Explicit Stance Detection in the Political Domain: A New Concept and Associated Dataset

Alexander R. Caceres-Wright[1(✉)], Naveen Udhayasankar[1], Grant Bunn[2], Stef M. Shuster[3], and Kenneth Joseph[1]

[1] University at Buffalo, Buffalo, NY, USA
{caceresw,naveenud,kjoseph}@buffalo.edu
[2] North Carolina State University, Raleigh, NC, USA
gbunn@ncsu.edu
[3] Michigan State University, East Lansing, MI, USA

Abstract. Stance detection, defined as the task of classifying an individual's attitude towards a target person or concept, offers the potential to understand political opinions at scale using social media data. However, recent studies have questioned the robustness and accuracy of current stance detection methods, highlighting issues such as generalizability in time and inconsistencies in annotations driven by subtle differences in annotation task design. We argue that central to these challenges is the unresolved question of what constitutes an expression of stance. To address this, the present work introduces a distinction between *explicit* and *implicit* stance expressions, and argue that a focus on explicit stance detection addresses many of the existing concerns with modern stance detection methods. To facilitate research on explicit stance detection, we then present a novel (and public) dataset of over 1000 tweets across 13 stance targets for explicit stance detection and evaluate baseline models to establish a foundation for future research in this area.

Keywords: Stance Detection · Large Language Models · Politics · Social Media

1 Introduction

Stance detection [19], the task of classifying an individual's attitude towards a target, has become one of the most popular tasks in the area of natural language processing [1,10,13]. Of particular interest is the area of *political* stance detection, where scholars have focused on detecting attitudes towards particular candidates [11] and towards broader politically-relevant claims [5]. The promise of effective political stance detection methods is that they may be able to help us understand political attitudes without the implementation of surveys that may be costly, or challenging (e.g. in hard-to-reach populations) [2,12].

R. Thomson et al. (Eds.): SBP-BRiMS 2024, LNCS 14972, pp. 3–14, 2024.
https://doi.org/10.1007/978-3-031-72241-7_1

However, several recent works have raised concerns about how effectively modern stance detection actually models public opinion. First, detailed analyses of existing methods suggest that existing stance detection methods are not robust to novel datasets, either with respect to changes in 1) who we aim to detect stance towards [16,17] or 2) the time at which we aim to detect stance [14,15]. Second, there are important questions about the extent to which we should expect opinion expressed on social media to actually reflect individuals' opinions, even when controlling for time [7]. Finally, gold standard datasets are subject to variation based on (often unreported) decisions about how to present the task to annotations [6]. For example, presenting annotators with the exact same tweets, but opting to present, or not present, user information alongside the tweet lead to significantly different annotations, even if both designs seem reasonable.

At the heart of these challenges is a core and largely unresolved question in both the NLP literature [7] and an open area of discussion in the relevant sociolinguistics literature [8]: *what should a person, or model, count as an expression of stance*? For example, consider a setting in which we aim to analyze a user's stance towards Donald Trump, given their tweet "I hate Kamala Harris." In the context of the 2024 presidential election, a politically knowledgeable annotator (or model) might reasonably infer that this user is pro-Trump. However, the inference relies upon an understanding that at this point in the 2024 presidential election cycle, there are two presumptive candidates - one for the Democrats and one for the Republicans. Furthermore in the context of the lead-up to the 2020 elections, one may also need to consider the possibility that the user was anti-Trump *and* anti-Harris and supporting a different candidate still in the running for the Democratic nomination [18].

This example highlights the aforementioned temporality concerns, but also brings to light an often recognized [4,11,17], but as-yet-unnamed distinction in the stance detection literature. Namely, there is a difference between *explicit* stance—where a user references the stance target and their stance towards them ("I love Trump")—versus *implicit* stance, where an annotator (or model) must *infer* stance based on available information. Designing stance detection models based only on *explicit* stance expressions has the limitation of lower recall, in that explicit user expressions of stance are relatively rare, whereas stance can (in theory) be inferred for any user [11]. But there are also a number of benefits-namely, we can expect that a model trained to detect only explicit mentions of stance is much more likely to avoid challenges of temporal drift and variations in annotation design.

To that end, the present work makes the following contributions:

- We introduce the concept of *explicit* stance detection, differentiating it from implicit stance detection and noting its' relative benefits and drawbacks
- We develop a novel and publicly available dataset[1] for explicit stance detection, consisting of over 1000 tweets across 13 different stance targets

[1] https://docs.google.com/spreadsheets/d/1ux2ap-vStSqhZ32VWtmQerrlo3qcDBl73PET9ECS7BA/edit?usp=sharing.

– We develop and evaluate a number of initial, straightforward baseline models for explicit stance detection for others to build upon in future work

2 Data

In this section, we describe the development of our annotated dataset for explicit stance detection. We define a tweet as expressing explicit stance when two conditions are met: 1) the tweet (explicitly) mentions the stance target, and 2) the tweet unambiguously expresses a stance towards the target. In cases where a tweet does not meet condition 1), stance should be labeled as "Not relevant," i.e. not relevant to the task. In cases where a tweet meets condition 1) but not condition 2), the tweet is labeled as "Neutral."

To begin, we draw on the publicly available (upon request) dataset from Shuster et al. [18], who collected over 500M tweets sent during 2020 leading up to the U.S. presidential election. Their data is of interest precisely because we expect *implicit* stance detection to be challenging in such a setting: users varied widely in their attitudes towards individual candidates, and thus (as in our introductory example) inference of stance towards one candidate given stance towards any other candidate is challenging. Here, then, a model that infers stance only from *explicit* stance expressions is particularly useful and desirable.

Table 1. Statistics for the annotated dataset for explicit stance detection. All agreement scores are Krippendorf's alpha; the final two percentage columns reflect the final, gold-standard dataset

Candidate	Mentions Agreement	Full Agreement	Agreement w/o Neutral	% Neutral	% Where Filter Failed
Michael Bennet	0.80	0.55	0.68	44.90	6.12
Joe Biden	–0.01	0.57	0.72	25.64	0.00
Pete Buttigieg	0.61	0.71	0.91	44.90	6.12
Amy Klobuchar	1.00	0.46	0.63	31.58	0.00
Deval Patrick	0.96	0.43	0.63	20.20	26.26
Bernie Sanders	0.92	0.43	0.65	20.20	5.05
Tom Steyer	1.00	0.58	0.56	29.67	0.00
Donald Trump	0.55	0.46	0.83	30.53	4.21
Andrew Yang	0.56	0.57	0.81	31.18	4.30
Tulsi Gabbard	1.00	0.71	0.80	29.00	0.00
Michael Bloomberg	1.00	0.52	0.58	23.71	0.00
John Delaney	1.00	0.48	0.70	40.00	0.00
Elizabeth Warren	0.79	0.49	0.83	26.80	2.06

From the dataset published by [18], we used a simple keyword-based filtering approach to identify all tweets that explicitly mentioned one of the twelve candidates in the 2020 DNC Primary, as well as Donald Trump. Stance targets are listed in the first column of Table 1. To do so, we created a dictionary containing a list of terms relevant to each candidate. If a tweet's text did not contain any of these terms it was excluded from the final dataset. This filtering step resulted in a set of 17,429,630 unique, non-retweeted tweets, or roughly 30.35% of the set of unique non-retweets in the original dataset. Once we had this filtered dataset,

we then sampled approximately 100 tweets per politician. From this sample, we then had three separate annotators label each tweet for two categories: 1) if the tweet did in fact mention the candidate, and 2) if so, the stance of the tweet towards that particular candidate. Following prior work on creating accurate, curated datasets, annotators of the data were the authors of the paper [6,7].

For (explicit) stance annotation, we used the standard three-label approach of positive, negative or neutral. Once all annotators completed their annotations, we calculated Krippendorff's alpha [9] to determine the level of agreement across the three annotators. Finally, for each tweet, we then calculated a single, final gold label by the principle of majority rules; i.e. by picking the label that at least two of the three annotators agreed on. In the limited number of cases where all three of the annotators disagreed, we asked an annotator who had not seen those tweets before to label those tweets and used these labels as gold.

Table 1 shows summary statistics for our explicit stance detection dataset. In the first column, we show annotator agreement (in terms of Krippendorf's alpha) for whether or not a tweet meets condition 1 above; i.e. whether or not the tweet actually mentions the candidate. The table reflects several important points.

First, detecting a mention of an account was not always easy. One measure of this is presented in the last column in Table 1, which displays the number of tweets where our keywords filter failed, meaning our filtering identified it as a tweet that mentioned that candidate when in fact it did not. Most obvious in this setting are issues surrounding Deval Patrick, whose last name is popular and thus filters in many irrelevant tweets. However, in most cases, our keyword filter was effective in identifying tweets relevant to candidates. Perhaps more interesting is that, as the "Mentions Agreement" column shows, even for human annotators, determining whether or not a tweet explicitly mentioned a candidate was challenging, especially for some candidates. This could be due to the unfamiliarity of some of the annotators with specific keywords our filter used, as well as differing opinions on what they considered to be explicit mentions. For example, some annotators considered decontextualied hashtags added at the end of tweets, such as '#YangGang' and '#Berniecrats', as explicit references to Andrew Yang and Bernie Sanders, respectively, while others did not. Ultimately, we opted to include these as explicit references in our dataset unless they had absolutely nothing to do with the rest of the tweet content (e.g. in the few instances of hashtag spamming we observed).

Second, even when annotators agreed that a candidate was mentioned, determining whether stance was expressed was hard for some candidates. When including the neutral category, Krippendorf alphas ranged from 0.43–0.71, in line with agreement on other annotation tasks for social media [3] but lower than when we a priori expected given an explicit mention of a candidate. These agreements increased, however, to a range of 0.56–0.91 when we exclude cases where the final gold-standard label was neutral. This reflects that allowing neutral labels vastly increase the amount of disagreement between the annotators. Finally, and related, is that our explicit stance detection dataset contains

significantly more neutral labels (roughly 30% of all annotations) than other existing datasets in the literature, which often have 10% or fewer neutral labels.

In summary, our explicit stance detection dataset shows that there are interesting challenges to be addressed in explicit stance detection versus implicit stance detection, namely in the context of the disproportionate number of cases in explicit stance detection (relative to the standard stance detection task) where a candidate is mentioned but where an explicit stance is not expressed. We now turn to an initial exploration of explicit stance classification, defining a suite of simple but reasonable baseline models and evaluating them.

3 Methods

In addition to contributing this new dataset to the research community, we develop a suite of three baseline methods to ground future work aiming to use our data to make predictions about expressions of explicit stance. Notably, we consider only open-source approaches to classification that are fully reproducible by the research community, i.e. that do not rely on a private entity opting not to depreciate a particular model version. More specifically, we consider two different models using three different approaches to leveraging large language models (LLMs) to accomplish these two tasks: a prompt-based use of Meta's Llama-2 model[2] [20] ("Llama-2 Generative"), a multiple choice pipelining approach with Llama-2[3] ("Llama-2 Pipeline"), and a similar multiple choice approach using the recently released Deberta-Polistance model,[4] a Deberta model fine-tuned for stance detection in a political setting.

The task of explicit political stance detection can be subdivided into two tasks: politician identification and stance classification. We compare two uses of Llama-2 in order to assess whether or not we can use simple prompts to see whether a generative approach based on prompting can solve both identification and stance classification in one step. In the Llama-2 Generative model, we prompt the model to provide both 1) which candidates were mentioned and 2) what the stance is towards that candidate. With respect to model hyperparameters, we experiment with a variety of 1) prompts, 2) temperatures, and 3) character limits (on the latter point, in order to assess how quickly we can complete the task). Hyperparameters evaluated are presented in the Appendix; we here take an optimistic view on performance and select for reporting results from the best-performing hyperparameter combination, because even with these optimistic scores the model still underperforms the other approaches.

In our Llama-2 Pipeline approach, we instead tell the model which candidate to provide a stance classification for. We do so via use of the `pipeline` abstraction provided by the `transformers` python library. More specifically, we evaluate the probability of the phrases "Positive towards [target]," "Neutral towards [target]," and "Negative towards [target]" in the model's next word prediction, and

[2] https://huggingface.co/meta-llama/Llama-2-13b-chat-hf.
[3] https://huggingface.co/meta-llama/Llama-2-13b-hf.
[4] https://huggingface.co/mlburnham/deberta-v3-large-polistance-affect-v1.1.

take the option with the highest probability. While this abstraction can be useful especially for individuals who do not want to, or have the necessary experience, to directly work with the models to achieve the same result, as we will see there are some cases where directly engaging with the model can achieve a better result.

Finally, in our Deberta-Polistance Pipeline approach, we again use a classification pipeline, and specify a set of labels for the pipeline to choose from. This is a three-step process. First, we format the labels with the candidate in question. Second, we pass the text of the tweets as well as our labels to the model in the form of a fill-in-the-mask prompt. Finally, we extract the stance with the highest next phrase probability according to the model.

For all three of our approaches, we passed each tweet to the model in separately. In the case of Llama's text generation, we tested a number of different prompt options, as well as temperatures and limits on the maximum number of characters that the response could return. Using the zero shot classification pipelines, we only ran through each tweet once, using the same labels as in Llama-2 Pipeline approach. We opt not to consider hyperparameters for this approach. Once we had the labels for each tweet, we then calculated the accuracy, precision, and recall for each model as well as each parameter combination for Llama-2 on a per candidate basis. In the case there was a mismatch between the number of annotations returned by the LLM, which primarily occurred in the case of Llama's text generations, we computed each of precision accuracy and recall in two different ways. First we labelled each tweet that Llama did not provide a stance for as neutral, and second we dropped any tweet from the set of gold labels that did not have a corresponding entry in the LLM labelled tweets.

4 Results

4.1 Detecting Candidate Mentions

As shown in Table 3 even in the best case our Llama-Generate method fails to correctly identify the candidate in almost 20% of the gold label tweets, and for many candidates fails on the identification task significantly more often than this. However, as we can see from Table 4, in the cases where Llama-Generate did successfully identify the politicians explicitly mentioned in the tweet, it was able to correctly identify the stance with a higher degree of accuracy than the Llama-Pipeline method. Failures to correctly identify the candidate were impacted by hyperparameter settings. For example, using a lower limit on the number of characters the model is allowed to return may cut off the part of the response discussing the relevant candidate. However, as we can see in Table 3 across all the different parameter combinations we tried, there were still significant failures even when extending to significantly longer response limits. To this end, another possible explanation could be in our choice of prompt, which is something else we attempted to control for. We tried four different prompts, which can be found in the appendix, and used each of them for each tweet. Again, these prompt

options did not have a large effect on the both the number of responses returned nor the metrics which will be discussed later in this paper (Table 2).

Table 2. Percent of Gold Labels Annotated by Llama Generate

Candidate	Percent Annotated	Candidate	Percent Annotated
Bennet	65.97	Patrick	52.92
Biden	77.08	Steyer	70.71
Bloomberg	76.07	Sanders	68.26
Buttigieg	75.09	Trump	53.82
Delaney	83.08	Warren	68.25
Gabbard	72.73	Yang	75.08
Klobuchar	73.84		

In contrast to the challenges faced by Llama-Generate, Table 4 shows that our filter was successful in the majority of cases, struggling to correctly identify only those candidates with more common names, such as Deval Patrick, or candidates such as Donald Trump, which have multiple prominent family members which share their last name thus leading to false positives. In summary, then, our findings for candidate mention performance conform to other relevant recent work on a broader set of computational social science tasks (including stance detection) suggesting that using domain-relevant knowledge (in this case, keywords to detect candidate mentions) is still an effective approach beyond relying on generative models to complete tasks without such knowledge [21].

Table 3. Percent of Label Allocation for Llama Using Generate

Candidate	Gold labels	Avg. Annotations	Gold Pct. Positive	Avg. Pct. Positive Labels	Gold Pct. Negative	Avg. Pct. Negative Labels	Gold Pct. Neutral	Avg. Pct. Neutral Labels
Bennet	98	64.65	7.14	27.55	41.84	24.27	44.90	48.18
Biden	117	90.19	8.55	25.72	65.81	58.25	25.64	16.02
Bloomberg	97	73.79	3.09	26.34	73.20	55.82	23.71	17.84
Buttigieg	98	73.59	7.14	24.21	41.84	50.09	44.90	25.70
Delaney	95	78.93	14.74	29.41	45.26	30.27	40.00	40.32
Gabbard	100	72.73	31.00	37.56	40.00	36.90	29.00	25.54
Klobuchar	95	70.15	6.32	32.66	62.11	37.62	31.58	29.72
Patrick	99	52.39	6.06	28.06	47.47	18.49	20.20	53.45
Steyer	91	64.35	25.27	40.13	45.05	25.54	29.67	34.32
Sanders	99	67.58	67.68	77.01	7.07	11.15	20.20	11.84
Trump	95	51.13	2.11	21.17	63.16	71.44	30.53	7.38
Warren	97	66.20	13.40	30.29	57.73	49.34	26.80	20.37
Yang	93	69.83	32.26	46.26	32.26	17.87	31.18	35.88

Table 4 shows accuracy under two conditions: 1) when we added in a label of neutral for any tweet in the gold annotations not found in the labelled set, and 2) by removing any tweet found in the gold set not found in the labelled set. Examining the results, we can make several general claims.

Table 4. Accuracy in Two Cases

Candidate	LLama Generate Adding Neutral	LLama Generate Dropping Diff From Gold	LLama Pipeline Adding Neutral	LLama Pipeline Dropping Diff From Gold	Deberta-Polistance Adding Neutral	Deberta-Polistance Dropping Diff From Gold
Bennet	0.46	0.44	0.38	0.36	0.61	0.59
Biden	0.56	0.64	0.66	0.71	0.79	0.86
Bloomberg	0.48	0.55	0.23	0.18	0.74	0.79
Buttigieg	0.45	—	0.42	−1.00	0.49	−1.00
Delaney	0.41	0.39	0.42	0.50	0.61	0.67
Gabbard	0.48	0.52	0.36	0.36	0.59	0.59
Klobuchar	0.46	0.50	0.60	0.50	0.71	0.25
Patrick	0.26	0.35	0.26	0.26	0.46	0.47
Steyer	0.48	0.53	0.43	0.38	0.74	0.69
Sanders	0.56	0.71	0.07	0.09	0.71	0.65
Trump	0.42	0.56	0.63	0.33	0.54	0.44
Warren	0.47	0.55	0.42	0.52	0.71	0.62
Yang	0.51	0.56	0.35	0.35	0.67	0.67

First, overall, we find that the Deberta-Polistance model performed better than Llama-2. Table 4 shows the average accuracy for text generation by Llama-2 as well both llama-2 Pipeline and Deberta-Polistance. As we can see, in all cases using Llama in a zero shot classification pipeline performs the worst out of the three models we used, while the polistance performs the best. We also see that while using Llama to generate text is not very successful at identifying the correct candidate, when it does it performs pretty well with identifying the correct stance. This suggests that if another tool is used to identify who is explicitly mentioned in the text, using a text generation such as Llama 2 may allow for improved human readability at a small trade off for accuracy. Additionally, Meta recently released Llama 3 which it reports has even better performance than Llama 2, which may improve the results of completing these tasks.

Second, however, is that this overall pattern masks significant variability across targets. Notably, Llama-Generate outperforms Deberta-Polistance for two of the more important candidates in the 2020 Election (Donald Trump and Bernie Sanders). More broadly, across models, our results is that there is wide variability across different candidates by our three models. As we can see from Table 4 the models struggled the most with Pete Buttigieg and Deval Patrick. This may be due to their relatively lesser known status as compared to the other candidates in our dataset. Even more interesting is that the agreement between our three annotators for Buttigieg is on the higher end of our agreement scores.

This variability, in turn, stems from a third interesting point: models had very different responses to the Neutral label, in ways that impacted their performance. For example, we see that gold labels for Pete Buttigieg had a large proportion of neutral tweets and from Table 3, and in turn that Llama-2 tends to favor positive or negative labels much more than neutral labels. More generally, out of the three stance labels used in this paper (positive, negative, and neutral), neutral was consistently the hardest for both human annotators as well as all three models used. As we can see in Table 3, three out of our thirteen candidates, Bennet, Buttigieg and Delaney, received a stance label of neutral about 40% of the time, while the remaining ten candidates had closer to 20% of their tweets labelled as neutral. While this could be due to the fact that the more prominent a candidate is, the greater the chance that individuals feel strongly in support or against them and are more likely to make those views public. Interestingly, with

the exception of Donald Trump and Deval Patrick, the percent of tweets that llama labelled as neutral was fairly similar to the percentage of gold annotations.

5 Discussion

While the core focus of our work is not effective classification but rather the identification of explicit stance detection as a task and the contribution of a public dataset for this task, our classification exercise here presents three interesting questions to be explored in future work:

- Explicit stance detection almost necessarily results in significantly more neutral labels than in stance detection more generally, because inference of user intent is discouraged. This has implications for downstream modeling that are interesting to explore further.
- This disparity in neutral labels is further impacted by candidate status, as is which model performs best. Why models show variability across targets is an interesting point that could be further explored with our data
- Finally, it is interesting to consider how best to make use of relevant domain knowledge for explicit stance detection, as existing work tends to make use of domain knowledge to draw inferences potentially unexpressed in the text itself (and thus not relevant in the explicit setting) [11]

6 Conclusion

The present work is the first to differentiate explicit stance detection as its own concept, arguing that defining and operationalizing explicit stance detection may address certain concerns with existing stance detection models. To help bootstrap the study of explicit stance detection, we have curated a dataset with two distinct data points per datapoint: if the data point does in fact mention the target in question and separately from that the stance of the tweet towards that target in particular. This second point is especially important for the given task since one tweet can mention multiple entities and can have different stances towards each named individual.

Taken together, our findings offer demonstrable evidence of the importance of explicit stance detection, and the potential for large language models to conduct explicit stance detection given expertly curated datasets. We contribute to the literature by offering a framework for using open source models to do so, as well as a dataset which can be used as a benchmark to test. Given the moderate success of both an out-of-the-box model such as Llama-2 and a fine tuned model such as Deberta-Polistance, we leave it to future scholars to continue pursuing how to correctly identify which individuals, if any are explicitly mentioned by a given tweet.

Appendix

We considered the following prompts for Llama-Generate:

1. The following is a tweet from the 2020 US presidential election, give me the politicians it explicitly makes mention of and if it is for, against, or neutral towards each politician it mentions. Do not return the text of the tweet nor any emojis, and return your response in the following format: Politicians Name: 'stance'(for or against or neutral). Separate each politician and stance pair with a semicolon. Here is the tweet:
2. The following is a tweet from the 2020 US presidential election, give me the politicians it explicitly makes mention of and if it is for, against, or neutral towards each politician it mentions. Do not return anything besides the politician(s) mentioned and stance pair(s). Return your response in the following format: Politicians Name: 'stance'(for or against or neutral). Separate each politician and stance pair with a semicolon. Here is the tweet:
3. The following is a tweet from the 2020 US presidential election, give me the politicians it explicitly makes mention of and if it is for, against, or neutral towards each politician it mentions. Do not return anything besides the politician(s) mentioned and stance pair(s). Return your response in the following format: Politicians Name: 'stance': for or against or neutral. Seperate each politician and stance pair with a semicolon. Only include a politician if they are on the following list; Pete Buttigieg, Michael Bloomberg, Joe Biden, Michael Bennet, John Delaney, Tulsi Gabbard, Amy Klobuchar, Deval Patrick, Bernie Sanders, Tom Steyer, Donald Trump, Elizabeth Warren, and Andrew Yang. Here is the tweet:
4. The following is a tweet from the 2020 US presidential election, give me the politicians it explicitly makes mention of and if it is for, against, or neutral towards each politician it mentions. Do not return the text of the tweet nor any emojis, and return your response in the following format: Politicians Name: 'stance': for or against or neutral. Seperate each politician and stance pair with a semicolon. Only include a politician if they are on the following list; Pete Buttigieg, Michael Bloomberg, Joe Biden, Michael Bennet, John Delaney, Tulsi Gabbard, Amy Klobuchar, Deval Patrick, Bernie Sanders, Tom Steyer, Donald Trump, Elizabeth Warren, and Andrew Yang. Here is the tweet:

For temperature, we considered each of: 0.001, 0.25, 0.5, 0.75, 0.8. For character limits, we considered 25, 50, 75, and 125 characters.

References

1. AlDayel, A., Magdy, W.: Stance detection on social media: state of the art and trends. Inf. Process. Manag. **58**(4), 102597 (2021)
2. Conrad, F.G., Gagnon-Bartsch, J.A., Ferg, R.A., Schober, M.F., Pasek, J., Hou, E.: Social media as an alternative to surveys of opinions about the economy. Soc. Sci. Comput. Rev. (2019). https://doi.org/10.1177/0894439319875692

3. Du, Y., Masood, M.A., Joseph, K.: Understanding visual memes: an empirical analysis of text superimposed on memes shared on twitter. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 14, pp. 153–164 (2020)
4. Ebrahimi, J., Dou, D., Lowd, D.: Weakly supervised tweet stance classification by relational bootstrapping. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 1012–1017 (2016). http://ix.cs.uoregon.edu/%7Edou/research/papers/emnlp16.pdf
5. Hardalov, M., Arora, A., Nakov, P., Augenstein, I.: A survey on stance detection for mis- and disinformation identification (2022). http://arxiv.org/abs/2103.00242
6. Joseph, K., Friedland, L., Hobbs, W., Lazer, D., Tsur, O.: ConStance: modeling annotation contexts to improve stance classification. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1115–1124 (2017)
7. Joseph, K., Shugars, S., Gallagher, R., Green, J., Mathé, A.Q., An, Z., Lazer, D.: (mis) alignment between stance expressed in social media data and public opinion surveys. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 312–324 (2021)
8. Kiesling, S.F.: Stance and stancetaking. Ann. Rev. Linguist. **8**(1), 409–426 (2022). https://doi.org/10.1146/annurev-linguistics-031120-121256
9. Krippendorff, K.: Computing krippendorff's alpha-reliability (2011)
10. Küçük, D., Can, F.: Stance detection: a survey. ACM Comput. Surv. **53**(1) (2020). https://doi.org/10.1145/3369026
11. Li, A., Liang, B., Zhao, J., Zhang, B., Yang, M., Xu, R.: Stance detection on social media with background knowledge. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 15703–15717. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.emnlp-main.972. https://aclanthology.org/2023.emnlp-main.972
12. McGregor, S.C., Mourão, R.R., Molyneux, L.: Twitter as a tool for and object of political and electoral activity: considering electoral context and variance among actors. J. Inf. Technol. Politics **14**(2), 154–167 (2017). https://doi.org/10.1080/19331681.2017.1308289
13. Mohammad, S., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: Semeval-2016 task 6: detecting stance in tweets. In: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), pp. 31–41 (2016)
14. Mu, Y., Jin, M., Bontcheva, K., Song, X.: Examining temporalities on stance detection towards COVID-19 vaccination. arXiv preprint arXiv:2304.04806 (2023)
15. Ng, L.H.X., Carley, K.: Flipping stance: social influence on bot's and non bot's COVID vaccine stance. arXiv preprint arXiv:2106.11076 (2021)
16. Ng, L.H.X., Carley, K.M.: Is my stance the same as your stance? A cross validation study of stance detection datasets. Inf. Process. Manag. **59**(6), 103070 (2022). https://www.sciencedirect.com/science/article/pii/S0306457322001728
17. Sen, I., Flöck, F., Wagner, C.: On the reliability and validity of detecting approval of political actors in tweets. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1413–1426. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.emnlp-main.110. https://www.aclweb.org/anthology/2020.emnlp-main.110
18. Shuster, S.M., Campos-Castillo, C., Madani, N., Joseph, K.: Who supports Bernie? Analyzing identity and ideological variation on Twitter during the 2020 democratic

primaries. Plos One **19**(4), e0294735 (2024). https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0294735
19. Somasundaran, S., Wiebe, J.: Recognizing stances in ideological on-line debates. In: Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, pp. 116–124. Association for Computational Linguistics (2010). http://dl.acm.org/citation.cfm?id=1860645
20. Touvron, H., et al.: Llama 2: open foundation and fine-tuned chat models (2023)
21. Ziems, C., Held, W., Shaikh, O., Chen, J., Zhang, Z., Yang, D.: Can large language models transform computational social science? Comput. Linguist. **50**(1), 237–291 (2024)