Behavior Research Methods (2019) 51:480–492
https://doi.org/10.3758/s13428-019-01200-w

Measuring the importance of context when modeling language comprehension

Justin Garten[1] · Brendan Kennedy[1] · Kenji Sagae[2] · Morteza Dehghani[3]

Published online: 19 February 2019

Abstract

It is widely accepted that language requires context in order to function as communication between speakers and listeners. As listeners, we make use of background knowledge — about the speaker, about entities and concepts, about previous utterances — in order to infer the speaker's intended meaning. But even if there is consensus that these sources of information are a necessary component of linguistic communication, it is another matter entirely to provide a thorough, quantitative accounting for context's interaction with language. When does context matter? What kinds of context matter in which kinds of domains? The empirical investigation of these questions is inhibited by a number of factors: the challenge of quantifying language, the boundless combinations of domains and types of context to be measured, and the challenge of selecting and applying a given construct to natural language data. In response to these factors, we introduce and demonstrate a methodological framework for testing the importance of contextual information in inferring speaker intentions from text. We apply Long Short-term Memory (LSTM) networks, a standard for representing language in its natural, sequential state, and conduct a set of experiments for predicting the persuasive intentions of speakers in political debates using different combinations of text and background information about the speaker. We show, in our modeling and discussion, that the proposed framework is suitable for empirically evaluating the manner and magnitude of context's relevance for any number of domains and constructs.

Keywords Methodological innovation · Text analysis · Continuous representations · Intent recognition

Introduction

In all forms of linguistic communication, successful interpretation is dependent on our ability to make use of knowledge beyond the words themselves. If we consider the sentence, "I want more", there is little we can conclude about the utterance without knowing the context of the "I" (who is speaking) or the "more" (what is wanted, why it is wanted). This fact is hugely important when attempting to formalize knowledge of the role of outside information on language's functionality. Generally speaking, when we study human language processing we cannot ignore the fact that language is fully embedded within its *context*, the conditions of its generation with respect to participants and environment.

This problem is easily specified, but not easily addressed in quantitative research. There are two significant issues with designing studies to discover these contextual factors. First, there is the issue of measuring the "interpretation" of language, if we are to make conclusions about its relationship with context. There is nothing approaching consensus in what language "means". Indeed, the meaning of language changes from field to field, subfield to subfield, person to person. The second issue is how to measure context. It can come in many forms: cultural and historical background knowledge, access to previous utterances in a dialogue, general knowledge of people, places, and

Justin Garten and Brendan Kennedy contributed equally to this work.

This work has been funded in part by NSF IBSS #1520031.

Electronic supplementary material The online version of this article (https://doi.org/10.3758/s13428-019-01200-w) contains supplementary material, which is available to authorized users.

✉ Morteza Dehghani
mdehghan@usc.edu

[1] Department of Computer Science, University of Southern California, 3620 S. McClintock Ave, Los Angeles, CA 90089-1061, USA

[2] Department of Linguistics, University of California, Davis, CA 95616, USA

[3] Department of Psychology and Computer Science, University of Southern California, Los Angeles, CA 90089, USA

events, or knowledge about the status, characteristics, or motivations of the speaker. Furthermore, it is evident from the literature that context does not have an unvarying effect on language understanding: across domains such as humor (Fine, 1983), sarcasm (Capelli et al., 1990), political persuasion (Cohen, 2003), and others, there are ostensibly different contextual sources of variance in the correct interpretation of such utterances. Thus, what one may find in a single domain might not hold for another.

Together, these issues make the goal of quantifying context in language understanding intractable. Therefore, we advocate for an *incremental* approach to gathering empirical evidence as to the effects of context. Clearly, we will not be able to make universal claims about meaning and context; however, if we adopt a methodological framework that allows the researcher to specify the "language understanding" task and the contextual items to be used, then we will be able to make specific claims about *when* context matters, *how* its effect translates across domains and language understanding tasks, and what *kinds* of context are more or less salient.

In this paper, we propose a methodological framework capable of facilitating such a research agenda. In data-driven studies of human language understanding, we aim to provide a tool for researchers by which they can provide evidence for or against a claim about context. This tool, which leverages recent advances in Natural Language Processing (NLP) for generating meaningful representations of text for modeling purposes, effectively simulates the language understanding task through computational means. It tests the level to which an algorithm performs in classifying some meaningful aspect of an utterance, and compares the success of models with and without access to contextual variables. In previous research, the only alternative is to conduct experiments where participants are provided with varying levels of background knowledge and asked to make conclusions about items, a technique sometimes used in political psychology (e.g. Chong & Druckman, 2004; Sniderman & Theriault, 2007). While arguably more valid, these studies are more expensive and harder to translate to findings in other studies and fields.

In this paper, we focus on two tasks: first, we introduce and justify the methods we use in our framework, which are established and used in a variety of NLP tasks. Then, in a demonstration of the use of this framework, we conduct a study of political speech acts and how their efficacy is enabled, in part, by background knowledge about the speaker. In reference to our discussion criteria above, the "language understanding task" is inferring the intentions of the speaker or, in other words, inferring the action accomplished by the language (i.e. "Speech Act Theory": Austin, 1975; Searle, 1969). In this, we assume the intentional action of political language to be critical in a listener's understanding, or interpretation, of the utterance. The contextual items we consider are the debate speaker's partisan affiliation (Democrat or Republican) and incumbency status (incumbent or challenger).

The challenge of quantifying text

The methodology we propose is rendered necessary by how difficult it is to quantify language. Without this challenge, the task of modeling context's impact on language is trivial through traditional statistical methods. But the question of how to map text to some mathematically accessible form — in a manner which preserves meaning — is one which has occupied researchers for decades. In the domain of psychology, the accepted standard for encoding text into numeric features is to calculate the relative frequencies of sets of words — given by domain dictionaries, which are expert-generated lists of words characteristic of a particular construct — per document (Pennebaker et al., 2001). These techniques have made new questions about the psychological underpinnings of language answerable; however, they are severely limited by aspects of language which are missed by using the frequency of pre-selected words. Firstly, the use of concept dictionaries will at best capture pre-selected slices of semantic and pragmatic intent. Because these words are selected by experts before interacting with real-world data, it is possible that they could miss certain components of the construct they purport to measure (false negatives), or fail to account for a word's multiple meanings (false positives). In certain respects, this has been addressed by using semantic similarity (Garten et al., 2017), which computes distributed representations of concepts words and compares these with the representations of text.

But even having accounted for the meaning of words by themselves, there is a second, more fundamental problem with these text analysis methods: if language is represented as a set of word frequencies (dubbed 'bag-of-words' in the NLP literature), we are then limited to testing hypotheses which operationalize meaning as a set of word frequencies. The obvious problem with this is that language is sequential; thus, if we are trying to simulate a human-level understanding of text, we fall short. What is not obvious, however, is how to solve this problem.

In terms of expressing the general sequential aspect of language, methods in NLP have been developed for the general task of "language modeling", which is the prediction of the "next" word, given the words that come before it. In the last decade, methods based on Recurrent Neural Networks (RNNs), a particular class of deep learning methods, have become, far and away, the best performing models for this and other critical NLP tasks (Jozefowicz et al., 2016). As with other deep learning-based advancements, this has come about through a combination of faster computers, huge training corpora, and improved neural architectures. In a big picture sense, this transition means that representations of text that go beyond bag-of-words approaches are more feasible, due to the expressive power of neural network architectures.

Two general areas of NLP research are of interest to our particular problem of providing general representations of language: word embedding and sentence embedding. In word embedding, the object is to learn a low (300 or less) dimension representation of every word in a vocabulary, such that similarly functioning words are close together, and vice versa with dissimilar words. In general, current approaches to word embedding (word2vec and GloVe; Mikolov et al., 2013a; Pennington et al., 2014, respectively) learn these word-level vector spaces based on the words they frequently co-occur with. These word embedding results, which are trained on huge, neutral corpora, provide rich semantic information and are frequently used in NLP applications.

A more difficult, and altogether more critical, area of neural NLP research is the construction of sentence embeddings, or the projection of sequences of words into a low-dimensional space that, similar to word embeddings, preserves meaning similarities and differences between text segments. Given the quantum leap from word to sentence, in terms of possible meanings, this second task is hugely challenging. Nevertheless, research in recent years has yielded impressive results. Established approaches in this area include Kiros et al. (2015), which learns generic, unsupervised sentence embeddings by learning to predict the *sentences* that frequently co-occur with a given sentence, and Palangi et al. (2016), which uses Long Short-term Memory (LSTM) networks, a variant of RNNs, to learn sentence-level representations in a supervised setting. In our method, we similarly use an LSTM-based encoder for representing text in continuous space. The details of these methods, and in particular the methodology we adopt, are discussed in greater detail in our Methods section.

Incorporating contextual variables

Lastly, given that we have designed a flexible framework for predicting a given aspect of language understanding using advanced representations of language, we consider the problem of introducing competing levels of types of contextual information. In this area, there is some precedence in NLP research. For example, representations learned with demographic information have been shown to improve classification performance on a range of tasks (Hovy, 2015; Johannsen et al., 2015; Garten et al., 2018), particularly in venues such as social media (Hovy & Søgaard, 2015) where group-based linguistic variations can be extreme. In particular, word representations learned on demographically split data (Bamman et al., 2014; Garimella et al., 2017) have proved useful for community-specific classification in areas such as sentiment analysis (Yang & Eisenstein, 2015).

However, these studies have relied on the availability of sufficient data in order to train domain-specific representations, often on the order of millions of documents. Here we consider situations much more common in the behavioral sciences, wherein we lack enough data to train domain-specific representations but have prior theoretical or experimental reasons to explore particular contextual factors. Given this added constraint, the designed methodology must therefore be flexible with respect to smaller datasets, such that significant differences in the impact of contextual variables are detectable.

For this purpose, we provide two approaches: (1) Split-sample modeling, where the predictive model is trained on data from one class and tested on the other; (2) Directly incorporating categorical inputs into the network, which is robust to multiple contextual variables.

Considered together, these methods form a predictive framework which allows context-based questions to be asked about language processing through model comparison. In our Methods Section, we discuss the contributing aspects of this methodological framework in detail.

Applying the methodology

We compare the relative merits and constraints of these approaches on the task of modeling speaker intent in political debate. We select speaker intent, an important aspect of language understanding, given that prior work has shown this domain to highlight the interaction between message, speaker, and audience (Grice, 1975; Goodwin, 1984). For example, strong partisans are more likely to accept or reject a claim based on whether the source shares their affiliation (Cohen, 2003; Jost et al., 2009). These tensions are highlighted in debates where the goal is explicitly to persuade listeners, but the target varies widely as politicians simultaneously attempt to attract undecided voters, motivate their own supporters, and demotivate the other side's voters (Jerit, 2004). Emotional arguments (Macagno & Walton, 2014) can be difficult to parse without extensive prior knowledge and some statements are even deliberately obscured so that only groups who already agree with a stance will even realize that a position is being taken (López, 2015). As such, contextual clues are essential for interpreting the intent of a given utterance.

To explore these questions, we annotated US general election presidential debates from the years 1960-2016 on five factors: attack, policy, value, group, and personal statements (for examples see Table 1). The result is a dataset which combines a difficult baseline task (intent recognition) with a situation where prior work suggests that background information about the identity of the speaker should be important.

We use this setting to demonstrate the use of the proposed framework, which uses LSTM networks to jointly learn mappings of the text of debate turns (one continuous speech by the speaker) to sets of intent labels. To evaluate the impact of contextual factors on intent in political speeches, we then incorporate information about speakers — their political party and incumbency status — as inputs into the model, and compare these predictions with those generated from text alone. In doing so, we provide evidence that several forms of intentional speaking in political debate require contextual information in order to function, suggesting that more thorough analysis, through the introduction of more contextual variables, will yield further results in this area.

Based on our results with one domain, we posit that our presented methods can be used generally for text-based analyses of context and language understanding. While there are shortcomings to our approach — the lack of interpretability of the internal states of the LSTM network and the high cost of additional annotation — this framework for investigating the informational requirements of basic language processing has bearing for many questions in the social sciences, particularly those for which the language dynamic is a dialogue between speakers and listeners.

Methods

In this section we detail the methodological components of our proposed framework for prediction-based analyses of contextual factors in text. This framework stitches together four distinct elements of established research in NLP and machine learning: word embeddings, which represent the semantic meaning of words in geometric space, reflecting similarity, analogy relations, and word-clustering (Mikolov et al., 2013a; Pennington et al., 2014); Long Short-term Memory Networks (LSTMs), a variant of Recurrent Neural Networks (RNNs), used here for sequence classification; Multi-task learning (Caruna, 1993), specifically regarding NLP multi-task learning (Collobert & Weston, 2008) for learning representations for multiple labels at the same time; and two approaches for comparing effects of contextual variables: an LSTM architecture which allows for the inclusion of multiple categorical variables as inputs, and a cross-validation strategy for single categorical variables. In order to clearly explain the technical aspects of the framework with regards to the modeling objectives discussed above, we provide substantial detail on each of these concepts, which jointly serves the purpose of introducing advanced neural methods to the readership in general. We also include a description of the baseline methods we use for validation of model performance in our experiments.

Text representation

In this section we specify the architecture and experimental approach used in the framework for text representation. Generally, we adopt standards in NLP which use recurrent neural networks (RNNs), initialized with word embeddings, for the vectorization of texts. RNNs have been used with great success on a variety of sequence modeling tasks, including optical character recognition (Graves & Schmidhuber, 2009), machine translation (Sutskever et al., 2014), and language modeling (Jozefowicz et al., 2016). An RNN is essentially a memory cell that, as it reads in inputs, simultaneously incorporates information from the *previous* input as well. Thus, the state of the memory cell reflects the information of the sequence, up to the input that has been processed. In this way, the network can be said to be learning "temporal" relationships, or how information in a sequence changes over time (Elman, 1990).

Thus, the intuition behind using RNNs for processing sequences of words is that the architecture learns an "evolving latent representation" of the words, as they are inputted in, one at a time. This evolving, "hidden" representation then is optimized according to a prediction task. For language modeling, the task is to predict the next word in the sequence; for classification task, the object is to produce a final representation (after all the words of the sequence have been fed into the model) that accurately predicts the class membership of the given piece of text.

For our context-driven framework for text analysis, the particular architecture we make use of relies on Long Short-Term Memory (LSTM) units (Hochreiter & Schmidhuber, 1997). These have been found to better capture the long-ranging dependencies often observed in linguistic information, especially over multiple sentence spans (Sundermeyer et al., 2012; Ji et al., 2016).

The final output of this network is an h-dimensional latent representation, where h is the length of the vector progressively learned through encoding the sentence. This text representation is then passed to a standard feed-forward neural network (Svozil et al., 1997) consisting of a d-node dense layer and final binary output layer (Fig. 1). All neural network models made use of rectified linear units (Nair & Hinton, 2010) as non-linear activation functions, were optimized using the Adam algorithm (Kingma & Ba, 2014), and trained over 30 epochs. All of these choices are standard practice in training such neural networks; we include these specifics for replicability.

The selection of the size of the dense layer is largely a factor of engineering; enough expressiveness (more units)

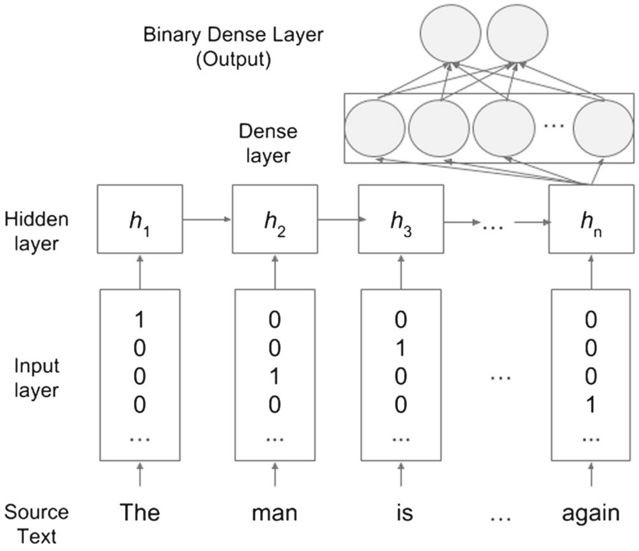

Fig. 1 Visualization of the standard architecture for LSTM text classification, with the input layer initialized to one-hot encodings (made-up values here). The input sentence is from our examples: "The man is practicing fuzzy math again"

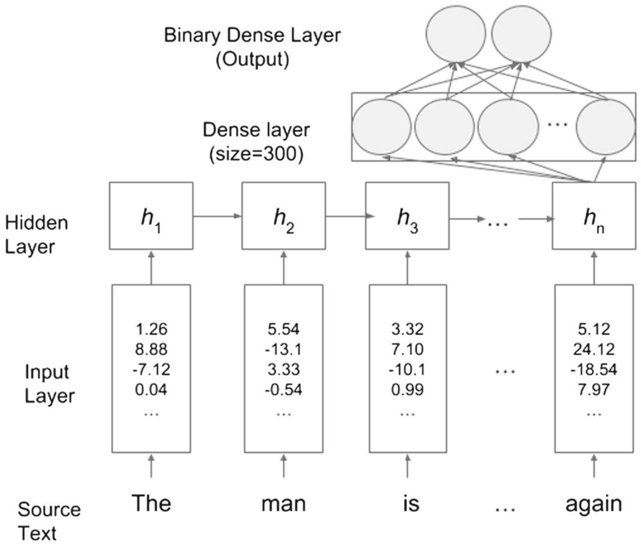

Fig. 2 Visualization of the standard architecture for LSTM text classification, with the input layer initialized with pre-trained word embeddings (made-up values here)

has to be provided, but not too much as this will 'overfit' the data (account for too much of the variance of the training data, thus fit poorly to the test data). For our experiments we use size 100, which was selected through experimentation. The last layer (2 units) is the output layer, which models a probability distribution over the two possible outputs (for binary classification).

In order to understand a key aspect of our method, word embeddings, it is necessary to understand the mechanics of how words are "fed in" to the model. The source words are first encoded into a continuous representation. This representation can be one of two things: first, it can be a simple indicator function, by telling the receiving memory cell the identity of the word being read (a "one-hot" encoding, with a 1 at the index of the given word and 0 elsewhere, seen in Fig.1); the second option is to initialize each word not as an indicator, but a pre-trained representation of that word in continuous space (Fig. 2).

This latter method, which we use in our framework, is a key element for adequately representing text in continuous space. These pre-trained representations are such that the vector corresponding to a given word, for example "mental", occupies a similar space (close by Euclidean distance) as the vector for "physical". The two words are not synonyms, but they often occur within the same context words, i.e. they are *used* similarly across large datasets. The object of using such embeddings is that, by knowing how words are used in general, they network has more information on the similarity and difference of different words than a simple one-hot encoding might offer. Additionally, even if we were to learn word representations with our dataset (which is possible), this representation would reflect on the context of our particular data, meaning that there would be no universal linguistic content in the word representations, only local.

For our framework, we use word embeddings pre-trained using the Word2Vec model (Mikolov et al., 2013b) on the the full text of the English Wikipedia.[1] By choosing this corpus (as opposed to, say, Google News[2]) the thinking is that the language contains in wikipedia text in discussing politics and historical events of previous time periods would have good coverage of the language used in our own corpus.

In evaluation of the effectiveness of LSTMs for representing language, it is useful to compare to a bag-of-words baseline method. As this baseline, we compare results for our experiments with LSTMs to (linear) Support Vector Machine (SVM) classifiers (Suykens & Vandewalle, 1999). These classifiers are state-of-the-art for a large amount of tasks, and operate by discovering a relationship between two classes by finding the dividing line which creates the "maximum" margin between the data points. As features for this classifier, we use the established method for document representation: term frequency-inverse document frequency ($tf - idf$), which first counts the frequency of unique words per document then normalizes each by the length of the document, and then multiplies by the inverse document frequency of the term across the corpus (Aizawa, 2003). This is used to down-weight words that frequently occur in a corpus and up-weight infrequent words (across the corpus) for the document in which it occurs. In general,

[1] http://wikipedia.org/

[2] https://github.com/mmihaltz/word2vec-GoogleNews-vectors/

such comparisons with a simple baseline clarifies how much "signal" can be learned just from the frequency of words, and how much *more* we can capture using an advanced method like LSTM.

Multi-task networks

When conducting an experiment, it is possible that the annotations for a text corpus will have more than one label. For example, in our demonstration of the pipeline, we have four labels corresponding to the different types of intent observed in political speech. In this case, the complicated, correlated relationship between these dimensions can be used to enforce our sentence representations to jointly model these dimensions, as opposed to training networks separately for each. This line of thinking with regard to multi-task learning is established as a way to prevent overfitting, as well as to learn difficult tasks when those tasks share some semantic space (Caruna, 1993).

We make use of the same recurrent network described above for generating text representations. However, rather than having that representation passed to a single prediction network, it passes to separate prediction networks (one for each of the labels) as shown in Fig. 3. This keeps the predictions separate while allowing information from each of the categories to pass back through the network and jointly influence the structure of the representations.

To compare the effectiveness of the multi-task network with the networks trained individually, we also introduce another possibility with handling the multiple labels. This approach consists of modifying the standard softmax (Priddy & Keller, 2005) approach to multi-class classification. The softmax works by normalizing network output values associated with multiple classes to yield a distribution over those classes. We begin with the same network structure however, rather than taking the maximum value as a fixed prediction, we instead apply a cutoff value where any labels over the cutoff are treated as positive predictions.

Techniques for testing contextual relationships

The main goal of our framework is to measure the effects of contextual variables on language understanding. In the LSTM-based methods above, we establish baselines based on the best possible representation of language, given the state-of-the-art in textual representation from the NLP literature. In this next step, we outline two techniques for testing the relationship between context and text for a given, measured construct.

The first approach requires no architectural modifications, but does not scale up to testing multiple contextual variables at once. The technique we propose is a "leave-one-class-out" cross-validation. That is, for class C, we train a given multi-task LSTM model on all samples where the class is not equal to C and test on those with class equal to C. By doing this, we can discover how "different" the relationship between label and context is, if there are marked differences in performance when the model is trained on category and tested on the other.

The second approach is to modify the structure of the network to incorporate categorical variables in addition to the continuous representation of the text that we generated using the LSTM. Each variable is encoded as a one-hot vector (a dummy variable), such that adding k contextual variables increases the number of features to $h + 2k$, where h is the size of the representation generated by the neural network. This is pictured in Fig. 4. The benefits of this method are clear: for as many categorical inputs as we want to add to the model, we can simply observe any changes in predictive performance. In addition, this simple approach, which is well-established as a method in machine learning for combining categorical and continuous variables, is flexible enough to be applied to any neural model, and not just LSTM networks.

For all our models, we plan to make our code available. Additionally, we refer the reader to freely available, accessible tutorials in creating multi-task networks in

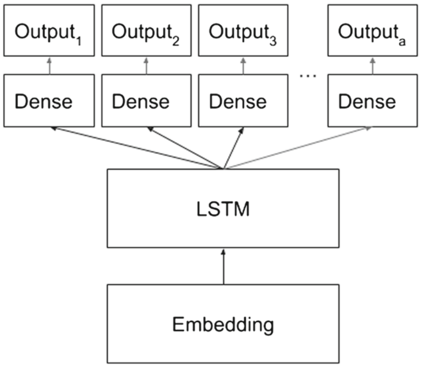

Fig. 3 The multi-task network structure

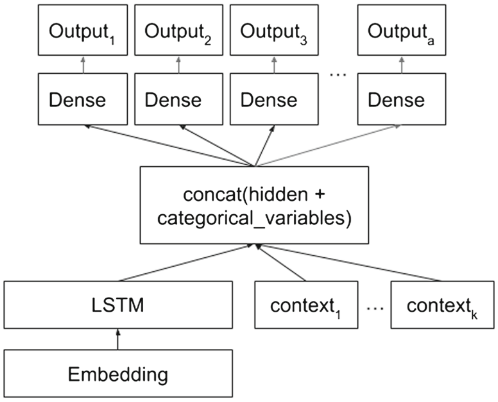

Fig. 4 The multi-task model with contextual variables added as inputs

'Keras', a programming language for developing neural architectures.[3]

Dataset: political debates

Prior political science research (Mendelberg, 2002) suggests that political debate does not follow ideal deliberative structure, which is defined in terms of "its ability to foster the egalitarian, reciprocal, reasonable and open-minded exchange of language" (p. 153). As such, we aimed to explore political debate in terms of persuasive argumentation, including argument types which are conventionally dismissed as fallacies (Macagno & Walton, 2014). We specifically tailor our conception of persuasion to the political arena, where speech as action is directed at a particular subset of voters. We focused on two types of arguments which are increasingly considered central to political choices and highlight the interaction between linguistic and background context in terms of language understanding: values and group identities.

The first axis we considered was the gap between personal and group identity claims. While traditionally candidates were thought of as "selling" themselves to an evaluative electorate, recent work suggests that far more decision making occurs in terms of group identity (Dawes et al., 1990; Carpini et al., 2004). Politicians either suggest that they are a part of a favored group or that their opponents are part of a disfavored one (Swanson & Mancini, 1996). In US electoral politics, claiming that someone is part of the "Washington elite" or doesn't understand "real Americans" has fallen beneath the level of cliche. The centrality of group alignment in political discourse has become particularly salient in light of recent elections around the world where racial, economic, and national identities have played critical roles. Background knowledge about the speaker is inherent to understanding when a group identity is being referenced, making this a prime dimension of political persuasion to examine.

The other axis we considered was the gap between discussions of policies and the values that underpin those policies. This division underscores many theories of rational voting wherein individuals may lack the expertise to evaluate specific policies but can more easily determine a preferred value orientation (Richardson, 2002; Christiano, 1995). The choice to focus on policy details or broad values provides important cues to how a politician is framing a given debate (Iyengar, 2005). Given that recent work suggests that political groups can fail to understand one another when discussing values (Graham et al., 2009), incorporating context provides valuable information to identifying these arguments. Knowledge of the political affiliation of a given speaker provides valuable clues as to how value arguments are being deployed and allows for far finer interpretation of ambiguous statements.

In order to evaluate these axes, US general election presidential debates from the years 1960-2016 were annotated by seven undergraduate students, each trained in the coding scheme and given an initial trial annotation. Each debate was assigned to at least two annotators to allow for agreement evaluation, and each "turn", or uninterrupted speech by a single speaker, in the debate was annotated individually. The final set covers 18 debates with a total of 43 annotations yielding paired annotations for 4670 debate turns with 29395 sentences and 292k words.

Based on our discussion of values and group identity above, utterances were annotated based on the presence or absence of five discourse strategies commonly used in debates: attacking one's opponent, description of policy, statement of values, group identity claim, and a discussion of personal characteristics. A single utterance could be tagged with multiple labels and examples of all label combinations were observed. This allows us to consider the two dimensions described above while comparing whether statements on these dimensions are used to support a candidate or attack their opponent. An utterance could be left unlabeled if none of the categories were present. Examples of each of the categories is provided in Table 1 and additional detail on each of the features (the descriptions given to the coders) is included in the Supplemental Materials.

Argument structures are generally difficult to annotate (Musi et al., 2016) and this is particularly true with these types of persuasive arguments. This task combines the difficulty of highly subjective tasks such as sarcasm detection (Bamman & Smith, 2015) with the unique requirements of historical text annotation. Annotators are asked to label arguments from periods with which they may not be familiar. One of our motivating considerations was that, due to the nature of these arguments, different annotators would pick up on different facets of a statement. Due to the difficulty of annotation and the resultant low levels of agreement, we hypothesized that we would be losing too much information by considering as 'positive' only those instances with full agreement. Given the subjectivity of labeling political speech acts, if at least one annotator labeled the debate turn as positive, we view this as neither a negative nor a positive instance, in that it could be argued that it was both positive and negative. As such, we report classification results on two different sets of labels, those created by the intersection of the two annotations (conventional inter-annotator agreement) and the union. These counts are reported in Table 2.

[3] https://keras.io/getting-started/functional-api-guide/#multi-input-and-multi-output-models

Table 1 Examples from each of the annotation categories

Annotation	Example
Attack	"The man is practicing fuzzy math again." (Bush, 2000)
Policy	"I'm going to invest in homeland security, and I'm going to make sure we're not cutting COPS programs..." (Kerry, 2004)
Value	"What is more moral than peace?" (Ford, 1976)
In(out)-group	"I fully recognize I'm not of Washington." (Bush, 2000)
Personal	"My wife Tipper, who is here, actually went on a military plane with General Sholicatchvieli..." (Gore, 2000)

This yielded paired annotations for a total of 4670 turns with 1453 of those from the candidates (as opposed to moderators or audience questions). The total counts for each label are reported in Table 2.

The kappas range from 0.43 and 0.64, generally considered to signify "moderate" to "significant" agreement (Viera & Garrett, 2005). Given the inherent subjectivity of the task, this is well within the expected range. Due to significantly lower agreement on the in-group category (kappa = 0.21), we excluded it from further analysis.

Experiments

Given our dataset of political debate turns, each annotated along four intent-based dimensions, we illustrate the use of a prediction-based framework for the study of context in language. In study 1, we evaluate the relationship between text and speaker intentions, absent background contextual knowledge. We compare the LSTM-based text classification method to a bag-of-words baseline, showing that meaningful gains in performance are achieved by using the neural architecture. In addition, we compare the bag-of-words and LSTM classification with the multi-task LSTM network, showing where jointly predicting the labels can outperform predicting them individually. In study 2, we demonstrate the first approach for isolating the effects of context: a split-sample strategy that trains and tests on different subsets of the data, separated by a given contextual variable. In study 3, we apply a modified LSTM architecture which directly integrates categorical variables. In each study, we discuss both the results and the general methodological considerations one must make when implementing these methods for other data.

Table 2 Turn-level agreement scores and counts for intersection and union data

Category	Kappa	Intersection	Union
Attack	0.49	822	1154
Policy	0.64	733	984
Value	0.46	450	840
Ingroup	0.21	203	697
Personal	0.43	504	922

We report Cohen's kappa (between ground truth and predicted values) across the test folds (these are discussed in each Study), as well as the standard error. For each of our experiments, we train and present results on two datasets: the intersection and union data, which are both discussed in Section 3.

Study 2: split-sample modeling

Having verified two major methodological aspects of the proposed framework — using LSTM sentence encoding over traditional word-level classifiers, and using a multi-task framework versus modeling each factor separately — we move onto the main contribution: integrating contextual variables. The object of the prior steps in the analysis is to obtain the best possible representation, absent contextual information.

For any contextual analysis, the variables for study ought to be selected based on the given domain, the theoretical questions about linguistic functionality posed by the experimenter, and the availability of data. For our demonstration, we consider two contextual factors: political party affiliation and incumbency status. Differences in the issue profiles and rhetorical strategies of the two primary US political parties[4] have been extensively studied (Heritage & Greatbatch, 1986).

We introduce two techniques for studying the effects of context: adopting a split-sample strategy, and modifying the architecture of the LSTM network to directly add contextual variables into the prediction.

For the split-sample approach, we use the multitask network from Study 1 with the previously described settings. Here, we split the available data by party, training separate models on either Republican or Democratic data. We evaluate against both the same party as the data is trained on and the other party. For the same party case, we evaluate based on leave one out cross validation as in the previous

[4] Although the period covered included two independent candidates (Anderson in 1980 and Perot in 1994), due to the small amount of data we focus on the two primary US political parties.

Table 3 Study 2: Split-sample approach for political party (intersection)

Train	All	Republican		Democrat	
Test	All	Rep	Dem	Rep	Dem
Attack	0.68 (0.05)	0.45 (0.08)	0.12 (0.03)	0.26 (0.03)	0.49 (0.07)
Policy	0.71 (0.03)	0.54 (0.06)	0.12 (0.05)	0.24 (0.04)	0.49 (0.05)
Value	0.54 (0.05)	0.37 (0.07)	0.09 (0.03)	0.06 (0.03)	0.41 (0.04)
Personal	0.6 (0.05)	0.43 (0.06)	0.11 (0.03)	0.03 (0.02)	0.56 (0.08)

Average kappa (SE) for the multitask model trained on Republican data and tested on each Democratic debate pair (and vice-versa)

experiment. When testing on the alternate party, we train on the complete available dataset and average test results over the available debates.

The intersection and union experimental results are seen in Tables 3 and 4, respectively. While we expected the intra-party models to do well, we anticipated that they would be hurt by the fact that they had access to less than half the training data of the complete model. In general, we found this to be the case; however, the magnitude of this difference varied with the label being considered. In Table 3, we observe that, for the intersection dataset, "Personal" has a large discrepancy between training/testing on the full data (0.60 + −0.05) and training/testing on Republican/Republican (0.43 + −0.06), but not nearly as large of a difference as for Democrat/Democrat (0.56 + −0.08). Our interpretation of this is that Republican personal referencing was more specialized than Democrat. But, when considering the union data results (Table 4), the dynamic is flipped, suggesting that Republican rhetoric is more ambiguous and difficult to classify than Democrat rhetoric.

In analyzing the differences between split-sample performance for political party, we see that all four dimensions display significant differences, based on which data they were trained and tested on. For the intersection data (Table 3), differences for models trained on Democrat data are either moderately large ("Attack", 0.26 vs. 0.49; "Policy", 0.24 vs. 0.49), or worlds apart ("Value", 0.06 vs. 0.41;"Personal", 0.03 vs. 0.56). Our interpretation of the complete inability for Democrat Value/Personal rhetoric to generalize to Republican rhetoric is that attacks and policy discussions look roughly the same linguistically, while value arguments and personal references are highly context dependent.

Again, when we examine the results for the union dataset, we see that this dynamic is not as prevalent. In fact, for "Attack" we see even less discrepancy in the Democrat-trained data (0.21 vs. 0.36), but greater discrepancy for "Policy" (0.35 vs. 0.64). This could be attributed to one of several explanations: either the intersection data makes the considered categories too sparse to reliably predict, or Democrats and Republicans differ in the level of nuance and diversity in their rhetoric speech.

Next, we follow the same method described above but apply it to incumbents and challengers. The choice of incumbency was a result of both prior work and our qualitative analysis where we observed that challengers seemed to attack more than incumbents. As such, we hoped to test both general differences and were particularly interested in whether the attack category would be better represented by models split along this axis. For each year, the candidate from the party in power is treated as the incumbent. In the included years, this was either the current president or vice president.

As seen in Table 5 for the intersection data and Table 6 for the union data, the results show again that there are significant differences in how text interacts with each dimension of speaker intent, for incumbents and challengers. In these results, we see almost the same pattern of differences as with party information. In particular, the model trained on challenger data is incapable of predicting "Value" and "Personal" labels for the incumbents data. We interpret to mean that incumbency and party information contains roughly the same information, as far as a predictive model is concerned.

Table 4 Study 2: Split-sample approach for political party (union)

Train	All	Republican		Democrat	
Test	All	Rep	Dem	Rep	Dem
Attack	0.72 (0.03)	0.43 (0.06)	0.15 (0.04)	0.21 (0.03)	0.36 (0.07)
Policy	0.77 (0.02)	0.58 (0.05)	0.34 (0.06)	0.35 (0.06)	0.64 (0.06)
Value	0.63 (0.03)	0.52 (0.05)	0.27 (0.05)	0.27 (0.05)	0.55 (0.04)
Personal	0.66 (0.03)	0.61 (0.05)	0.17 (0.05)	0.19 (0.04)	0.52 (0.03)

Average kappa (SE) for the multitask model trained and tested as in Table 3

Table 5 Study 2: Split-sample approach for incumbency status (intersection)

Train	All	Incumbent		Challenger	
Test	All	Inc	Cha	Inc	Cha
Attack	0.68 (0.05)	0.62 (0.07)	0.14 (0.03)	0.25 (0.04)	0.5 (0.06)
Policy	0.71 (0.03)	0.54 (0.05)	0.15 (0.05)	0.35 (0.03)	0.47 (0.05)
Value	0.54 (0.05)	0.52 (0.07)	0.06 (0.04)	0.04 (0.03)	0.39 (0.05)
Personal	0.6 (0.05)	0.53 (0.06)	0.08 (0.03)	0.01 (0.05)	0.6 (0.06)

Average kappa (SE) for the multitask model trained on incumbent data and tested on each challenger debate pair (and vice-versa)

Table 6 Study 2: Split-sample approach for incumbency status (union)

Train	All	Incumbent		Challenger	
Test	All	Inc	Cha	Inc	Cha
Attack	0.72 (0.03)	0.46 (0.06)	0.1 (0.03)	0.17 (0.03)	0.37 (0.07)
Policy	0.77 (0.02)	0.6 (0.06)	0.32 (0.06)	0.33 (0.07)	0.63 (0.05)
Value	0.63 (0.03)	0.55 (0.04)	0.26 (0.05)	0.27 (0.05)	0.5 (0.05)
Personal	0.66 (0.03)	0.63 (0.05)	0.18 (0.04)	0.23 (0.05)	0.48 (0.04)

Average kappa (SE) for the multitask model trained and tested as in 5

There are additional explanations of this dynamic, however. One possible explanation is that democrats were over-represented in the challenger set and, as seen in with the split-sample party study, their attacks proved far easier to classify. This imbalance is a characteristic of our data, and could be addressed with further, more detailed analyses.

Study 3: directly incorporating contextual variables

In Study 2, we found that for both party affiliation and incumbency status, the interaction between and text and speaker intent varied highly across contextual divides. These results fit with prior work and strongly suggested that useful information is contained in these categories. The next question in this line of experimentation was how these divisions would interact. For example, do Republican challengers behave differently from Democratic challengers? The separate train/test procedure in the previous study is not suitable for such a task, given that the data becomes too sparse. Given this, we next evaluate the potential of incorporating these contextual factors directly into a single model.

Using the multi-task network specified in the Methods section, we compare models with contextual variables added into the structure of the model. We train the network to have a hidden vector of size 300, and compare models with incumbency added, partisanship added, and both added. We evaluated models in the same manner as in Study 1, training on 17 debate pairings and testing on the held-out pairing.

Table 7 Study 3: Integrated model (intersection)

	No Context	+Party	+Incumbency	+Both
Attack	0.68 (0.05)	0.70 (0.05)	0.69 (0.05)	0.70 (0.04)
Policy	0.71 (0.03)	0.70 (0.02)	0.72 (0.02)	0.71 (0.02)
Value	0.54 (0.05)	0.53 (0.04)	0.52 (0.04)	0.51 (0.05)
Personal	0.60 (0.05)	0.62 (0.05)	0.64 (0.04)	0.61 (0.04)

Averaged Kappa for the baseline multitask model, the model with party information added, incumbency information added, and both added

Table 8 Study 3: Integrated model (union)

	No Context	+Party	+Incumbency	+Both
Attack	0.72 (0.03)	0.78 (0.03)	0.79 (0.03)	0.79 (0.03)
Policy	0.77 (0.02)	0.87 (0.02)	0.87 (0.01)	0.88 (0.01)
Value	0.63 (0.03)	0.72 (0.03)	0.73 (0.03)	0.74 (0.03)
Personal	0.66 (0.03)	0.71 (0.02)	0.73 (0.02)	0.72 (0.02)

Averaged Kappa for the baseline multitask model, the model with party information added, incumbency information added, and both added

The results for experiments for intersection and union data shown in Tables 7 and 8, respectively. There are no significant improvements with the intersection data, but significant improvements of varying effect size for the union data. This is perhaps the most convincing case we have for the following conclusion: when measuring the importance of context for political persuasion, context matters more for those instances which are more ambiguous, or less definitive.

Another question we hoped to answer with this information was whether or not the contextual factors interacted. From the results we report, this question remains unanswered, as the model with both factors included doesn't significantly outperform the individually context-augmented models. Conceptually, one would expect that adding party information results in improvements on a certain type of sample, while incumbency adds information on a different set. We do not conduct such post-hoc, investigative analyses here, though this is an important aspect of analysis and one that we plan to conduct in later studies.

In our view, this combined model is a natural complement to the split-sample analyses conducted in Study 2. In the split-sample analyses, we were able to see the direction and magnitude of differences between a binary contextual variable, showing us how differences in political rhetoric (between party or incumbency) vary between dimensions of speaker intent. In the combined model, we were able to compare results more directly between the intersection and union data, and we were able to detect whether or not contextual information from multiple variables interacted. Though we found this not to be the case, for any context-based analysis such results are critical.

Conclusion

Determining the informational requirements of human language processing can be done by simulating the language understanding task, using advanced representations of text alongside contextual variables. In this paper, we demonstrate the effectiveness of this type of research, examining the role that basic background information of a speaker in comprehending political discourse. The soundness of our methodology with respect to modeling various aspects of language, largely drawn from the NLP literature, provides validation for the findings of future research endeavors which use our framework.

The validity of results from any application of this framework are supported by a number of technical considerations: (1) The use of an LSTM architecture with pre-trained word embeddings, which markedly improved prediction scores over a standard baseline method (SVM with tf-idf features), improves our certainty that contextual factors are (or are not) influencing the comprehension task; (2) By using a multi-task learning architecture, we account for experimental settings in which a construct falls along multiple dimensions, thus more accurately capturing the relationship between construct and language; and (3) We introduce and demonstrate two approaches for experimenting with the effects of context on language processing, which are both clear to interpret and parsimonious in terms of model construction.

In our demonstration of our methodological framework, our experiments demonstrate how predictive text analysis, using varying sets of contextual information, can be used to test and develop theories as to the contextual requirements of language processing. In our case, we compare the effects of incorporating contextual information about speakers with linguistic information to predict a speaker's communicative intentions. Importantly, by applying advanced methods for representing linguistic meaning and not relying on word-level frequency statistics, we are able to more confidently isolate the effects of linguistic fluency from contextual knowledge.

In terms of the case study we focus on, prior work on political rhetoric has established the importance of partisan identity and candidate status to understanding what is being said at any moment. Basic facts about the identities of the speakers are assumed as common knowledge in these contexts and provide critical background knowledge for understanding. Further, speakers act on the assumption that the listeners know something about who they are; this assumption defines many of their moves as they attempt to navigate the narratives about their candidacies. For language analysis, incorporation of these pieces of background information is natural as we would expect that it is precisely the sort of information which would not be present in the text itself. In these studies, we have shown that this information can be used to improve computational analysis of political rhetoric. In particular, considering "types" of speakers (in terms of party and status) can improve the ability to model and understand the intent of highly subjective statements.

Another key contribution of this work is the development of a novel annotated dataset of value and identity statements over the challenging domain of political debates. This dataset is freely available which we hope will help further work on these types of arguments, critical to the analysis of modern political rhetoric. Especially in light of recent elections around the world where identity arguments have played a central role, the capacity to model the ways in which these arguments are deployed is essential. The ability to computationally identify these arguments opens the door to applications such as automated real-time monitoring of the types of arguments being made across a variety of media.

Contextual knowledge is not just a factor for political rhetoric. When language occurs as an interaction between speakers and listeners, it is a form of social action (Austin, 1975). Therefore, accounting for the conditions of language's generation is a necessary step of analysis if we are to truly understand language in its communicative aspects, in any setting. But, even if we accept that "context matters", it is often difficult to determine exactly *when* context matters, and how its importance varies between domains and media. While we cannot exhaustively reapply this framework to other domains, constructs, and contextual data, language researchers across disciplines can conduct their own context-based analyses and contribute to the general body of knowledge on how context impacts language understanding.

We identify two main areas of future contribution to this framework. First, in the analysis of political discourse, we explore the effects of context in the scope of only one axis of language understanding (intention) and two components of circumstantial context (party and incumbency status), but these are tips of the iceberg. More contextual variables can be introduced into the system, for example knowledge about entities and events that are referenced in political text. Additionally, other forms of annotation can be applied to such data, and other data can be analyzed — for example, social media text or news media.

Another direction open area of future contribution is the advancement and integration of techniques for representing text in continuous space. Purely in terms of NLP and machine learning methods, our framework does not propose anything novel in itself. Using fairly established techniques, packaged together in a cohesive framework, we provide a tool for future text analysis. But, even now, cutting edge methods for mapping text to continuous space are being developed, and many advances are expected in the coming years. Thus, future work in the methods aspect of our research involves progressively integrating more complicated and sophisticated methods into context-based text research.

Two significant issues, which we do not address in our experiments, bear mentioning for those applying this framework to new datasets. First, there are restrictions introduced by using a "black-box" statistical method for learning sequential relationships in text. Our claims, which have to do with isolating external knowledge from the knowledge of "linguistic fluency", enabled by LSTM networks, do not account for external knowledge which may have been learned with the included word embeddings. At this current state, there does not exist a clear solution to this problem: increasing the interpretability of deep learning-based approaches for modeling language is a clear and constant goal of NLP research. As interpretable models continue to gain traction and validity, we will be working to augment our proposed framework so as to allow greater clarity in context-based findings.

A second concern has to do with the issue of sample complexity. The question one must ask is: how much data is too little, and how much is too much? If a researcher uses too small of a dataset, then it is possible that significant differences will be found by virtue of the model not being robust to sparsity. In the other case, it is possible that a researcher may achieve positive results just be throwing data at the model, and not by virtue of legitimate, true differences introduced by contextual factors. In this paper, we were unable to satisfactorily address this concern, though it is planned to explore the fallibility of the framework through experiments with synthetic data. In the meantime, it is suggested to replicate one's own results using smaller subsets of one's data.

Beyond the impact of our research on behavioral research and text analysis, we view our results as indicative of the fact that context ought to be integrated into computational research in language. We view the improvement in prediction when considering context to validate the claim that there are aspects of linguistic knowledge that are currently inaccessible even to advanced neural network-based representations of text. This reinforces work in other fields which finds that demographic factors improve predictive performance, but extends the set of evidence to include a more challenging area: speech acts in political discourse. From this, we conjecture that a more modular and nuanced approach to language quantification ought to be adopted, such that researchers take into account the variation in the participants of language creation: speakers and listeners.

To conclude, we reiterate that communication depends on understanding and adapting to context. When we model language and we do not account for this communicative aspect, any psychological or other effects we discover are thus invalidated. While contextual dynamics in linguistic communication can be quantified by behavioral experiments, text analysis methods can also address the question of context in communication. By sufficiently accounting for global linguistic knowledge via word embeddings, accounting for local linguistic knowledge using RNN-based text encoders, and incorporating contextual variables into a prediction framework, these questions can be asked of any text corpus for which we have meaning-based annotations.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

Aizawa, A. (2003). An information-theoretic perspective of tf-idf measures. *Information Processing & Management*, *39*(1), 45–65.

Austin, J. L. (1975). *How to do things with words*. Oxford University Press.

Bamman, D., & Smith, N. A. (2015). Contextualized sarcasm detection on twitter. In *Icwsm* (pp. 574–577).

Bamman, D., Dyer, C., & Smith, N. A. (2014). Distributed representations of geographically situated language. In *Proceedings of d annual meeting of the association for computational linguistics (short papers)*(Vol. 828, pp. 834).

Capelli, C. A., Nakagawa, N., & Madden, C. M. (1990). How children understand sarcasm: The role of context and intonation. *Child Development*, *61*(6), 1824–1841.

Carpini, M. X. D., Cook, F. L., & Jacobs, L. R. (2004). Public deliberation, discursive participation, and citizen engagement: A review of the empirical literature. *Annual Review of Political Science*, *7*, 315–344.

Caruna, R. (1993). Multitask learning: A knowledge-based source of inductive bias. In *Machine learning: Proceedings of the tenth international conference* (pp. 41–48).

Chong, D., & Druckman, J. N. (2007). Framing public opinion in competitive democracies. *American Political Science Review*, *101*(4), 637–655.

Christiano, T. (1995). Voting and democracy. *Canadian Journal of Philosophy*, *25*(3), 395–414.

Cohen, G. L. (2003). Party over policy: The dominating impact of group influence on political beliefs. *Journal of Personality and Social Psychology*, *85*(5), 808.

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on machine learning* (pp. 160–167).

Dawes, R. M., Van de Kragt, A. J., & Orbell, J. M. (1990). *Cooperation for the benefit of us?-not me, or my conscience*. University of Chicago Press.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14*(2), 179–211.

Fine, G. A. (1983). Sociological approaches to the study of humor. In *Handbook of humor research* (pp. 159–181). Springer.

Garimella, A., Banea, C., & Mihalcea, R. (2017). Demographic-aware word associations. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 2285–2295).

Garten, J., Hoover, J., Johnson, K. M., Boghrati, R., Iskiwitch, C., & Dehghani, M. (2017). Dictionaries and distributions: Combining expert knowledge and large scale textual data content analysis. *Behavior Research Methods*, 1–18.

Garten, J., Kennedy, B., Hoover, J., Sagae, K., & Dehghani, M. (2018). Incorporating demographic embeddings into language understanding. *Cognitive Science*.

Goodwin, C. (1984). Notes on story structure and the organization of participation. *Structures of Social Action: Studies in Conversation Analysis*, 225–246.

Graham, J., Haidt, J., & Nosek, B. A. (2009). Liberals and conservatives rely on different sets of moral foundations. *Journal of Personality and Social Psychology*, *96*(5), 1029.

Graves, A., & Schmidhuber, J. (2009). Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in neural information processing systems* (pp. 545–552).

Grice, H. P. (1975). *Logic and Conversation*, *1975*, 41–58.

Heritage, J., & Greatbatch, D. (1986). Generating applause: A study of rhetoric and response at party political conferences. *American Journal of Sociology*, *92*(1), 110–157.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

Hovy, D. (2015). Demographic factors improve classification performance. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers)* (Vol. 1, pp. 752–762).

Hovy, D., & Søgaard, A. (2015). Tagging performance correlates with author age. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 2: Short papers)* (Vol. 2, pp. 483–488).

Iyengar, S. (2005). Speaking of values: The framing of american politics. In *The forum* (Vol. 3, pp. 1–8).

Jerit, J. (2004). Survival of the fittest: Rhetoric during the course of an election campaign. *Political Psychology*, *25*(4), 563–575.

Ji, Y., Haffari, G., & Eisenstein, J. (2016). A latent variable recurrent neural network for discourse relation language models. arXiv:1603.01913.

Johannsen, A., Hovy, D., & Søgaard, A. (2015). Cross-lingual syntactic variation over age and gender. In *Proceedings of the nineteenth conference on computational natural language learning* (pp. 103–112).

Jost, J. T., Federico, C. M., & Napier, J. L. (2009). Political ideology: Its structure, functions, and elective affinities. *Annual Review of Psychology*, *60*, 307–337.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., & Wu, Y. (2016). Exploring the limits of language modeling. arXiv:1602.02410.

Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv:1412.6980.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Skip-thought vectors. In *Advances in neural information processing systems* (pp. 3294–3302).

López, I. H. (2015). *Dog whistle politics: How coded racial appeals have reinvented racism and wrecked the middle class*. Oxford University Press.

Macagno, F., & Walton, D. (2014). *Emotive language in argumentation*. Cambridge University Press.

Mendelberg, T. (2002). The deliberative citizen: Theory and evidence. *Political Decision Making, Deliberation and Participation*, *6*(1), 151–193.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Efficient estimation of word representations in vector space. arXiv:1301.3781.

Mikolov, T., Yih, W.-t., & Zweig, G. (2013b). Linguistic regularities in continuous space word representations. In *Hlt-naacl* (pp. 746–751).

Musi, E., Ghosh, D., & Muresan, S. (2016). Towards feasible guidelines for the annotation of argument schemes. *ACL*, *2016*, 82.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th international conference on machine learning (icml-10)* (pp. 807–814).

Palangi, H., Deng, L., Shen, Y., Gao, J., He, X., Chen, J., & Ward, R. (2016). Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, *24*(4), 694–707.

Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, *71*(2001), 2001.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).

Priddy, K. L., & Keller, P. E. (2005). *Artificial neural networks: An introduction* (Vol. 68). SPIE Press.

Richardson, H. S. (2002). *Democratic autonomy: Public reasoning about the ends of policy*. Oxford University Press on Demand.

Searle, J. R. (1969). *Speech acts: An essay in the philosophy of language* (Vol. 626). Cambridge University Press.

Sniderman, P. M., & Theriault, S. M. (2004). The structure of political argument and the logic of issue framing. Studies in public opinion: Attitudes, nonattitudes, measurement error, and change, 133–65.

Sundermeyer, M., Schlüter, R., & Ney, H. (2012). Lstm neural networks for language modeling. In *Thirteenth annual conference of the international speech communication association*.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104–3112).

Suykens, J. A., & Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, *9*(3), 293–300.

Svozil, D., Kvasnicka, V., & Pospichal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Chemometrics and Intelligent Laboratory Systems*, *39*(1), 43–62.

Swanson, D. L., & Mancini, P. (1996). *Politics, media, and modern democracy: An international study of innovations in electoral campaigning and their consequences*. Greenwood Publishing Group.

Viera, A. J., & Garrett, J. M. (2005). Understanding interobserver agreement: The kappa statistic. *Family Medicine*, *37*(5), 360–363.

Yang, Y., & Eisenstein, J. (2015). Putting things in context: Community-specific embedding projections for sentiment analysis. *Arxiv-Social Media Intelligence*.