8/

Acknowledgements

We would like to thank Arun Raveendran and Hannah Kronschnabl for their valuable assistance and insightful comments throughout this project. Further, we would also like to thank the Horizon 2020 project REMINDER (Grant agreement ID: 727072) for providing resources and support for this research project.

References

Bennett, W. L. (1990). Toward a theory of press-state relations in the united states. *Journal of Communication*, *40*(2), 103–127.

Benson, R. (2009). What makes news more multiperspectival? a field analysis. *Poetics*, *37*(5-6), 402–418.

Berkowitz, D. A. (2009). Reports and their sources. In K. Wahl-Jorgensen & T. Hanitzsch (Eds.), *The handbook of journalism studies* (pp. 165–179). Routledge.

Boumans, J., & 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.

Brown, S., Green, E., Moritz, T., Reimann, R.-P., & Speicher, S. (2017). Changing the narrative: Media representation of refugees and migrants in europe.

Brunner, A. (2015). *Automatische erkennung von redewiedergabe [automatic speech recognition]*. De Gruyter.

Brunner, A., Tu, N. D. T., Weimer, L., & Jannidis, F. (2020). To bert or not to bert. comparing contextual embeddings in a deep learning architecture for the automatic recognition of four types of speech, thought and writing representation. In S. Ebling, D. Tuggener, M. Hürlimann, M. Cieliebak, & M. Volk (Eds.), *Proceedings of the 5th swiss text analytics conference & 16th conference on natural language processing*.

Byszuk, J., Woźniak, M., Kestemont, M., Le'sniak, A., Šeľa, A., & Eder, M. (2020). Detecting direct speech in multilingual collection of 19th-century novels. In R. Sprugnoli & M. Passarotti (Eds.), *Proceedings of lt4hala 2020-1st workshop on language technologies for historical and ancient languages* (pp. 100–104).

Carpenter, S. (2008). Source diversity in us online citizen journalism and online newspaper articles. *International Symposium on Online Journalism*, *4*(1), 3–28.

Crawley, H., McMahon, S., & Jones, K. (2016). *Victims and villains. migrant voices in the british media*. Centre for Trust, Peace; Social Relations.

Elson, D. K., & McKeown, K. R. (2010). Automatic attribution of quoted speech in literary narrative. *Twenty-fourth AAAI conference on artificial intelligence*, *24*(1). https://doi.org/10.1609/aaai.v24i1.7720

Fengler, S., & Kreutler, M. (2020). *Stumme migranten, laute politik, gespaltene medien. die berichterstattung über flucht und migration [silent migrants, noisy politics, divided media. the coverage of flight and migration]*. Otto Brenner Stiftung.

Gans, H. J. (1979). Deciding what's news: Story suitability. *Society*, *16*, 65–77. https://doi.org/10.1007/BF02701600

Goutte, C., & Gaussier, E. (2005). A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In D. E. Losada & J. M. Fernandez-Luna (Eds.), *Advances in information retrieval* (pp. 345–359). Springer Berlin Heidelberg.

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.

Hall, S., Critcher, C., Jefferson, T., Clarke, J., & Robert, B. (1978). *Policing the crisis: Mugging, the state and law and order*. Macmillan Press.

Hassenzahl, M., & Tractinsky, N. (2006). User experience - a research agenda. *Behaviour & Information Technology*, *25*(2), 91–97. https://doi.org/10.1080/01449290500330331

He, H., Barbosa, D., & Kondrak, G. (2013). Identification of speakers in novels. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1)*, 1312–1320. https://aclanthology.org/P13-1129

Hepp, A. (2016). Kommunikations- und medienwissenschaft in datengetriebenen zeiten [communication and media studies in data-driven times]. *Publizistik*, *61*(3), 225–246.

Krestel, R., Bergler, S., & Witte, R. (2008). Minding the source: Automatic tagging of reported speech in newspaper articles. *Reporter*, *1*(5), 4.

Krippendorff, K. (2004). Reliability in content analysis. *Human Communication Research*, *30*(3), 411–433.

Laugwitz, B., Schrepp, M., & Held, T. (2006). Konstruktion eines fragebogens zur messung der user experience von softwareprodukten [construction of a questionnaire to measure the user experience of software products]. In A. M. Heinecke & H. Paul (Eds.), *Mensch & computer 2006: Mensch und computer im strukturwandel [human & computer 2006: Human and computer in structural change]* (pp. 125–134).

Lazaridou, K., Krestel, R., & Naumann, F. (2017). Identifying media bias by analyzing reported speech. *2017 IEEE International Conference on Data Mining (ICDM)*, 943–948.

Lind, F., Heidenreich, T., Eberl, J.-M., Galyga, S., Edie, R., Herrero-Jiménez, B., Gómez Montero, E. L., Berganza, R., & Boomgaarden, H. G. (2020). *REMINDER: Historical Media Analysis on Migration 2003-2017 (OA edition)*. https://doi.org/10.11587/IEGQ1B

Löblich, M. (2010). *Die empirisch-sozialwissenschaftliche wende in der publizistik- und zeitungswissenschaft [the empirical-social scientific turn in journalism and newspaper studies]*. Halem.

Mahrt, M. (2015). Mit big data gegen das "ende der theorie"? [with big data against the "end of theory"?] In A. Maireder, J. Ausserhofer, C. Schumann, & M. Taddicken (Eds.), *Digitale methoden in der kommunikationswissenschaft [digital methods in communication studies]* (pp. 23–37). Springer.

Manning, C., Krestel, R., & Naumann, F. (2014). The stanford corenlp natural language processing toolkit. *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 55–60. https://doi.org/10.3115/v1/P14-5010

Manning, P. (2001). *News and news sources: A critical introduction*. Sage.

Muzny, G., Fang, M., Chang, A. X., & Jurafsky, D. (2017). A two-stage sieve approach for quote attribution. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, *1*, 460–470.

O'Keefe, T., Pareti, S., Curran, J. R., Koprinska, I., & Honnibal, M. (2012). A sequence labelling approach to quote attribution. *Proceedings of the 2012 Joint Conference on Empirical Methods on Natural Language Processing*, 790–799.

Pareti, S. (2015). *Attribution: A computational approach* (Doctoral dissertation). University of Edinburgh.

Pareti, S., O'Keefe, T., Konstas, I., Curran, J. R., & Koprinska, I. (2013). Automatically detecting and attributing indirect quotations. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 989–999.

Pouliquen, B., Steinberger, R., & Best, C. (2007). Automatic detection of quotations in multilingual news. *Proceedings of Recent Advances in Natural Language Processing*, 487–492.

Reich, Z. (2011). Source credibility and journalism. between visceral and discretional judgment. *Journalism Practice*, *5*(1), 51–67.

Scheible, C., Klinger, R., & Padó, S. (2016). Model architectures for quotation detection. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 1736–1745.

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*, *37*(4), 550–572.

Strippel, C., Bock, A., Katzenbach, C., Mahrt, M., Merten, L., Nuernbergk, C., & Waldherr, A. (2018). Die zukunft der kommunikationswissenschaft ist schon da, sie ist nur ungleich verteilt [the future of communication science is already here, it's just unevenly distributed]. *Publizistik*, *63*(1), 11–27.

Thorbjørnsrud, K., & Figenschou, T. U. (2016). Do marginalized sources matter? a comparative analysis of irregular migrant voice in western media. *Journalism Studies*, *17*(3), 337–355.

Tuchman, G. (1972). Objectivity as strategic ritual: An examination of newsmen's notions of objectivity. *American Journal of Sociology, 77*(4), 660–679. Retrieved March 13, 2023, from http://www.jstor.org/stable/2776752

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.

Van Atteveldt, W., Sheafer, T., Shenhav, S. R., & Fogel-Dror, Y. (2017). Clause analysis: Using syntactic information to automatically extract source, subject, and predicate from texts with an application to the 2008–2009 gaza war. *Political Analysis, 25*(2), 207–222.

Waldherr, A. (2019). Messinstrumente und sinnkonstruktionen: Methoden als antreiber und taktgeber der kommunikationswissenschaft [measuring instruments and constructions of meaning: Methods as drivers and pacesetters in communication studies]. *Medien & Zeit, 34*(1), 40–47.

Welbers, K., van Atteveldt, W., & Kleinnijenhuis, J. (2021). Extracting semantic relations using syntax. *Computational Communication Research, 3*(2), 180–194.