

Automating the Detection of Linguistic Intergroup Bias Through Computerized Language Analysis

Journal of Language and Social Psychology
2025, Vol. 44(3-4) 343–366
© The Author(s) 2025



Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/0261927X251318887
journals.sagepub.com/home/jls



Katherine A. Collins¹  and Ryan L. Boyd² 

Abstract

Linguistic bias is the differential use of abstraction, or other linguistic mechanisms, for the same behavior by members of different groups. Abstraction is defined by the Linguistic Category Model (LCM), which defines a continuum of words from concrete to abstract. Linguistic Intergroup Bias (LIB) characterizes the tendency for people to use abstract words for undesirable outgroup and desirable ingroup behavior and concrete words for desirable outgroup and undesirable ingroup behavior. Thus, by examining abstraction in a text, we can understand the implicit attitudes of the author. Yet, research is currently stifled by the time-consuming and resource-intensive method of manual coding. In this study, we aim to develop an automated method to code for LIB. We compiled various techniques, including forms of sentence tokenization, sentiment analysis, and abstraction coding. All methods provided scores that were a good approximation of manually coded scores, which is promising and suggests that more complex methods for LIB coding may be unnecessary. We recommend automated approaches using CoreNLP sentiment analysis and LCM Dictionary abstraction coding.

Keywords

natural language processing, LIWC, linguistic category model, text analysis, linguistic intergroup bias, implicit bias, language, biased language, social bias

¹University of Saskatchewan, Saskatoon, SK, Canada

²University of Texas at Dallas, Richardson, TX, USA

Corresponding Author:

Katherine A. Collins, Department of Psychology and Health Studies, University of Saskatchewan, Saskatoon, SK, Canada.

Email: katie.collins@usask.ca

It is a given that what we say reflects what we think—we wield language as a tool of explicit self-expression. Surprisingly, however, our words can also reveal thoughts that we did *not* intend to express—we tend to unintentionally use words that align with our implicit beliefs, habits, and broader psychological composition (Boyd & Markowitz, 2024; Boyd & Schwartz, 2021). It has been argued that these subtle and systematic asymmetries in language use, or linguistic biases, are a mechanism through which pervasive implicit beliefs, such as stereotypes, are spread and maintained. Unfortunately, research in this area is currently stifled by the time-consuming and resource-intensive process required to detect linguistic bias. In this paper, we report on our development of an automated approach to the detection of Linguistic Intergroup Bias.

Bias, including stereotypes and prejudices, are subtly embedded within the words we use; our language *implies* rather than expresses them. Though this is done in a number of ways (e.g., grammatical structures, norm referents, marked exclusions, etc.), one of the major means is through the description of an individual's or group's behavior (Beukeboom & Burgers, 2019). This is because stereotypes, by their nature, define what is expected for the behavior of group members. In the linguistic bias paradigm, researchers have consistently demonstrated that these expectancies are reflected in language use through the use of different descriptions for the *same* behavior by members of *different* groups.

Linguistic bias can be defined as the differential use of abstraction. In Psychology, abstraction is commonly measured with the Linguistic Category Model (LCM; Semin & Fiedler, 1988), which outlines four different word types that can be placed along a continuum from concrete to abstract: Descriptive action verbs (DAVs), interpretive action verbs (IAVs), state verbs (SVs), and adjectives (ADJs), respectively. Some researchers have argued that a fifth word type—nouns (NNs)—be added as the most abstract category (Carnaghi et al., 2008; Seih et al., 2017). While any of these word types can be used to accurately describe a single behavior, they vary in terms of their cognitive implications. For example, if Person A hits Person B, the behavior can be described as *A hits B* (DAV) or *A is a bully* (NN). Both may be accurate but, in going from concrete to abstract, the description loses information about the specific details of the event and gains information about the individual performing the behavior. That is, low levels of abstraction maintain a strong link to the specific physical details of the behavioral event (*hits*) while higher levels of abstraction make assumptions or generalizations about the performer (*bully*).

In her classic work, Anne Maass and colleagues (Maass et al., 1989, 1995) provided the first demonstration of linguistic bias: The Linguistic Intergroup Bias (LIB). This is now the most researched and well-known form of linguistic bias. It characterizes the tendency for people to describe desirable ingroup and undesirable outgroup behaviors abstractly, and undesirable ingroup and desirable outgroup behaviors concretely. This pattern implies a bias—that only desirable behavior can be expected from the ingroup and only undesirable behavior can be expected from the outgroup. The expression of LIB is explained by a motivational process in which the ingroup is advantaged, without reliance on stereotypic expectancies (Maass, 1999)—a prejudiced attitude rather than a stereotypic belief. In contrast, the Linguistic Expectancy Bias (LEB; Wigboldus et al.,

2000) characterizes the tendency to abstractly describe behaviors that align with stereotypic expectancies and concretely describe behaviors that do not align with stereotypic expectancies. The same behavior may thus be described abstractly when it is expected, e.g., *he is a genius*, or concretely when it is not expected, e.g., *she writes the answer to the math problem*, based on stereotypes (e.g., *men are better at math than women*). The expression of LEB is explained by a cognitive process that relies on stereotypic expectancies (Maass, 1999; Wigboldus et al., 2000).

It is not possible, however, to infer whether the expression of linguistic bias is the result of a motivational or cognitive process based on an examination of language alone. A general pattern for the *interpretation* of linguistic bias thus holds: People describe the behaviours of others using language that reflects their implicit beliefs or attitudes. Simply put, linguistic bias is the tendency to use abstract descriptions for belief-consistent behavior and concrete descriptions for belief-inconsistent behavior. This distinction characterizes many forms of linguistic bias. For example, more irony and negations are used to describe belief-inconsistent rather than consistent information (Beukeboom et al., 2010; Burgers & Beukeboom, 2016). This systematic distinction results in the perpetuation of these beliefs and attitudes (Bourhis & Maass, 2005); the impact of belief-inconsistent behavior is limited through the use of concrete language by focusing on the specific circumstances in which the behavior takes place, while the impact of belief-consistent behavior is maximized through the use of abstract language by encouraging generalizations beyond the behavior. Linguistic biases are thus a reflection of the speaker's implicitly held stereotypes, and abstraction is one mechanism through which these biased beliefs are subtly expressed. In this study, we will focus on the Linguistic Intergroup Bias given that it is particularly suited to computerized analysis.

Despite the existence of prohibitive norms that act to suppress the explicit expression of stereotypes, linguistic bias is commonplace (Beukeboom, 2014). It occurs in many contexts, including interpersonal conversations, courtrooms, and media, and languages, such as English, Italian, Dutch, and French, and is expressed without intention or awareness (Franco & Maass, 1996). Given the ubiquity of conditions under which such unintentional biases in verbal behavior can be found, linguistic bias has been used as an implicit measure of prejudice (von Hippel et al., 1997). This work suggests that it could be our consistent immersion in, and repeated exposure to, biased language that leads to stereotype sharedness.

The linguistic bias paradigm, which we use to refer to research on how bias or implicit beliefs about social groups are reflected in subtle linguistic formulations, is particularly suited to the study of current socio-cultural issues. Politician's claims of "fake news" and the emergence of social media reporting have increased concerns about the trustworthiness of news (e.g., Domonoske, 2018; Lombrozo, 2018; Williams & Nettlefold, 2018). These events are sparking controversy and timely conversations about stereotypes and biased reporting, to which linguistic bias is directly applicable. The public is primed for insights of this nature and in need of a method to quantify bias. There are websites, for example, dedicated to providing ratings of bias in various outlets (e.g., Media Bias¹; AllSides²). Yet despite this link, there is a

disconnect between researchers and consumers of news media: Media Bias, for example, (1) includes subjective ratings from the public, similar to movie-rating sites, and (2) does not include any of the objectively measurable linguistic biases used by researchers. AllSides has multiple rating systems and state that they sometimes consider academic research or third-party data, but these appear to be studies of media outlets (e.g., Fox News) and not forms of linguistic bias that could occur within any article. Thus, at this time, linguistic bias research is uniquely positioned to have a potentially large and meaningful impact but is failing to do so.

There are two main reasons why researchers are prevented from seizing this opportunity. First, research in this area relies on manual coding to detect linguistic bias, which is time-consuming and resource-intensive. To maintain objectivity, it is necessary to first train independent coders to identify word types from the LCM (e.g., training manual by Coenen, Hedeboew, & Semin, Personal Communication, May 19, 2014). Once trained, coders will do a detailed, word-by-word, reading of the text, which is a tedious and lengthy process. Once complete, an expert rater resolves differences in codes using the same intensive process. Once manual coding is complete, researchers can then calculate an Abstraction score. In this calculation, the number of each type of word category is multiplied by a weight determined by the theoretical level of abstraction as defined by the LCM, then divided by the total number of words³:

$$\frac{(\text{DAV} \times 1) + (\text{IAV} \times 2) + (\text{SV} \times 3) + (\text{ADJ} \times 4) + (\text{NN} \times 5)}{\text{DAV} + \text{IAV} + \text{SV} + \text{ADJ} + \text{NN}}$$

As shown, higher weights indicate a higher level of abstraction. Abstraction scores thus range from 1 (*concrete*) to 5 (*abstract*). Despite the widespread use of this scoring method, it should be noted that, to our knowledge, this numerical coding scheme has never been psychometrically tested. Given the process of manual coding, linguistic bias detection is not a reactive approach; detection cannot usually take place until well after the current event is no longer news.

Second, due to the reliance on manual coding, the length and number of texts that can be coded is limited by resources and time, so researchers typically analyze a small number of texts that are relatively short in length. Historically, for example, researchers have asked participants to directly compare single words or multiple variations of a single sentence, which may place undue emphasis on the word types. Others have analyzed texts that have been spontaneously generated by participants or used carefully and systematically manipulated texts, but this data is still low in number and short in length. That is not to say that all research in this area is limited in this way (e.g., Dragojevic et al., 2017), only that to do otherwise requires a considerable investment of resources. Further, texts are analyzed for a single bias instead of a more complex combination of biases as likely occurs in reality.

It is thus no surprise that researchers have turned towards alternative methods. There are a variety of algorithmic approaches to the coding of concreteness, which involve two general approaches (see Yeomans, 2021, for a review). In word-level approaches, researchers develop large corpora of individual words that have been rated out of context by people. These ratings are then used to calculate the concreteness scores

of other texts. For example, Brysbaert et al. (2014) crowdsourced ratings of over 60,000 word and two-word expressions, to build a data set of mean concreteness ratings for nearly 40,000 well-known English words. Although these ratings have been used to predict manually coded LCM scores (by Johnson-Grey et al., 2020), the meaning of these ratings do not align with the LCM or its intent to differentiate word categories. While the concrete ratings provide a reasonable proxy for the LCM, as they were defined as *those things that can be experienced through the senses or by action* (e.g., *jump*), the abstract ratings do not, as they were defined as *those things that cannot be experienced through the sense or by action and are best explained through words* (e.g., *justice*). Brysbaert concreteness ratings then do not, and are not intended to, capture that abstract descriptions of the LCM involve the generalization of a behavior. As with Brysbaert, automated methods of coding concreteness using word-level approaches typically do not easily correspond to the LCM.

In categorical approaches, researchers first identify different word categories then count the number of instances of each category in a text, which is similar to manual coding of the LCM. A variety of different categories have been used, with only three being based on the LCM. Stone et al.'s (1966) ground-breaking study resulted in the first automated method of coding abstraction, but only for the first three levels of the LCM. Seih et al. (2017) built on this work and extended the automated method to code for the remaining two levels of the LCM. In this study, the researchers created an LCM dictionary of the most common 7489 verbs by compiling verbs from the General Inquirer LCM Dictionary (Stone et al., 1966) and a corpus of approximately 44 million words, asking coders to classify the verbs into DAVs, IAVs, and SVs, and adding different verb forms (e.g., *make*, *makes*, *made*, and *making*). This dictionary can be used in Linguistic Inquiry and Word Count (LIWC; Pennebaker et al., 2007) to automatically count the number of DAVs, IAVs, and SVs within a given text. The researchers then used TreeTagger⁴ to identify ADJs and NNs and to convert these parts of speech tags into count ratios. The abstraction score is calculated in the same way as with manual codes. This computerized method successfully distinguished between the levels of the LCM and these results aligned with those generated by manual coding.

Johnson-Grey et al. (2020) built on the approach of Pennebaker et al. (2007) to not only include word categories but also syntactic features common to concrete and abstract texts. By manually coding a corpus of 1,439 sentences to identify the top third most concrete and top third most abstract texts, the researchers were able to identify 22 syntactic features that distinguish between concrete and abstract (verbs + adjectives) categories. For example, the presence of a *numeric modifier* was more common in concrete texts and use of the *third-person* was more common in abstract texts. The researchers then calculated a Syntax-LCM score, which is a typical Abstraction score modified to include syntactic features (SF):

$$\frac{([DAV + ConcreteSF] \times 1) + ([IAV + VerbSF] \times 2) + (SV \times 3) + ([ADJ + AdjectiveSF] \times 4)}{(DAV + ConcreteSF + IAV + VerbSF + SV + ADJ + AdjectiveSF)}$$

Notably, this approach did not include NNs. Nevertheless, the analyses demonstrated that manual codes from three separate data sets had a stronger correlation with Syntax-LCM than the LCM Dictionary and that the Syntax-LCM made a unique contribution beyond that of the LCM Dictionary in predicting abstraction scores based on manual codes.

Based on a review of the various automated approaches to the coding of concreteness, Yeomans (2021) found that the expression of concreteness is domain specific (i.e., a product of the area or context), in that automated and manual concreteness codes were more strongly correlated within specific domains rather than across domains. Given this, he recommends manual coding, machine learning, or word-level approaches to the automated coding of concreteness. Though both LCM-based measures performed well in descriptive texts, they performed less well in other domains (e.g., advice). However, this might be expected given that the LCM arose out of research on the causality implicit in various descriptions of behavior (Semin, 2012). Thus, it is arguable, and in line with Yeoman's (2021) findings, that LCM-based measures are best suited to that domain. The lower performance of the LCM-based measures in other domains may also be expected considering that the manual codes of concreteness were not based on the LCM, but rather a subjective evaluation of the texts along various dimensions, including specificity, actionability, abstractness/generality, or, most simply, condition based on the manipulation of construal level. One could argue that a strong association between LCM-based codes and these types of manual concreteness should not be expected. In any case, these two LCM-based measures tend to be positively correlated (Yeomans, 2021) and are the most appropriate within the descriptive texts typical of the linguistic bias paradigm as well as being the most interpretable, in that they align well with manual codes of the LCM. It is for these reasons that we will focus on these two automated approaches to abstraction coding in the current study.

These automated approaches to the calculation of abstraction offer a substantial reduction in the time, effort, and resources required to manually code abstraction. However, they are not sufficient to capture the meaning of *Linguistic Intergroup Bias* because the impact of abstraction changes depending on the nature of the context. A negative bias, for example, would be the use of abstract language to maximize the impact of an undesirable behavior (*she is a liar*) whereas a positive bias would be the use of abstract language to maximize the impact of a desirable behavior (*she is a caregiver*). Importantly, a determination of whether a single text about a specific person or social group is biased relies on the balance of abstraction of both desirable and undesirable behaviors. The Linguistic Intergroup Bias is usually determined by a two-way interaction between content (desirable versus undesirable behavior) and group membership of the described person or social group (e.g., ingroup versus outgroup member) on abstraction scores. In cases where there is only one social group involved or where a single measure of bias is needed (e.g., Maass et al., 1995), a bias index is used. The calculation of a bias index is typically:

$$\text{Desirable Abstraction} - \text{Undesirable Abstraction}$$

A bias score for a single text thus ranges from -5 (*negative bias*) to $+5$ (*positive bias*),⁵ with scores at, or around, 0 indicating a lack of bias—or an equal balance in the abstracted depiction of desirable and undesirable information.

There has been much research on the automated detection of bias. Researchers have, for example, created models to detect explicit social biases in Wikipedia articles (e.g., Field et al., 2022; Hube, 2017), classify hate speech on Twitter (e.g., Davidson et al., 2017), identify linguistic indicators of fake news (e.g., Rashkin et al., 2017) and media bias (e.g., Spinde et al., 2021), and describe framing devices in media coverage of police violence (e.g., Ziems & Yang, 2021). However, to date, there has been no published attempt to automate the detection of Linguistic Intergroup Bias, as defined and used within the linguistic bias paradigm. To address this gap, we present our development of an automated method to detect Linguistic Intergroup Bias, which involves combining automated LCM-based measures of abstraction along with sentiment analysis.

Manually Coded Data

For the development of our methods, we used unpublished data from a previous study (Collins et al., 2018). In this study, 80 participants were randomly assigned to one of two versions of a biased text, which described 12 behaviors of one fictional group member. The texts included six examples each of desirable (*friendly*, *hospitable*) and undesirable (*intrusive*, *sexist*) behaviors. In the positively biased version of the text, desirable behaviors were described using abstract language and undesirable behaviors were described using concrete language. In the negatively biased version of the text, desirable behaviors were described using concrete language and undesirable behaviors were described using abstract language. A pilot study ensured that concrete and abstract descriptions were perceived by participants as equally desirable or undesirable in valence.

Participants in the study believed they were evaluating the usefulness of secondary information in textbooks. The biased text was ostensibly an excerpt from an ethnography featured in an Anthropology textbook. After exposure to the biased text, participants were asked to describe the fictional group member and the events from the text as they would to friends. More specifically, the question was

Imagine that a couple of your friends are going to [the fictional country]. You want to help your friends know what to expect on their trip. Please describe [name of fictional group member] and the events reported in the ethnography as you would describe them to your friends. Try to be as accurate and detailed as possible.

Participants were able to write their response in an open text box with no character limit. We did not measure or control the degree to which participants repeated the exact behaviors described in the biased text.

Of the 80 resultant texts, four were excluded because they did not answer the question ($n = 3$) or showed awareness that the described group was fictional ($n = 1$), leaving a total of 76 texts that were manually coded for linguistic bias by two independent raters who were blind to both the study's purpose and participant condition. The coders were first provided with information on how to code for the LCM via familiarity

with the original LCM article by Semin and Fiedler (1988) and a coding manual by Coenen, Hedeboom, and Semin (personal communication, May 19, 2014). KC explained the process, provided an example, answered questions, and created a template spreadsheet with one text per row and columns for desirable content, undesirable content, and abstraction.

Though the nature of the question ensured that most sentences were relevant, coders were instructed to first identify relevant statements or those that were about the fictional group or fictional group member. Coders then determined valence by determining whether the statement was positive or negative and assigned it a value from 1 to 5 in correspondence to the levels of the LCM with the inclusion of nouns (Carnaghi et al., 2008). Finally, coders calculated a mean score separately for desirable and undesirable content. KC merged the spreadsheets for analysis.

Interrater reliability of both valence and abstraction was assessed separately using Cohen's kappa, as calculated by the *irr* package version 0.84.1 (Gamer et al., 2019) for R version 4.3.2 (R Core Team, 2023) with confidence intervals by the *psych* package version 2.4.3 (Revelle, 2024). Based on Cohen's unweighted kappa for categorical data, there was almost perfect agreement between raters on valence, $\kappa = .910$, 95% CI [0.868, 0.952], $p < .001$, indicating that 82.81% of data were reliable (McHugh, 2012). A weighted version of Cohen's kappa was used for interrater reliability of abstraction, given that this data involves more than two ordinal levels. Interrater reliability was substantial, $\kappa = 0.855$, 95% CI [0.794, 0.916], $p < .001$, indicating that 73.10% of data were reliable (McHugh, 2012). Discrepancies were resolved by an expert independent rater, who was also blind to the study's purpose and participant condition, except in three cases that were missed by the expert rater. In these cases, the mean score was used.

The results indicated that, in general, participants expressed a Linguistic Intergroup Bias that corresponded with the biased text to which they were exposed. That is, those participants exposed to the positively biased text expressed a bias in which they described desirable behavior abstractly and undesirable behavior concretely. Those participants exposed to the negatively biased text expressed the opposite pattern, though this trend was not significant. This original data provided a corpus of texts alongside manually coded LCM abstraction scores, as well as a bias index that was calculated based on the abstraction scores and the un/desirable nature of the described behavior.

autoLIB Development

Our aim was to develop an automated approach to the coding of Linguistic Intergroup Bias or autoLIB. To do this, we compared eight automated approaches that were generated based on six existing methods from natural language processing. There were four main steps. The first step was to break down each available text into smaller units or tokens for analysis, which is generally referred to as tokenization. The second step was to identify the sentiment or overall desirability (positivity or negativity) of each token. The third step was to code each token for linguistic abstraction

according to the LCM, which resulted in an abstraction score for each token and mean abstraction scores for desirable content and undesirable content within each text. The fourth and final step was to calculate a bias index based on the mean abstraction scores for desirable and undesirable content within each text. These steps are outlined below in further detail.

Sentence Tokenization

The first step was to split the original 76 texts into individual sentences, similar to the approach of Johnson-Grey et al. (2020). We used two methods of sentence tokenization that were freely available through BUTTER³, a free and open-source software that provides an interface to access a wide range of text analysis tools without requiring the use of programming, and R. First, from BUTTER, we used the regular expression below, which segmented the texts into a total of 262 sentences:

$$(? < ! \backslash w \backslash . \backslash w .) (? < ! [A - Z] [a - z] \backslash .) (? < = \backslash . | \backslash ?) \backslash s$$

The mean number of words per sentence was 15.72 ($SD = 7.95$). Second, we used the *tidytext* package (version 0.3.2; Silge & Robinson, 2016) to segment the texts into a total of 253 sentences. The mean number of words per sentence was 16.28 ($SD = 8.41$).

Sentiment Analysis

The second step was to use established methods for sentiment analysis to categorize each sentence into desirable and undesirable content. We compared two methods of sentiment analysis, both of which were implemented using BUTTER. At the time we started this work, we chose VADER and CoreNLP based on both accessibility and performance, though technological advancements in the field of natural language processing mean that there are a variety of other methods that could be tested.

First, we used VADER-tots,⁷ which tries to disambiguate positive and negative words in a simple but effective manner. It is based on the open-source VADER (Valence Aware Dictionary and sEntiment Reasoner) sentiment analysis tool (Hutto & Gilbert, 2014) and its port to other systems,⁸ which is specifically attuned to the short texts typical of social media. The ultimate classification of a sentence from this approach uses the “compound” measure, which weighs in both positive and negative scores into a composite score. If the compound measure is greater than .05, then it treats the statement as positive, and if the compound measure is less than −.05 is treated as negative. Compound measures between −.05 and +.05 are classified as neutral.

Second, we used Stanford’s CoreNLP framework⁹ (Manning et al., 2014) for sentiment analysis. The CoreNLP framework is a suite of text analysis tools written in Java, including parts-of-speech tagging (Toutanova et al., 2003) and sentiment analysis (Socher et al., 2013). CoreNLP sentiment analysis accounts for the context of a word (e.g., contrasting conjunctions and negations), making it more sophisticated than simply categorizing words based on their definition (e.g., *terrible*, *great*). In

this method, each sentence is classified into 1 of 5 categories: “Very Negative,” “Negative,” “Neutral,” “Positive,” and “Very Positive.”

It is important to note that both methods of sentiment analysis process the sentence as an entire unit. More specifically, both VADER and CoreNLP assign a single sentiment score to the entire sentence based on the overall sentiment conveyed. This means that the nuances of mixed sentiment may not be fully captured when sentences contain clauses of opposing sentiment, which is a known issue in sentiment analysis. In this study, sentences were aggregated by participant thereby minimizing the noise from more complex linguistic constructions.

LCM Abstraction

The third step was to use previously established methods for automated coding of abstraction for all sentences: LCM Dictionary (Seih et al., 2017) and Syntax-LCM (Johnson-Grey et al., 2020).¹⁰ Both methods for LCM coding produce an abstraction score, which ranges from 1 to 5 with a higher number indicating more abstraction similar to abstraction scores from manual coding.

The LCM Dictionary, as used in LIWC (Pennebaker et al., 2007), contains only DAVs, IAVs, and SVs. Seih et al. (2017) then used the open-source TreeTagger to count ADJs and NNs. Similar to Johnson-Grey et al. (2020), we identified parts-of-speech using the widely used *Stanford.NLP.NET POS Tagger*,¹¹ again via BUTTER. From the output, the columns for “JJ,” “JJR,” “JJS” were summed to obtain a count of ADJs, and the columns “NN,” “NNS,” “NNP,” “NNPS” were summed to obtain a count of NNs based on the Penn Treebank.¹² The LCM Dictionary approach most clearly aligns with the manual coding of the LCM. It involves the automated identification of the various word categories of the original LCM (Semin & Fiedler, 1988), resulting in an abstraction score with the same range and interpretation of manually coded abstraction scores.

The Syntax-LCM approach (Johnson-Grey et al., 2020) builds on the LCM Dictionary approach to include both part-of-speech tags and syntactic features to approximate abstraction scores (as described earlier). The calculation involves combining the weights derived from the LCM along with syntactic features (that identify verbs and adjectives) within each sentence. Johnson-Grey et al. (2020) argue that these syntactic features are appropriately included in LCM calculations given that syntax is often used to help determine word categories. Despite the addition of syntactic features, the abstraction score derived using the Syntax-LCM method follows the same range and interpretation as manually coded LCM abstraction scores. The scores were computed in R using modified scripts available as supplementary files from Johnson-Grey et al. (2020).

Bias Index

In the final step, we calculated a bias index. When VADER was used for sentiment analysis, mean abstraction for the negative sentences was subtracted from the mean abstraction for the positive sentences, and the neutral category was ignored. When

CoreNLP was used for sentiment analysis, mean abstraction for the sentences classified as Very Negative and Negative was subtracted from the mean abstraction for the sentences classified as Very Positive and Positive, and, again, the neutral category was ignored. Positive and negative categories are referred to as desirable and undesirable content, respectively, to avoid confusion with the positively and negatively biased conditions of Exposure. A positive bias score indicates a higher level of abstraction for desirable content compared to undesirable content, a negative bias score indicates a higher level of abstraction for undesirable content compared to desirable content, and a bias score around 0 indicates no difference in the level of abstraction between desirable and undesirable content.

Results

Comparison of Automated Methods to Manual Coding

Correlations. Each generated automated bias index was correlated with the original manually coded bias index. As shown in Table 1, the bias indices from all automated methods were highly ($p < .001$) and strongly ($>.30$; Gignac & Szodorai, 2016; $>.33$ Weinerová et al., 2022) correlated with the bias index based on manual codes. Sentence tokenization made little difference in the correlations. Methods using VADER for sentiment analysis resulted in slightly stronger correlations with the manually coded bias index. Approaches with both CoreNLP and Syntax-LCM seem to have slightly weaker correlations with the manually coded bias index.

These relationships are shown visually in Figure 1. The confidence intervals for all methods were narrowest near zero or neutrality, and widest at the extremes or regions

Table 1. Correlations between automated bias indices and manually coded bias index.

Tokenization	Sentiment	Abstraction	Bias index			<i>r</i> <i>df</i> = 74
			Min	<i>M</i> (SD)	Max	
Manually coded texts regex	VADER	LCM Dictionary	−4.00	0.25 (1.89)	4.30	1
		Syntax-LCM	−3.92	1.99 (2.06)	4.20	.446***
	CoreNLP	LCM Dictionary	−2.44	1.28 (1.34)	3.25	.457***
		Syntax-LCM	−4.20	−0.54 (2.31)	4.19	.391***
tidytext	VADER	LCM Dictionary	−3.20	−0.33 (1.51)	3.25	.358**
		Syntax-LCM	−3.92	1.98 (2.05)	4.20	.443***
	CoreNLP	LCM Dictionary	−2.44	1.30 (1.33)	3.25	.460***
		Syntax-LCM	−4.20	−0.57 (2.37)	4.19	.388***
			−3.20	−0.37 (1.53)	3.25	.354**

Note: Bias indices using manual coding and LCM Dictionary can range from −5 to +5 and bias indices using Syntax-LCM can range from −4 to +4. A positive score indicates a positive bias, a negative score indicates a negative bias, and scores at or near zero indicate neutrality. The table presents observed minimum, mean, and maximum for each bias index, and the correlation between each automated bias index and the bias index resulting from manual coding. ** $p \leq .01$; *** $p \leq .001$.

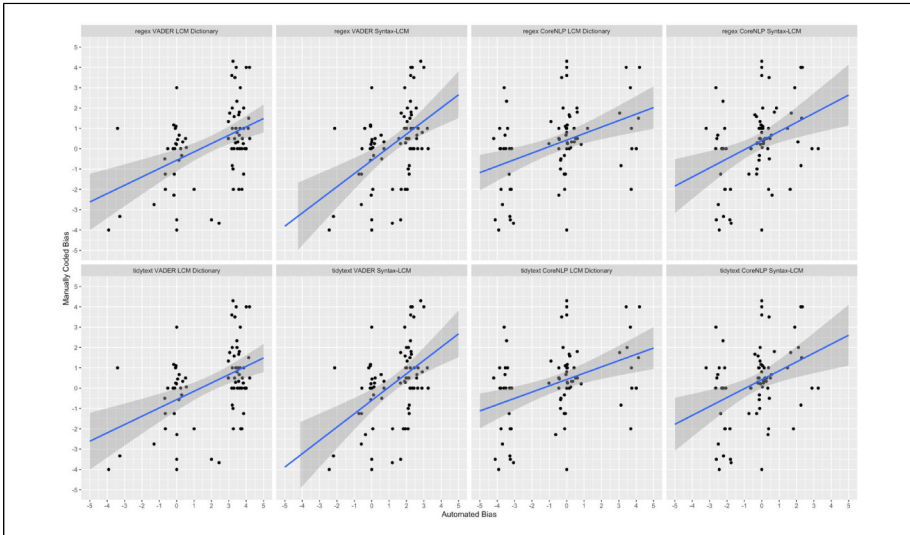


Figure 1. Scatterplots between automated bias indices and manually coded bias index.

Note: The y-axis is manually coded bias, while each plot shows a different automated bias index along the x-axis, with the line of best fit and 95% confidence interval. Rows represent sentence tokenization, with the first row being regex and the second row being tidytext. VADER methods appear in the four plots to the left, while CoreNLP methods appear in the four plots to the right.

indicating bias. Methods using VADER had a narrower confidence interval in the positive bias region and a wider confidence interval in the negative bias region, suggesting that use of VADER leads to more estimates of positive bias. Methods using CoreNLP had more regular confidence intervals, with those in combination with LCM Dictionary having the narrowest confidence intervals throughout the range. Further, these methods appear to result in three groups corresponding to negative bias, neutrality, and positive bias, with most estimates occurring in the neutral region.

The automated bias indices were also significantly and strongly correlated with each other. In Figure 2, the correlation matrix is arranged to best show the pattern of relationships. From the correlations, we can see that both sentence tokenization and LCM-based coding made little to no difference. Further, approaches with the same sentiment analysis method tend to be more highly correlated together: VADER with other VADER approaches and CoreNLP with other CoreNLP approaches, with lower, but still significant correlations, across methods of sentiment analysis. A visual examination of histograms showed that most automated variables may not align with a normal distribution.

Replication of Original Results

Our aim was to use the eight generated automated approaches to replicate the results of the original study, which involved three main parts. First, a mixed analysis of variance

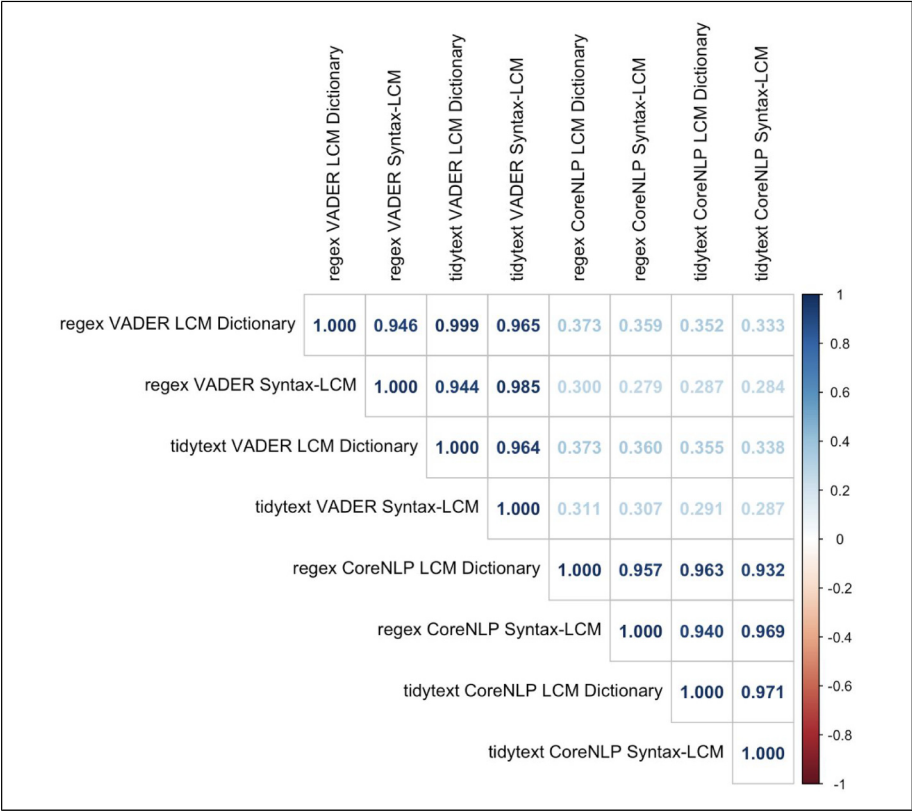


Figure 2. Simplified correlation matrix.

on abstraction scores to determine whether participants described desirable and undesirable behaviors of the fictional group using different levels of abstraction based on the text to which they were exposed. Second, an examination of the pattern of means and simple main effects of this interaction. Third, a simplified t-test of the effect of exposure on bias indices.

Mixed Analysis of Variance. In the original data, the assumption of normality was violated as tested with Shapiro–Wilks. However, given that our sample sizes are relatively equal ($n_1 = 43$, $n_2 = 32$), the single univariate outlier was removed, degrees of freedom for error was greater than 20 ($df_e = 73$), and the use of a two-tailed test, we can expect analysis of variance to be robust to this violation (Tabachnick & Fidell, 2007, pp. 87, 319). In fact, F tests are robust to even severe departures of normality (Blanca et al., 2017). The assumption of homogeneity of variance was violated according to Levene’s test, but not according to the F_{\max} test (F_{\max} ratio = 4.58) so we proceeded with a mixed analysis of variance.¹³

We thus conducted a 2 (Exposure: Positively Biased Text or Negatively Biased Text) \times 2 (Content: Desirable and Undesirable) mixed analysis of variance on mean levels of abstraction, with Exposure as a between-subjects variable and Content as a within-subject variable. The desirability of the coded Content was determined by the independent raters. There was no significant main effect of Exposure, $F(1, 73) = 0.459$, $p = .500$, meaning that the mean level abstraction did not differ between those exposed to the positively biased versus negatively biased texts. There was no significant main effect of Content, $F(1, 73) = 2.728$, $p = .103$, meaning that the mean level of abstraction did not differ for desirable and undesirable content. There was, however, a significant interaction between Exposure and Content, $F(1, 73) = 5.622$, $p = .020$, $\eta^2_G = .038$, as hypothesized. The results of the mixed analysis of variance for manually coded abstraction scores can be seen in top row of Table 2. Table 3 shows the pattern of means with subscripts to indicate the significance of all pairwise comparisons.

Though there were similar issues in meeting the assumptions for mixed analysis of variance on abstraction scores from automated approaches, we decided to conduct the same analysis on untransformed mean abstraction scores generated from each automated method to aid both comparison and interpretability. As shown in Table 2, automated methods using CoreNLP for sentiment analysis were best able to replicate the original results, with the interaction reaching significance. However, the pattern of simple main effects for those interactions were opposite to those based on manual coding (Table 3).

We also note that both VADER and CoreNLP seem to result in a greater distinction between desirable and undesirable content in comparison to human coders (Table 3), with VADER methods resulting in a pronounced main effect of Content. Though this demonstrates the efficacy of VADER in detecting sentiment, such a strong effect may potentially overshadow the more subtle yet theoretically important interaction effect. In support of this interpretation, methods using VADER did not result in a significant interaction, while methods using CoreNLP did (Table 2).

Table 2. Main and interaction effects of the mixed analysis of variance on abstraction scores.

Method			Exposure	Content	Exposure \times content
Manually coded texts			.500	.103	.020
regex	VADER	LCM Dictionary	.292	<.001	.092
		Syntax-LCM	.467	<.001	.118
	CoreNLP	LCM Dictionary	.310	.078	.022
		Syntax-LCM	.237	.099	.042
tidytext	VADER	LCM Dictionary	.322	<.001	.104
		Syntax-LCM	.597	<.001	.148
	CoreNLP	LCM Dictionary	.569	.069	.032
		Syntax-LCM	.520	.064	.042

Note: This table presents p values for both main effects and the interaction term from a mixed analysis of variance on each bias index. Values for effects that are significant are in bold text.

Pattern of Means. In the original study, participants expressed a linguistic bias that aligned with the text to which they were exposed. As shown at the top of Table 3, participants exposed to a positively biased text had a significantly higher mean abstraction for desirable content than for undesirable content ($p = .003$, $\eta^2_G = .137$). Though participants exposed to the negatively biased text had a slightly higher mean abstraction for undesirable content than for desirable content, this difference was not significant ($p = .626$).

As can be seen from Table 3, there is little difference in the pattern of means between using regular expression and *tidytext* for sentence tokenization. When CoreNLP was used for sentiment analysis, the pattern of means aligns with those based on manual codes, though the significance of simple main effects differs. When VADER was used for sentiment analysis, only the pattern of means for those exposed to the positively biased text is in line with those based on manual codes. More specifically, the use of VADER appears to lead to a positive bias, with more content being identified as desirable (Figure 1) and with desirable content having higher abstraction regardless of condition (Table 3).

Welch's Test. We conducted a simplified alternative analysis on the bias indices. A one-tailed Welch's t -test for independent samples was run on each bias index with Exposure as the independent variable. In all cases, the test was significant or approaching significance, as shown in Table 4. Importantly, only bias indices built with CoreNLP had means that would be similarly interpreted as the manually coded bias index based on sign, but not based on magnitude. That is, the scores for those exposed to the positively biased text would likely be interpreted as close enough to zero to indicate no bias. Bias indices built with VADER led to positive means

Table 3. Pattern of means and pairwise comparisons.

Method			Positively biased		Negatively biased	
			Desirable	Undesirable	Desirable	Undesirable
Manually coded texts			3.39 _a	2.53 _b	2.74 _{a,b}	2.89 _{a,b}
regex	VADER	LCM Dictionary	3.53 _a	1.06 _b	2.90 _a	1.24 _b
		Syntax-LCM	2.19 _a	0.63 _b	1.84 _a	0.77 _b
	CoreNLP	LCM Dictionary	2.76 _{a,b}	2.61 _{a,b}	1.89 _a	2.96 _b
		Syntax-LCM	1.70 _{a,b}	1.64 _{a,b}	1.16 _a	1.80 _b
tidytext	VADER	LCM Dictionary	3.51 _a	1.06 _b	2.90 _a	1.24 _b
		Syntax-LCM	2.18 _a	0.63 _b	1.88 _a	0.78 _b
	CoreNLP	LCM Dictionary	2.62 _{a,b}	2.53 _{a,b}	1.89 _a	2.97 _b
		Syntax-LCM	1.62 _{a,b}	1.59 _{a,b}	1.16 _a	1.85 _b

Note: Subscripts show the significance of all pairwise comparisons across rows. Means with the same subscript (e.g., 3.39 and 2.74 in the first row) are not significantly different, whereas means with different subscripts (e.g., 3.39 and 2.53 in the first row) are significantly different.

across conditions, which would lead to significant deviations from an interpretation based on the manual codes.

Discussion

Our study presented the development and investigation of eight related automated approaches to the detection of Linguistic Intergroup Bias, which involved constructing all possible combinations of six existing approaches to sentence tokenization, automated coding of the LCM (involving dictionary approaches, parts-of-speech tagging, and syntactic features), and sentiment analysis. The similarly sized ($.354 \leq r \leq .460$) and highly significant ($.00003 \leq p \leq .00171$) correlations suggest that all automated approaches produced scores that were good approximations of the manual codes. It is important to note that these correlations were achieved without the influence of shared method variance, which likely results in more conservative estimates. Nonetheless, these correlations are significant and are considered substantial within the broader context of psychological research (Gignac & Szodorai, 2016; Weinerová et al., 2022), where the observed range of correlations is lower likely due to the complexity of the constructs being measured and the diverse methods employed. This indicates that automated methods, despite differences in operationalization, do capture relevant constructs akin to those assessed by human coders, which supports their convergent validity and potential utility in psychological research. These results are thus promising and suggest that automating the detection of Linguistic Intergroup Bias through computerized language analysis is not only possible but may not require more complex computational methods.

While recent technological advancements in machine learning and deep learning are exciting, they are not always the best choice for every application. If we were to use such techniques for automated Linguistic Intergroup Bias detection, it would require training an algorithm on manually coded data and developing a complex model to predict scores accurately. However, this approach has two significant drawbacks: (1) it demands a very large amount of training data, which is hard to produce within the linguistic bias paradigm, and (2) it results in an opaque solution, making it unclear how the algorithm makes decisions or what features it uses.

Instead, this study employs a method that is transparent and explainable, closely mirroring how a human would manually code the data. This theory-driven, top-down approach is particularly important for social science applications, where understanding the psychological mechanisms behind language is crucial. Any automated method for Linguistic Intergroup Bias detection must also involve the identification of its unique linguistic formulation, given that this is what distinguishes it from other linguistic biases. The correlations achieved with this method, peaking at .460, are impressive and demonstrate that simpler, theory-driven techniques can be not only sufficient but perhaps even more suitable than complex, data-driven methods for the automated detection of Linguistic Intergroup Bias. The effectiveness and rationale behind this approach are noteworthy, particularly as these preliminary results were achieved with “off the shelf” methods that were selected for their theoretical relevance, and they required minimal data for testing and no additional tuning.

Table 4. The effect of exposure on bias indices based on Welch's test.

Method		Exposure		t	p
		Positively biased	Negatively biased		
Manually coded texts regex	VADER	0.77	-0.15	-2.250	.014
		2.43	1.66	-1.70	.047
	CoreNLP	1.55	1.07	-1.67	.050
		0.07	-.64	-2.38	.010
tidytext	VADER	0.07	-.64	-2.17	.019
		2.40	1.66	-1.64	.052
	CoreNLP	1.55	1.11	-1.54	.063
		0.03	-.68	-2.20	.016
		0.03	-.68	-2.08	.020

From the correlations and the pattern of results, we find that different methods of sentence tokenization and LCM-based abstraction coding do not make a meaningful difference in bias detection. The main difference in approaches came from the selection of sentiment analysis methods (VADER or CoreNLP), though either method resulted in bias scores that were significantly and strongly correlated with the manually coded bias index. VADER approaches were slightly more strongly correlated with the manually coded bias index and CoreNLP approaches were better able to replicate the original pattern of means but not the significance pattern for simple main effects. Further, CoreNLP approaches were more likely to lead to means that would be interpreted similarly, though not exactly, as means based on manual coding.

That method of sentiment analysis made a larger difference than sentence tokenization or LCM-based coding is not surprising given that the former is relatively more subjective than the latter. While sentence tokenization and LCM-based coding methods are based on grammar and syntax, which can be constructed based on a series of rules and definitions, sentiment analysis relies on interpretation of the meaning of a sentence. This means that, by the nature of this variable, there is a larger margin for differences based on the algorithm used. Based on the results, an automated approach using CoreNLP may be preferable to one based on VADER, given the tendency for research on the Linguistic Intergroup Bias to rely on interpretation of the pattern of means. However, VADER was developed specifically for Twitter linguistic data so it may be preferable for use in social media contexts, in which unique linguistic norms have emerged (Benamara et al., 2018).

It is important to note that the LCM-based measures differ slightly in their interpretation. While the LCM Dictionary method results in scores that have a one-to-one correspondence with manually coded abstraction scores, the Syntax-LCM method does not, given that it incorporates syntactic features in the identification of levels of the LCM. Previous research found that Syntax-LCM scores were more strongly correlated with manually coded abstraction than LCM Dictionary scores (Johnson-Grey et al., 2020), but, in our correlations between bias indices, we found that neither method led to consistently stronger correlations. More specifically, approaches with Syntax-LCM were more strongly correlated with manually coded bias when combined with VADER and approaches with LCM Dictionary were more strongly correlated with manually coded bias when combined with CoreNLP. Given the greater interpretability of these scores, and that LCM-based methods did not impact the pattern of results, we recommend that future automated approaches rely on the LCM Dictionary method.

Our study thus suggests that an automated approach built on CoreNLP sentiment analysis and LCM Dictionary abstraction coding is recommended, with sentence tokenization making little or no difference. However, this study is limited in several ways. First, our analysis is based on a small corpus of texts. While the manual coding of a small number of texts is typical within the linguistic bias paradigm, given the difficult process of manual coding, a large and diverse corpus or corpora are ideal for developing and refining automated methods for natural language processing. Given this, we are building a larger corpus of manually coded texts for further testing of these automated

methods. It would also be preferable to test these automated methods on not one but multiple sources of texts, as context could impact the performance of these methods. That is, we are not able to verify whether these automated methods will perform the same across domains without further research.

Our texts were also particularly suited to detecting Linguistic Intergroup Bias as participants were asked to describe a (fictional) social group. Given this, most texts were relatively easy to manually code: Straightforward and focused on a single relevant topic, often with the targeted social group in the subject position of simple and actively phrased sentences. Linguistic Intergroup Bias is only applicable to descriptions of behavior by a performer from a specific social group of interest. In the manual coding of naturally produced texts, such as newspaper articles, only relevant sentences will be coded. We must consider then that naturally produced texts are not always so homogeneous—texts may involve the discussion of various groups, for example, or no social group at all. Using any one of these automated approaches on natural language would thus involve first identifying which sentences are about a social group of interest, which we had no need to do in this study. However, even if not able to select relevant sentences, this approach would still represent time-savings for researchers if they need only select relevant sentences and not manually code them.

Despite the relative ease of these automated approaches compared to manual coding of linguistic bias, they are still difficult for researchers to implement without computer programming experience or collaboration with computer scientists. These approaches variously require different software, both proprietary and open source, some only available on certain operating systems, as well as familiarity with different frameworks of linguistic analyses (e.g., parts-of-speech tagging, CoreNLP), then merging multiple data sets from these different sources. Given this, these automated methods are not feasible for many researchers. Future research should develop a simplified method of Linguistic Intergroup Bias detection, such as a dedicated R package or BUTTER plugin, which would make these methods more accessible. It will then be much easier for future researchers to test these methods within different domains and refine them as necessary.

It is important to acknowledge that a bias found with these automated approaches to the coding of Linguistic Intergroup Bias may not *only* represent the Linguistic Intergroup Bias. Stereotypes tend to be disproportionately negative and disadvantage outgroups, meaning that the examination of abstraction as a function of valence could also be capturing a Linguistic Expectancy Bias. It can also be argued that the texts in this study, specifically, better reflect a Linguistic Expectancy Bias, since the design of original study involved a fictional group without reference to the ingroup—presumably leading to epistemic and not ingroup protective motivation. That is, it is believed that when there are stereotypic expectancies available and a lack of context that encourages the promotion or protection of the ingroup (e.g., competition, ingroup threat), then cognitive processing, and ergo the Linguistic Expectancy Bias, will occur (Maass, 1999). Thus, it must be understood that though our methodological approach to automation mirrors that of coding for the Linguistic Intergroup Bias, it may also capture the Linguistic Expectancy Bias.

Historically, the distinction between the Linguistic Intergroup Bias and the Linguistic Expectancy Bias relies on a determination of the processes that underlie their expression. Maass (1999) suggests, however, that the Linguistic Intergroup Bias is likely a function of *both* expectancies and ingroup protection motivations and concludes that the two biases may lead to identical outcomes in language. Ultimately, it is not possible to identify the process responsible for biased expression based on an examination of language alone. That is, in an examination of language alone, especially natural language that is generated outside of manipulated and controlled experimental situations, one cannot determine whether the language was due to motivated or cognitive processing. Further, the distinction between these two biases may not be relevant or conceptually important for researchers who are interested in the reception of language and examine language to infer beliefs: The critical point that language reflects our culture, attitudes, and beliefs remains the same. It may thus be argued that it is more appropriate to refer to the automated detection of “linguistic bias” and thereby avoid the distinction altogether. In the same way, automated coding of Linguistic Intergroup Bias may not be a viable approach for those researchers interested in the specific mechanics, context, and underlying processing involved in the expression or production of linguistic biases.

In sum, this study is the first to present automated approaches to the detection of Linguistic Intergroup Bias. Though all examined automated approaches provide a good approximation of manually coded Linguistic Intergroup Bias in terms of correlations, we recommend an automated approach built on CoreNLP sentiment analysis and LCM Dictionary abstraction coding based on the pattern of means and ease of interpretation. Despite the limitations of this study, it is a first step towards a much-needed methodological advancement within the linguistic bias paradigm. The development of an automated method to detect Linguistic Intergroup Bias would fundamentally change how research is conducted within this paradigm—making research both easier and quicker.

Declaration of Conflicting Interests


The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The authors disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This article draws on research supported by the Social Sciences and Humanities Research Council (grant number: 430-2020-00212).

ORCID iDs

Katherine A. Collins  <https://orcid.org/0000-0001-8833-7203>

Ryan L. Boyd  <https://orcid.org/0000-0002-1876-6050>

Notes

1. <https://mediabiasfactcheck.com>
2. <https://www.allsides.com/unbiased-balanced-news>
3. It should be noted that the inclusion of nouns is by no means standard. Carnaghi et al. (2008) found that, in comparison to adjectives, nouns were perceived to provide more information about the described person, imply a longer duration, and allow for better predictions about the future, in line with the cognitive implications of the Linguistic Category Model (Semin & Fiedler, 1988). However, nouns may be more easily visualized, similar to concrete verbs, so their position as the most abstract level of the LCM may be contested.
4. <https://www.cis.lmu.de/~schmid/tools/TreeTagger/>
5. When nouns are excluded, bias scores range from -4 (*negative bias*) to +4 (*positive bias*).
6. <https://www.butter.tools/>
7. https://github.com/ryanboyd/VADER_Tots
8. <https://github.com/codingupastorm/vaderssharp>
9. <https://stanfordnlp.github.io/CoreNLP/additional.html>
10. We also calculated scores using Brysbaert et al.'s (2014) approach and found a similar pattern of results. This method is not presented as we opted to use the automated methods that are more easily interpretable and comparable to the meaning of the LCM as used within the linguistic bias paradigm, as outlined in our introduction. These methods are also less computationally intensive.
11. <http://sergey-tihon.github.io/Stanford.NLP.NET/samples/POSTagger.html>
12. https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html
13. To increase our confidence in the results, we also transformed our dependent variable of manually coded abstraction using a reflected square root and various reflected logarithmic transformations, following the procedures outlined by Tabachnick and Fidell (2007, pp. 94–95). As determined by a visual examination of quantile-quantile plots, the log2 transformation resulted in an approximately normal distribution. In support of this, the skewness values of the transformed variable were all less than 1. There was also improved homogeneity of variance as shown by the F_{\max} test (F_{\max} ratio = 2.94), but not Levene's Test. Importantly, the mixed analysis of variance on the reflected log2 transformed abstraction scores showed the same significant interaction ($F(1, 73) = 4.117, p = 0.046, \eta_G^2 = .026$). There was also a significant main effect of Content ($F(1, 73) = 4.417, p = 0.039, \eta_G^2 = .028$). The interpretation of the pattern of means was also the same (with no difference between Content for those exposed to the Negatively Biased text and what can be interpreted as an increase in abstraction for desirable content for those exposed to the Positively Biased text). Given that the interaction is the effect of theoretical interest, the identical pattern of results, and that the meaning of abstraction scores based on the LCM are well known and easy to interpret, we will report the analysis of untransformed scores.

References

- Benamara, F., Inkpen, D., & Taboada, M. (2018). Introduction to the special issue on language in social media: Exploiting discourse and other contextual information. *Computational Linguistics*, 44(4), 663–681. https://doi.org/10.1162/coli_a_00333
- Beukeboom, C. J. (2014). Mechanisms of linguistic bias: How words reflect and maintain stereotypic expectancies. In J. P. Forgas, O. Vincze, & J. László (Eds.), *Social cognition and communication* (pp. 313–330). Psychology Press.

- Beukeboom, C. J., & Burgers, C. (2019). How stereotypes are shared through language: A review and introduction of the social categories and stereotypes communication (SCSC) framework. *Review of Communication Research*, 7, 1–37. <https://hdl.handle.net/1871.1/097872c1-d328-4c7f-a1d8-1dd78e0a8632>
- Beukeboom, C. J., Finkenauer, C., & Wigboldus, D. H. J. (2010). The negation bias: When negations signal stereotypic expectancies. *Journal of Personality and Social Psychology*, 99(6), 978–992. <https://doi.org/10.1037/a0020861>
- Blanca, M. J., Alarcón, R., Arnau, J., Bono, R., & Bendayan, R. (2017). Non-normal data: Is ANOVA still a valid option? *Psicothema*, 29(4), 552–557. <https://doi.org/10.7334/psicothema2016.383>
- Bourhis, R., & Maass, A. (2005). Linguistic prejudice and stereotypes. In U. Ammon, N. Dittmar, K. J. Mattheier, & P. Trudgill (Eds.), *Sociolinguistics: An international handbook of the science of language and society* (2nd ed., pp. 1587–1602). Walter De Gruyter.
- Boyd, R. L., & Markowitz, D. M. (2024). Verbal behavior and the future of social science. *American Psychologist*, Advance online publication. 1–23. <https://doi.org/10.1037/amp0001319>
- Boyd, R. L., & Schwartz, H. A. (2021). Natural language analysis and the psychology of verbal behavior: The past, present, and future states of the field. *Journal of Language and Social Psychology*, 40(1), 21–41. <https://doi.org/10.1177/0261927X20967028>
- Brysbaert, M., Warriner, A. B., & Kuperman, V. (2014). Concreteness ratings for 40 thousand generally known English word lemmas. *Behavioral Research Methods*, 46(3), 904–911. <https://doi.org/10.3758/s13428-013-0403-5>
- Burgers, C., & Beukeboom, C. J. (2016). Stereotype transmission and maintenance through interpersonal communication: The irony bias. *Communication Research*, 43(3), 414–441. <https://doi.org/10.1177/0093650214534975>
- Carnaghi, A., Maass, A., Gresta, S., Bianchi, M., Cadinu, M., & Arcuri, L. (2008). Nomina sunt omina: On the inductive potential of nouns and adjectives in person perception. *Journal of Personality and Social Psychology*, 94(5), 839–859. <https://doi.org/10.1037/0022-3514.94.5.839>
- Collins, K. A., Clément, R., & Seih, Y.-T. (2018, June 20–23). *An investigation of the transmission of beliefs using linguistic inquiry and word count*. 16th International Conference on Language and Social Psychology, Edmonton, Alberta, Canada.
- Davidson, T., Warmsley, D., Macy, M. W., & Weber, I. (2017). Automated hate speech detection and the problem of offensive language. *Proceedings of the International AAAI Conference on Web and Social Media*, 11(1), 512–515. <https://doi.org/10.1609/icwsm.v11i1.14955>
- Domonoske, C. (2018). Danish man is first person sentenced under Malaysia's anti-fake-news law. NPR. <https://www.npr.org/sections/thetwo-way/2018/04/30/607068241/danish-man-is-first-person-convicted-under-malaysias-anti-fake-news-law>
- Dragojevic, M., Sink, A., & Mastro, D. (2017). Evidence of linguistic intergroup bias in U.S. Print news coverage of immigration. *Journal of Language and Social Psychology*, 36(4), 462–472. <https://doi.org/10.1177/0261927X16666884>
- Field, A., Park, C. Y., Lin, K. Z., & Tsvetkov, Y. (2022). Controlled analyses of social biases in wikipedia bios. In *Proceedings of the ACM Web Conference 2022*, pp. 2624–2635.
- Franco, F. M., & Maass, A. (1996). Implicit versus explicit strategies of out-group discrimination: The role of intentional control in biased language use and reward allocation. *Journal of Language and Social Psychology*, 15(3), 335–359. <https://doi.org/10.1177/0261927X960153007>
- Gamer, M., Lemon, J., Fellows, I., & Singh, P. (2019). irr: Various coefficients of interrater reliability and agreement. <https://CRAN.R-project.org/package=irr>
- Gignac, G. E., & Szodorai, E. T. (2016). Effect size guidelines for individual differences researchers. *Personality and Individual Differences*, 102, 74–78. <https://doi.org/10.1016/j.paid.2016.06.069>

- Hube, C. (2017). Linguistic models for analyzing and detecting biased language. In: Proceedings of the 26th International Conference on World Wide Web Companion - WWW'17 Companion, pp. 717–721.
- Hutto, C. J., & Gilbert, E. E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. In: Eighth International Conference on Weblogs and Social Media (ICWSM-14), Ann Arbor, MI, June 2014.
- Johnson-Grey, K. M., Boghrati, R., Waksalak, C. J., & Dehghani, M. (2020). Measuring abstract mind-sets through syntax: Automating the linguistic category model. *Social Psychological and Personality Science*, 11(2), 217–225. <https://doi.org/10.1177/1948550619848004>
- Lombrozo, T. (2018). The psychology of fake news. NPR. <https://www.npr.org/sections/13.7/2018/03/27/597263367/the-psychology-of-fake-news>
- Maass, A. (1999). Linguistic intergroup bias: Stereotype perpetuation through language. In *Advances in experimental social psychology* (Vol. 31, pp. 79–121). Elsevier.
- Maass, A., Milesi, A., Zabbini, S., & Stahlberg, D. (1995). Linguistic intergroup bias: Differential expectancies or in-group protection? *Journal of Personality and Social Psychology*, 68, 116–126. <https://doi.org/10.1037/0022-3514.68.1.116>
- Maass, A., Salvi, D., Arcuri, L., & Semin, G. (1989). Language use in intergroup contexts: The linguistic intergroup bias. *Journal of Personality and Social Psychology*, 57(6), 981–993. <https://doi.org/10.1037/0022-3514.57.6.981>
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55–60.
- McHugh, M. L. (2012). Interrater reliability: The kappa statistic. *Biochemia Medica*, 22(3), 276–282. <https://doi.org/10.11613/bm.2012.031>
- Pennebaker, J. W., Booth, R. J., & Francis, M. E. (2007). *Linguistic inquiry and word count: LIWC [Computer software]*. LIWC.net.
- Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., & Choi, Y. (2017). Truth of varying shades: Analyzing language in fake news and political fact-checking. In: Proceedings of the 2017 conference on empirical methods in natural language processing, pp. 2931–2937.
- R Core Team. (2023). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. <https://www.R-project.org/>
- Revelle, W. (2024). *psych: Procedures for psychological, psychometric, and personality research*. R package (Version 2.4.3). Northwestern University. <https://CRAN.R-project.org/package=psych>
- Seih, Y.-T., Beier, S., & Pennebaker, J. W. (2017). Development and examination of the linguistic category model in a computerized text analysis method. *Journal of Language and Social Psychology*, 36(3), 343–355. <https://doi.org/10.1177/0261927X16657855>
- Semin, G. R. (2012). The linguistic category model. In P. A. M. Van Lange, A. W. Kruglanski, & E. Tory Higgins (Eds.), *Handbook of theories of social psychology: Volume 1* (pp. 309–327). Sage Publications Ltd. <http://doi.org/10.4135/9781446249215.n16>
- Semin, G., & Fiedler, K. (1988). The cognitive functions of linguistic categories in describing persons: Social functions and language. *Journal of Personality and Social Psychology*, 54(4), 558–568. <https://doi.org/10.1037/0022-3514.54.4.558>
- Silge, J., & Robinson, D. (2016). tidytext: Text mining and analysis using Tidy Data Principles in R. *Journal of Open Source Software*, 1(3), 1–37. <https://doi.org/10.21105/joss.00037>
- Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 conference on empirical methods in natural language processing, pp. 1631–1642.

- Spinde, T., Rudnitskaia, L., Mitrović, J., Hamborg, F., Granitzer, M., Gipp, B., & Donnay, K. (2021). Automated identification of bias inducing words in news articles using linguistic and context-oriented features. *Information Processing and Management*, 58(3), 102505. <https://doi.org/10.1016/j.ipm.2021.102505>
- Stone, P. J., Dunphy, D. C., Smith, M. S., & Ogilvie, D. M. (1966). *The General Inquirer: A computer approach to content analysis*. M.I.T. Press.
- Tabachnick, B. G., & Fidell, L. S. (2007). *Experimental design using ANOVA*. Thomson.
- Toutanova, K., Klein, D., Manning, C., & Singer, Y. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the 2003 Human Language Technology conference of the North American chapter of the Association for Computational Linguistics (HLT-NAACL 2003), pp. 252–259.
- von Hippel, W., Sekaquaptewa, D., & Vargas, P. (1997). The linguistic intergroup bias as an implicit indicator of prejudice. *Journal of Experimental Social Psychology*, 33(5), 490–509. <https://doi.org/10.1006/jesp.1997.1332>
- Weinerová, J., Szűcs, D., & Ioannidis, J. P. (2022). Published correlational effect sizes in social and developmental psychology. *Royal Society Open Science*, 9(12), 220311. <https://doi.org/10.1098/rsos.220311>
- Wigboldus, D. H. J., Semin, G. R., & Spears, R. (2000). How do we communicate stereotypes? Linguistic bases and inferential consequences. *Journal of Personality and Social Psychology*, 78(1), 5–18. <https://doi.org/10.1037/0022-3514.78.1.5>
- Williams, K., & Nettlefold, J. (2018). Can you tell fact from fiction in the news? Most students can't. The Conversation. <https://theconversation.com/can-you-tell-fact-from-fiction-in-the-news-most-students-cant-102580>
- Yeomans, M. (2021). A concrete example of construct construction in natural language. *Organizational Behavior and Human Decision Processes*, 162, 81–94. <https://doi.org/10.1016/j.obhdp.2020.10.008>
- Ziems, C., & Yang, D. (2021). To protect and to serve? Analyzing entity-centric framing of police violence. *Findings of the Association for Computational Linguistics: EMNLP, 2021*, 957–976. <https://doi.org/10.18653/v1/2021.findings-emnlp.82>

Author Biographies

Katherine A. Collins (Ph.D., University of Ottawa) is an Assistant Professor in the Department of Psychology and Health Studies at the University of Saskatchewan and a member of the Métis Nation of Saskatchewan. In her research, Collins takes a social psychological approach to the study of sociocultural issues, including bias and inequity, with a particular interest on the roles of, and intersections between, language, culture, and identity.

Ryan L. Boyd (Ph.D., The University of Texas at Austin) is a computational social scientist at the University of Texas at Dallas who studies how language reveals and shapes human psychology, from individual personalities to broad social trends. His work covers a range of topics, including mental health, social interaction, cognition, emotion, and behavior. He has published over 100 scholarly papers and is the co-editor of the *Handbook of Language Analysis in Psychology*. Boyd has developed dozens of free and open-source text analysis tools and is one of the lead researchers behind the widely-used Linguistic Inquiry and Word Count (LIWC) software. He serves on editorial and advisory boards for several interdisciplinary journals in the computational social sciences and psychology.