This is a repository copy of *A lightweight approach for user and keyword classification in controversial topics*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/224134/

Version: Accepted Version

A Lightweight Approach for User and Keyword Classification in Controversial Topics

Ahmad Zareie, Kalina Bontcheva, Carolina Scarton

Department of Computer Science, The University of Sheffield, Sheffield, UK
{*a.zareie, k.bontcheva, c.scarton*}*@sheffield.ac.uk*

Abstract

Classifying the stance of individuals on controversial topics and uncovering their concerns is crucial for social scientists and policymakers. Data from Online Social Networks (OSNs), which serve as a proxy to a representative sample of society, offers an opportunity to classify these stances, discover society's concerns regarding controversial topics, and track the evolution of these concerns over time. Consequently, stance classification in OSNs has garnered significant attention from researchers. However, most existing methods for this task often rely on labelled data and utilise the text of users' posts or the interactions between users, necessitating large volumes of data, considerable processing time, and access to information that is not readily available (e.g. users' followers/followees). This paper proposes a lightweight approach for the stance classification of users and keywords in OSNs, aiming at understanding the collective opinion of individuals and their concerns. Our approach employs a tailored random walk model, requiring just one keyword representing each stance, using solely the keywords in social media posts. Experimental results demonstrate the superior performance of our method compared to the baselines, excelling in stance classification of users and keywords, with a running time that, while not the fastest, remains competitive.

Keywords: Users Classification Keyword Classification Stance Detection Random Walk

1 Introduction

In society, individuals may hold varying opinions on controversial topics. Controversial topics refer to subjects that create polarisation in individuals' stances, leading to dichotomous opinions (e.g., Leave vs. Remain in the Brexit context, Pro-vaccine vs. Anti-vaccine in the COVID-19 context, or support for a specific candidate in an election); this resulting position is known as the stance. In controversial topics, classifying users based on their stance, discovering the concerns within each class and tracking the evolution of classes and concerns over time are regarded as crucial for social scientists and policymakers [2]. Data collected

from Online Social Networks (OSNs) provide valuable insights for this crucial task since it can be seen as a proxy for opinions in societies. In these networks, individuals express and discuss their viewpoints to disseminate their opinions and potentially influence others' perspectives. Therefore, the stance of users can be inferred based on their activities in OSNs; additionally, the keywords they use in their posts can express their concerns. In this paper, we propose a **random walk approach** to classify users and keywords in OSNs based on their **stance on a controversial topic**. This approach is lightweight, relying solely on keywords shared by users for the classification task. This approach only requires one keyword to represent each stance as prior knowledge, without the need for pre-determined thresholds. Hereafter, we use 'hashtag' interchangeably with 'keyword', but the method can be used with any token as keywords.

Analysing data in OSNs to discover the stance of users and the generated content (posts) has increasingly become the focus of researchers from various disciplines, and a multitude of methods have been proposed for this task [2, 3]. Previous work focuses mainly on supervised models, which rely on a set of data known as labelled (training) data as prior knowledge. These supervised methods could be divided into two groups: (i) content-based [11, 5]: requiring a set of texts as labelled data and utilising the text of posts to understand the stance, and (ii) interaction-based [1, 13, 14]: relying on a set of users representing each stance as labelled data and using the interactions between pairs of users (e.g., likes, retweets, replies, and mentions on Twitter – now X) to understand the stance. The labelled data required in supervised methods is typically generated through a manual process, involving human judgments, which is time-consuming (particularly when tracking the evolution of public opinion over time [2]) and highly affected by annotators' biases. Data labelling may also lead to language- and domain-dependant models, requiring further data annotation for handling new domains and languages. Unsupervised methods [6, 10, 12] have also been proposed for classifying individuals' stances without the need for labelled data. These methods utilise linguistic features extracted from post content, such as lexicons, grammatical dependencies, and the latent meanings of words in context. Alternatively, they may leverage users' profile features to infer their stance. However, methods relying on linguistic features may encounter challenges in multilingual domains due to their dependency on language-specific characteristics [3]. Meanwhile, those utilising profile features may face difficulties in generalising across diverse social media platforms, given the variability in profile features offered by each platform.

Our method employs a lightweight unsupervised approach, that is language- and profile-agnostic, setting it apart from these previous studies. Moreover, it does not rely on user interactions, which may not always be available on certain social media platforms (e.g., users sharing opinions in a Telegram group). To our knowledge, Darwish et al. [5] and Coletto et al. [4] are the only previous work addressing a classification task similar to our work. Darwish et al. [5] propose an approach that relies on predefined sets of users, each representing a particular stance. The stance of each user is then determined based on their similarity to these predefined sets. Although this method relies on keywords

(hashtags) shared by users to detect stances, similar to ours, it differs in that it requires a set of users for each stance as labelled data, whereas our method relies on only one keyword for each stance. Coletto et al. [4] propose an approach that classifies keywords and users iteratively. Similar to our approach, this method relies on the keywords shared by users and requires only one keyword for each stance as prior knowledge. However, this method requires pre-defined thresholds for classification, and determining the optimal thresholds for various domains is not trivial and, in some cases, not feasible. Additionally, our experimental evaluation demonstrates that our lightweight method outperforms both of these methods.

The main contribution of this paper is the lightweight classification method, which utilises a tailored random walk approach to effectively classify users and hashtags with minimal user data and prior knowledge. We apply our method, called the Lightweight Random Walk Method (LRM), to analyse conversations about the UK General Election 2019 and Brexit on Twitter (now X). Our approach shows superior performance compared to baseline methods in accurately classifying users into different stance classes and identifying the most prominent hashtags within each class. Our method also offers competitive running time compared to baseline methods. Additionally, the proposed model is utilised to track the evolution of stance classes over time in both the general election and Brexit datasets. Results obtained by tracking the evolution in the Brexit data are consistent with previous studies applying content and text analysis [8, 9]. Although there is a lack of previous studies on analysing stance in the 2019 general election, our work corroborates with reports [7] showing quantitative analysis about this election in social media.

2 Lightweight Random Walk Method (LRM)

Let $U = \{u_1, u_2, \cdots, u_n\}$ be a set of users sharing a set of hashtags $H = \{h_1, h_2, \cdots, h_m\}$, where n and m indicate the number of users and hashtags, respectively. Let $R_{n \times m}$ be the sharing records with R_{ki} denoting the number of times u_k shares h_i; $R_{ki} = 0$ if u_k does not share h_i. The notation $\mathcal{R}_i$ indicates the total number of times that hashtag h_i is shared, calculated as the sum of sharing instances across all users, i.e., $\mathcal{R}_i = \sum_{k=1}^{n} R_{ki}$, and notation $\mathscr{R}_k$ represents the total number of times u_k has shared hashtags, calculated as the sum of sharing instances of u_k across all hashtags, i.e., $\mathscr{R}_k = \sum_{i=1}^{n} R_{ki}$.

Suppose there are t stances (represented as t classes) towards a controversial topic, with each stance c (where $c \in \{1, \cdots, t\}$) associated with a given hashtag s_c, referred to as the seed hashtag or simply the seed. The objective is to classify hashtags and users into these t classes using only one hashtag (seed) associated with each stance.

The classification in the proposed method is carried out as follows: (1) *Hashtag classification:* for each hashtag, we calculate the similarity between the hashtag and each seed. Then, the hashtag is assigned to class c where s_c has the highest similarity to the hashtag. The set of hashtags assigned to class c

is denoted by $\mathcal{C}_c$. *User classification:* For each user, based on the class of the hashtags shared by the user, we determine the user's inclination towards each class. The user is assigned to class c when they exhibit the highest inclination towards the class. The method comprises four steps described in the following sub-sections.

2.1 Generation of a hashtag-sharing graph

To construct the hashtag-sharing graph, sharing records are projected onto a graph where nodes represent hashtags and edges denote relationships between pairs of hashtags. This graph is defined using a matrix $A_{m \times m}$ where A_{ij} denotes the strength of the relationship between hashtags h_i and h_j. The value of A_{ij} is computed using Eq. (1).

$$A_{ij} = \frac{\sum_{u_k \in U} \min(R_{ki}, R_{kj})}{\min(\mathcal{R}_i, \mathcal{R}_j)} \tag{1}$$

The value A_{ij} falls within the range of $[0, 1]$, with a larger value indicating a stronger relationship.

2.2 Determination of similarity between hashtags and seeds

We apply a modified local random walk algorithm to determine the similarity between each hashtag $h_i \in H$ and each seed s_c (where $c \in \{1, \cdots, t\}$). In a local random walk algorithm, given a transition matrix A (where A_{ij} is regarded as the probability that a random walker will go to h_j from h_i), a walker starts walking from one node and continues traversing the graph until a specified stopping criterion is met. The general equation of the random walk is as Eq.(2):

$$\pi_c(z) = A^T \cdot \pi_c(z-1) \tag{2}$$

where $\pi_c(z)$ is an $m \times 1$ vector in which the i-th element, denoted by $\pi_{ci}(z)$, represents the probability of visiting node h_i in the z-th step of the walk. The notation A^T indicates the transpose of the transition matrix, and $\pi_c(0)$ is a vector with all zero values except the c-th element, which is 1.

In our modified random walk algorithm, a walker starts from a seed hashtag s_c and traverses the graph to calculate the similarity between each hashtag and the seed. However, to avoid assigning undue similarity between the given hashtag s_c and other hashtags strongly related to other seed hashtags, the walker should be prevented from visiting hashtags with strong relationships to other seeds. This visiting may occur in two cases if a walker starting from s_c: (i) visits a different seed s_j and then proceeds to hashtags strongly related to s_j, or (ii) visits a general hashtag, which is used by individuals from different stances (e.g., the hashtag *#ge2019* in the context of the 2019 UK general election), and then proceeds to hashtags strongly related to other seeds. To address case (i), we set all the values in the corresponding row and column of $j \in \{1, \cdots, t\} - \{c\}$ in the transition matrix to zero to prevent the walker from

visiting hashtags through seed s_j, ensuring that s_c is the only seed involved in the walk. To address (ii), we utilise the notation of entropy to adjust the transition probabilities through every hashtag h_i, considering the generality of the hashtag across various stances. For this purpose, we first compute an entropy E_i, as shown in Eq. (3). The value of E_i falls within the range of $[0, 1]$, with a larger value indicating that the hashtag h_i is more neutral.

$$E_i = \frac{-\sum_{c=1}^{t} \frac{A_{ic}}{\mathscr{A}_i} \cdot log(\frac{A_{ic}}{\mathscr{A}_i})}{log(t)} \tag{3}$$

In Eq. (3), $\mathscr{A}_i$ indicates the sum of the relationships between h_i and all seed hashtags. Then, the value A_{ic} is decreased as $A_{ic} = (1 - E_i)/d(i) \cdot A_{ic}$, where d_i is the degree of the node h_i in the hashtag-sharing graph (the number of non-zero values in the i-th row of the matrix $A_{m \times m}$).

The random walk approach is iterated for each seed hashtag s_c, to calculate the similarity between hashtags and seeds. In each iteration, the matrix A is row-normalised to ensure that the sum of values in each row is 1. Then, a walker traverses the graph by starting from a seed s_c. In a random walk approach, the similarity of every hashtag $h_i \in H$ to s_c can be calculated using Eq. (4).

$$S_{ci} = \sum_{z=1}^{\rho} \pi_{ci}(z) \tag{4}$$

In this equation, $\pi_{ci}(z)$ represents the i-th element of vector $\pi_c(z)$, which can be calculated using Eq. (2), and ρ denotes the number of steps in the random walk approach. To balance between running time and classification performance, we empirically set this value to 10 in our experiments.

2.3 Classification of Hashtags

After determining the similarity between hashtags and all seeds, each hashtag $h_i \in H$ is assigned to class c where the hashtag has the largest similarity value to the seed s_c. We use the notation $\mathcal{C}_c$ to refer to the set of hashtags assigned to class c. Additionally, we determine the stance intensity of each hashtag h_i as shown in Eq. (5).

$$\mathcal{I}(h_i) = \frac{-\sum_{c=1}^{t} \frac{s_{ci}}{i} \cdot log(\frac{s_{ci}}{i})}{log(t)} \tag{5}$$

where denotes the sum of the similarity values between h_i and all seeds. The stance intensity $\mathcal{I}(h_i)$ falls within the range of $[0, 1]$, where a larger value indicates a greater stance intensity.

2.4 Classification of Users

In this step, the stance and intensity of the hashtags used by each user u_k are utilised to determine the user's inclination towards each class. The user

is assigned to class c where they exhibit the greatest inclination towards the hashtags in $\mathcal{C}_c$. The inclination of u_k towards class c is calculated using Eq. (6).

$$\mathcal{L}_{kc} = \sum_{h_i \in \mathcal{C}_c} \mathcal{I}(h_i) \cdot \frac{R_{ki}}{\mathscr{R}_k} \tag{6}$$

In this equation, $\mathscr{R}_k$ represents the total number of times u_k has shared hashtags, and R_{ki} represents the number of times user u_k has shared hashtag h_i. The value of $\mathcal{L}_{kc}$ ranges form 0 to 1, with a larger value indicating a stronger inclination of u_k towards class c.

3 Experimental Evaluation

3.1 Datasets

To assess the effectiveness of the proposed method, we employ two Twitter (X) datasets collected via Twitter Streaming APIs. For each dataset, initially, we compile a list of the most frequently used hashtags in the relevant context and collect tweets containing at least one of these hashtags.

(i) The UK General Election Dataset (*Election* dataset) covers the UK general election held on 12 December 2019. This dataset was collected over four weeks, from 16 November 2019 to 12 December 2019, and includes 234,800 unique hashtags shared by 929,441 unique accounts, totalling 14,233,501 sharing records. In this dataset, the five major political parties are treated as distinct classes: Labour (LAB), Conservative (CON), Liberal Democrats (LDM), Scottish National (SCN), and Green (GRE). For each party, a hashtag that explicitly represents the party's stance is selected as a seed. The chosen seeds are #votelabour, #voteconservative, #votelibdem, #votesnp, and #votegreen for the LAB, CON, LDM, SCN, and GRE classes, respectively.

(ii) Brexit Referendum Dataset (*Brexit* dataset) pertains to the Brexit referendum, held on 23 June 2016, where the UK voted to decide whether to leave or remain in the European Union. This dataset was collected over four weeks, from 27 May 2016 to 23 June 2016. It includes 158,673 unique hashtags shared by 1,242,530 unique accounts, totalling 15,446,338 sharing records. In this dataset, two classes are considered: Remain (RMN) and Leave (LVE). The seed hashtags chosen for this dataset are #voteremain and #voteleave for the RMN and LVE classes, respectively.

For each dataset, we establish golden sets of users and hashtags. The golden set of users comprises accounts of members of parliament, candidates, or government ministers known for supporting each stance present in the collected dataset, acting as representatives of their respective classes. A hashtag qualifies as a golden hashtag for a class if it is shared at least 30 times by more than one account in that class and at least five times more frequently than by accounts in

any other class. These criteria ensure the establishment of a reasonable golden hashtag set with a practical size. Note that the golden sets of users and hashtags are used exclusively for evaluation purposes. Table 1 summarises the number of golden users and hashtags identified for evaluation in each stance (class) and dataset. During the experiments on both datasets, hashtags with low engagement are filtered out by excluding those with sharing frequencies lower than the average frequency of all hashtags. Similarly, users with sharing records fewer than the average sharing records of all users are also removed.

3.2 Setting

We compare our proposed LRM method with the following five methods: (i) The Iterative Classification Method (ICM) [4] which uses iterative classification to determine the stance of hashtags and users. (ii) The Hashtag Similarity Method (HSM) [5], originally designed for user classification using seed users, has been adapted in this paper due to the lack of seed users. It now classifies hashtags based on similarity to seed hashtags. (iii) Random Method (RDM) that randomly assigns hashtags and users to classes. (iv) The Label Propagation Method (LPM) which constructs a graph of hashtag co-occurrences and uses label propagation to assess the similarity between each hashtag and seeds. (v) Simple Random Walk (SRM) that generates a graph of hashtag co-occurrences and applies a random walk to measure the similarity to the seeds. ICM uses iterative classification for both hashtags and users, while RDM assigns them randomly. HSM, LPM, and SRM classify hashtags based on similarity to seeds. For user classification, these methods assign users to the class with the most shared hashtags.

Table 1: The golden sets (users and hashtags)

Dataset	Election dataset					Brexit dataset	
Class	LAB	CON	LDM	SCN	GRE	RMN	LVE
User	349	446	602	104	367	445	145
Hashtag	80	109	53	29	32	36	28

Three experiments were conducted for evaluation. The first assesses performance in classifying golden users and hashtags using F_1 scores for each class and F_{Macro} score as an indicator of overall performance. The second evaluates method efficiency based on average running time. Lastly, the third uses the proposed method to track the evolution in the weeks leading up to the events (election/referendum).

3.3 Results

In the first experiment, each method is applied to classify the users and hashtags using the seed hashtags for both datasets; and evaluated using the golden users and hashtags sets. Table 2 shows the results in terms of F_1 and F_{Macro} for the Election dataset, whilst Table 3 summarises the results for the Brexit dataset.

The best performance in each class (F_1) and across all classes (F_{Macro}) is shown in **bold face** in these tables.

Table 2: Classification performance in the Election dataset

Score	Hashtag Classification						User Classification					
	ICM	HSM	RDM	LPM	SRM	LRM	ICM	HSM	RDM	LPM	SRM	LRM
F_1 (LAB)	**0.919**	0.710	0.225	0.609	0.741	0.909	0.944	0.659	0.248	0.569	0.827	**0.975**
F_1 (CON)	**0.913**	0.870	0.249	0.688	0.850	**0.913**	0.911	0.891	0.237	0.532	0.861	**0.969**
F_1 (LDM)	0.901	0.516	0.191	0.351	0.739	**0.913**	0.831	0.367	0.198	0.140	0.811	**0.938**
F_1 (SCN)	0.897	**0.981**	0.134	0.963	0.732	0.912	0.852	**0.944**	0.069	0.727	0.459	0.897
F_1 (GRE)	0.867	0.595	0.138	0.514	0.639	**1.000**	0.826	0.022	0.173	0.011	0.765	**0.956**
F_{Macro}	0.899	0.734	0.188	0.625	0.740	**0.929**	0.873	0.577	0.185	0.396	0.745	**0.947**

Table 3: Classification performance in the Brexit dataset

Score	Hashtag Classification						User Classification					
	ICM	HSM	RDM	LPM	SRM	LRM	ICM	HSM	RDM	LPM	SRM	LRM
F_1 (RMN)	0.909	0.750	0.541	0.293	0.933	**0.941**	0.907	0.580	0.617	0.071	0.905	**1.000**
F_1 (LVE)	0.880	0.767	0.432	0.613	0.878	**0.917**	1.000	0.446	0.280	0.337	0.258	**1.000**
F_{Macro}	0.895	0.758	0.487	0.453	0.906	**0.929**	0.954	0.513	0.448	0.204	0.582	**1.000**

Table 4: Running time of the different methods for both datasets (in minutes)

Dataset	ICM	HSM	RDM	LPM	SRM	LRM
Election dataset	144.03	20.18	02.20	57.23	38.51	26.02
Brexit dataset	60.08	10.51	01.15	44.15	33.01	12.10

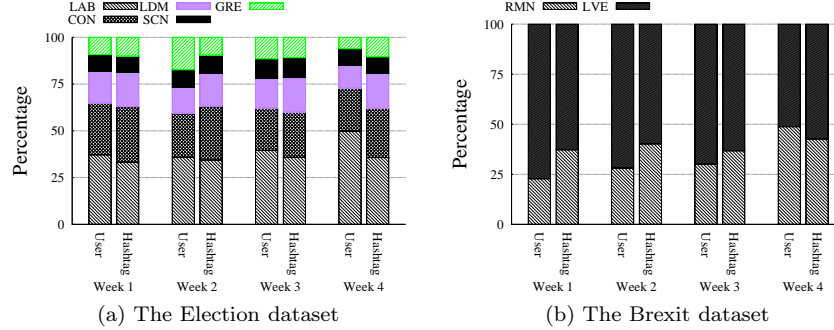(a) The Election dataset   (b) The Brexit dataset

Figure 1: Percentage of users and hashtags in each class

Regarding F_{Macro} scores, the proposed method (LRM) outperforms the other methods, with ICM providing the second-best results. Notably, SRM using a simple random walk also exhibits competitive performance, suggesting that employing the random walk approach is effective for calculating hashtag similarities for the task outlined in this paper. However, the modifications implemented in the random walk approach in LRM further enhance classification

performance. The difference between the F_{Macro} scores obtained by LRM and those of ICM (the second-best method) indicates that LRM exhibits greater superiority in user classification compared to hashtag classification. This observation underscores the effectiveness of our heuristic, which leverages hashtag stance intensity for user classification, enhancing performance. Regarding per-class F_1 scores, the LRM method consistently outperforms other methods, with few exceptions, where it shows comparable performance. The worst performance of LRM is in the Election dataset for users and hashtags in the *SCN* class. This may be because *SCN* is a relatively small class when compared to the other classes (see Table 1), and the proposed method may not properly determine the similarity between the hashtags in this class and the associated seed hashtag.

The second experiment evaluates the efficiency of methods by measuring their running time (Table 4). LRM demonstrates superior efficiency compared to LPM, SRM, and ICM. This efficiency stems from LRM's approach of constructing the hashtag-sharing graph based solely on user hashtag-sharing frequencies, unlike LPM and SRM, which process all tweets to find hashtag co-occurrences for graph generation. The iterative processing of posts by ICM renders it the slowest method. While RDM and HSM are faster than LRM, they exhibit significantly lower classification performance, as observed in the first experiment.

In the third experiment, we employ LRM to analyse the evolution of the number of users and hashtags in each class over the four weeks leading up to the event (election/referendum). To conduct this analysis, we extract data collected in each week and utilise LRM to classify users and hashtags. Figure 1 illustrates the percentage of all users and hashtags in each class over the four weeks. In the Election dataset (Figure 1a) it is observed that *LAB* consistently maintains a higher percentage of users and hashtags compared to other classes, with *CON* following closely. Interestingly, even in the week preceding the election (week 4), the percentage of *LAB* users and hashtags remains higher than that of *CON*. This observation aligns with reports suggesting greater engagement with Labour accounts on social media platforms [7]. In the Brexit dataset (Figure 1b), the activity of users supporting *LVE* is higher than that of users advocating for *RMN*. However, as the referendum date approaches, the percentage of *RMN* users and hashtags increases. By week 4, the percentage of users advocating *RMN* and *LVE* is 48.80% and 51.20%, respectively, closely reflecting the outcome of the referendum and findings from research predicting the outcome based on Twitter post analysis [8, 9]. We further analyse the data specifically on the day of the referendum, 23 June 2019, to ascertain the percentage of users advocating each stance. The results indicate a higher percentage of users advocating for *RMN* (59.59%). This finding aligns with previous research, that shows a similar trend: a large engagement of the *LVE* campaign since the early stages of the referendum process, with *RMN* catching up at a later stage [8].

4   Conclusion

This paper introduced a novel lightweight method for understanding public opinions on controversial topics, discovering the hashtags used to advocate each stance, and tracking the evolution over time. The method requires only one hashtag representing each stance and relies solely on the hashtags shared by the users. Unlike methods that rely on text analysis techniques or user interactions, our approach is language-agnostic and can be applied across various social networks without requiring user interactions. It also aligns with a real-world scenario, where a social scientist or other stakeholders would be interested in monitoring a given event, while they can easily identify seed hashtags. Experimental results demonstrate that our method outperforms similar methods and baselines.

Acknowledgment

References

[1] Aldayel, A., Magdy, W.: Your stance is exposed! analysing possible factors for stance detection on social media. Proceedings of the ACM on Human-Computer Interaction **3** (2019)

[2] ALDayel, A., Magdy, W.: Stance detection on social media: State of the art and trends. Information Processing & Management **58**(4), 102597 (2021)

[3] Alturayeif, N., Luqman, H., Ahmed, M.: A systematic review of machine learning techniques for stance detection and its applications. Neural Computing and Applications **35**(7), 5113–5144 (2023)

[4] Coletto, M., Lucchese, C., Orlando, S., Perego, R.: Polarized user and topic tracking in twitter. In: Proceedings of the 39th International Conference on Research and Development in Information Retrieval. p. 945–948 (2016)

[5] Darwish, K., Magdy, W., Zanouda, T.: Improved stance prediction in a user similarity feature space. In: Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. p. 145–148 (2017)

[6] Darwish, K., Stefanov, P., Aupetit, M., Nakov, P.: Unsupervised user stance detection on twitter. Proceedings of the International AAAI Conference on Web and Social Media **14**(1), 141–152 (2020)

---

[1]https://www.vigilantproject.eu

[7] Fletcher, R.: Did the conservatives embrace social media in 2019. UK election analysis pp. 1–123 (2019)

[8] Grčar, M., Cherepnalkoski, D., Mozetič, I., Kralj Novak, P.: Stance and influence of twitter users regarding the brexit referendum. Computational social networks **4**, 1–25 (2017)

[9] Khatua, A., Khatua, A.: Leave or remain? deciphering brexit deliberations on twitter. In: 2016 IEEE 16th international conference on data mining workshops. pp. 428–433 (2016)

[10] Kobbe, J., Hulpuș, I., Stuckenschmidt, H.: Unsupervised stance detection for arguments from consequences. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. pp. 50–60 (2020)

[11] Mohammad, S.M., Sobhani, P., Kiritchenko, S.: Stance and sentiment in tweets. ACM Transactions on Internet Technology **17**(3) (2017)

[12] Samih, Y., Darwish, K.: A few topical tweets are enough for effective user stance detection. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. pp. 2637–2646 (2021)

[13] Williams, E.M., Carley, K.M.: Tspa: Efficient target-stance detection on twitter. In: 2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 242–246 (2022)

[14] Zhou, L., Zhou, K., Liu, C.: Stance detection of user reviews on social network with integrated structural information. Journal of Intelligent & Fuzzy Systems **44**(2), 1703–1714 (2023)