Few-shot Learning for Cross-Target Stance Detection by Aggregating Multimodal Embeddings

Parisa Jamadi Khiabani and Arkaitz Zubiaga
Queen Mary University of London, London, UK

Abstract—Despite the increasing popularity of the stance detection task, existing approaches are predominantly limited to using the textual content of social media posts for the classification, overlooking the social nature of the task. The stance detection task becomes particularly challenging in cross-target classification scenarios, where even in few-shot training settings the model needs to predict the stance towards new targets for which the model has only seen few relevant samples during training. To address the cross-target stance detection in social media by leveraging the social nature of the task, we introduce CT-TN, a novel model that aggregates multimodal embeddings derived from both textual and network features of the data. We conduct experiments in a few-shot cross-target scenario on six different combinations of source-destination target pairs. By comparing CT-TN with state-of-the-art cross-target stance detection models, we demonstrate the effectiveness of our model by achieving average performance improvements ranging from 11% to 21% across different baseline models. Experiments with different numbers of shots show that CT-TN can outperform other models after seeing 300 instances of the destination target. Further, ablation experiments demonstrate the positive contribution of each of the components of CT-TN towards the final performance. We further analyse the network interactions between social media users, which reveal the potential of using social features for cross-target stance detection.

Index Terms—stance detection, social media, multimodal, classification.

I. INTRODUCTION

IN the information-driven world we now inhabit, a large amount of opinion texts can be found on the Web. The presence of this content is ever growing as social networking platforms become increasingly popular, which according to recent statistics are used by around 65% of American adults [1], attracting a great deal of public attention [2]. However, given the volume of content posted daily, monitoring the opinions expressed in social media platforms remains a time-consuming task which is not manageable without the support of automated means [3], [4]. Hence, there is a need to develop novel methods for automated classification and processing of these texts to determine the stance expressed in texts with the aim of mining public opinion.

Stance classification is concerned with identifying a person's or a post's standpoint towards a target [5], which is generally classified as one of in favor of (supporting) or against (opposing) the target in question [6], [7], [8], [5], [9], [10]. Stance classification from social media data is however a challenging task [11], [12], given the diverse and informal nature of social media data. Despite recent progress in stance classification, there is still substantial room for improvement,

particularly when it comes to enabling generalisability of classifiers to deal with new targets [13]. This is the case in cross-target stance classification; where a classifier has seen training data associated with particular targets but needs to predict the stance towards new targets, of which the model has seen no or few instances. An ability to deal with new targets is however important, given the evolving nature of the targets intended for the analysis. For example, an interest for mining stances towards US President Donald Trump can eventually shift towards Joe Biden as the new president takes office.

Previous research in cross-target stance classification has introduced approaches that leverage the textual content of posts, generally by using transfer learning strategies [14], [12]. Previous research has however been limited to the sole use of the textual content of posts to determine the stance, hence overlooking the potential of other features inherent to social media. Following the intuition of the theory of homophily, which suggests that like-minded users will tend to follow each other and like each other's posts, we therefore propose further digging into network features for cross-target stance detection. Our objective here is both to demonstrate that network information can be uniquely valuable to enhance the cross-target stance detection task as well as to propose a novel methodology that effectively does so. We propose a novel method, CT-TN, which encapsulates text and network features through a proposed architecture that aggregates both feature types for improved stance classification. Our aim here, in line with much of the previous work, is to use a small number of instances from the destination target in the training phase through few-shot learning.

To the best of our knowledge, CT-TN is the first multimodal architecture for cross-target stance classification which combines text and network features. By experimenting on six different combinations of source and destination targets in a few-shot cross-target stance detection scenario, we demonstrate the effectiveness of CT-TN to consistently outperform two state-of-the-art cross-target stance detection models as well as a state-of-the-art pre-trained language model, RoBERTa.

Contributions. Through our study, we make the following key contributions:

- We propose CT-TN (Cross-Target Text-Net) model, a model that encapsulates multimodal embeddings by integrating state-of-the-art text and graph embedding strategies for the cross-target stance classification task.
- We investigate the effectiveness of CT-TN in the few-shot cross-target stance detection task by using the P-Stance dataset, one of the largest stance datasets which enables

experimenting with combinations of six different source and destination target pairs.

- We perform ablation experiments to investigate the impact of the different components that form CT-TN, assessing whether and the extent to which each of them is contributing positively to improved performance of the model.
- In addition to our initial experiments with 400 shots from the destination target used for training, we further investigate scenarios where fewer shots are available, investigating its impact on the CT-TN model, and assessing how many shots the model needs to perform competitively.

We find that our model can consistently outperform state-of-the-art text-based cross-target stance detection models such as TGA-Net and CrossNet. Ablation experiments with CT-TN alternatives demonstrates the contribution of its different components. While we demonstrate the effectiveness of CT-TN, we also observe that it becomes less effective when we dramatically reduce the number of shots used during training, suggesting that the contribution of carefully integrated network features becomes useful after 300+ shots, but is less reliable when fewer shots (100 or 200) are available.

Paper structure. The remainder of this paper is organised as follows: Section II discusses work related to ours looking at the challenges of cross-target stance detection as well as the use of multimodal embeddings for stance detection. We introduce our proposed method CT-TN in Section III, followed by the experiment settings which we describe in Section IV. We present our experiments results in Section V as well as we further discuss and delve into them in Section VI. We conclude the paper in Section VII.

II. RELATED WORK

We discuss related work in two main directions: research on cross-target stance detection and the use of multimodal embeddings in stance detection.

A. Cross-target Stance Detection

Despite a substantial body of research in stance detection in recent years [15], [6], [16], the more challenging task of cross-target stance detection has received less attention. One of the first approaches to cross-target stance detection is **Bicond** [17], which combined multiple layers of LSTM models in different settings encoding the texts from left to right and from right to left. [18] developed **CrossNet**, which added an Aspect Attention Layer to the Bicond model, which enabled discovering domain-specific aspects for cross-target stance inference, utilising self-attention to signal the core parts of a stance-bearing sentence. Their model consists of four main layers: Embedding Layer, Context Encoding Layer, Aspect Attention Layer, and Prediction Layer. Their model showed to outperform the Bicond model.

A different type of approach focused on transferable topic modelling. This is the case of the **VTN** model [14]. This model uses shared latent topics between two targets as transferable knowledge in order to achieve model adaptation. The latent topics are determined by using Neural variational inference [19]. Another model by [12], called **SEKT**, proposed to leverage external knowledge to perform stance detection across targets. Still limited to processing textual content, they proposed to generate a semantic-emotion heterogeneous graph (SE-graph) which is fed to a GCN and a BiLSTM for the classification.

One of the best-known models among those presented recently is Topic-Grouped Attention (**TGA**), introduced by [20], and is one of our key baseline models. The model consists of four main phases: (i) Contextual Conditional Encoding, i.e. using contextual emebddings like BERT to embed a document and topic together, (ii) Generalized Topic Representations (GTR), i.e. using Ward hierarchical clustering, (iii) Topic-Grouped Attention, i.e. using learned scaled dot-product attention (compute similarity scores), and (iv) Label Prediction, i.e. feed-forward neural network to compute the output probabilities. This model proved competitive in comparison with a range of other baseline models, showing greater generalisability across different targets than other models.

However, existing approaches are limited to processing the textual part of the posts only for the cross-target stance detection. Here, we further design this research by proposing the first model that leverages network features in addition to text, through our proposed model CT-TN. To the best of our knowledge, this is the first study that takes advantage of using network/social information along with text into the cross-target stance detection task. In our experiments, in addition to CT-TN, we also experiment with TGA-Net and CrossNet as competitive baseline models.

B. Multimodal Embeddings and Stance Detection

Text and Graph Embeddings. There has been a substantial body of research in recent years in developing embedding approaches that deal with either texts or graphs separately.

When it comes to **text-based embeddings**, non-contextualised representations such as Word2vec [21] and GloVe [22] were soon followed by more sophisticated, contextualised representations such as ELMo [23] and OpenAI's GPT [24]. The latter use unidirectional language models in order to learn general language representations, restricting the efficiency of the pre-trained models.

Recently, researchers spend more time on applying transfer learning by fine-tuning large pre-trained language models for downstream NLP/NLU tasks, including a small number of examples which achieves distinguish performance improvement regarding these tasks. Despite the fact that pre-trained language models are used for this approach, they suffer from a main limitation which is needing huge corpora for pre-training plus the high computational cost of days needed for training [11].

Transformer models [25], including the likes of BERT [26] have more recently become the state-of-the-art models for text representation in text classification. Models under this category include BERT, RoBERTa [27], XLM [28] and XLM-R [29]. In our work, we make use of the RoBERTa model as a component of CT-TN, as well as a baseline model when used on its own.

When it comes to **graph embeddings**, different approaches have been proposed which operate at the node, sub-graph or different levels of granularity. These types of model include unsupervised techniques like DeepWalk [30], a method based on random walks, utilizing local neighborhoods of a node. A more popular method for generating embeddings from graphs is Node2Vec [31], which consists of a flexible biased random walk procedure to explore networks, being one of the first Deep Learning attempts to learn from graph data. There are similarities between Deepwalk [30] and Node2Vec [31] in that they both maximise the probability of node co-occurrences in sampled random-walks approaches across the graph. However there is a difference based on how the random walks are sampled. The former uses unbiased random walks, whereas the latter biases the random walks using two random walk hyperparameters return parameter (p) and in-out parameter (q). The p parameter controls the likelihood of immediately revisiting a vertex; and the q parameter q is responsible to controlling the likelihood that the walk revisits a vertex's one-hop neighbourhood. Deepwalk can in fact be viewed as a special case of node2vec where p=q=1. Another line of research towards graph embedding methods focuses on supervised techniques that utilise custom neural network components that can operate directly on graphs. One of the methods in this line of research is that of the wave architecture [32], which has the advantage of propagating long-range information across a graph compared to the graph convolution architectures. In our case, however, our focus in this work is on unsupervised graph embedding methods, and use of wave based architectures and other supervised strategies is left for future work.

The key focus of our work is on leveraging a cross-target stance architecture that can effectively combine textual and network information together in comparison to state-of-the-art baseline models that focus solely on textual content. Hence, we leverage existing graph embedding models rather than building new architectures, with the aim of directly tackling our key objective. To this aim, we use PecanPy [33] as the component to implement graph embeddings, which is an optimised implementation of Node2Vec that makes it more efficient thanks to its paralellisation. We use PecanPy to extract embeddings from three different types of network information: followers, friends and likes.

Multimodal Embeddings in Stance Detection. Combining textual embeddings with graph embeddings has been studied in previous research, however barely in the context of stance detection. This is the case of [34], who proposed an approach to enrich a BERT transformer by incorporating knowledge graph embeddings trained from Wikidata. Through experiments on a book classification task, they showed that their method could outperform other text-only baseline methods.

Despite the inherently social nature of the stance detection task, the vast majority of the research has been limited to textual features and embeddings. The main exception to this is the work by [6], who demonstrated that social signals could also be helpful to predict the stance expressed by a user, suggesting that it could even be possible to predict the stance of a user who has not posted anything, solely based on their network. This work is however limited to in-target stance detection.

To the best of our knowledge, no work has studied the multimodal aggregation of textual and graph embeddings for cross-target stance detection. Through the introduction of CT-TN, we aim to propose the first approach that effectively achieves this in the cross-target stance detection task.

III. METHODOLOGY

A. Problem Formulation

We formulate the stance detection task as that in which each of the posts in a collection $P = \{p_1, p_2, ..., p_n\}$ has to be classified into one of the stances $S = \{favor, against\}$, where each post p_i expresses a stance towards a target t. We have a dataset where each post expresses a stance towards one of the targets in a collection $T = \{t_1, t_2, ..., t_m\}$. The cross-target stance detection task consists in predicting the stance expressed in posts referring to target t_i, where the training data is composed of posts referring to other targets excluding t_i, hence requiring a transfer of knowledge from a set of targets to another. In the few-shot cross-target stance detection scenario, however, we experiment with a small number of instances referring to t_i incorporated into the training data, to relax this cross-target scenario. In our particular case, we experiment with 400 instances (i.e. 400 shots) pertaining to t_i incorporated into the training data in the few-shot scenario; in subsequent experiments, we test with smaller numbers of instances in the few-shot scenario, namely 100, 200 and 300.

B. Proposed Method: CT-TN

The CT-TN architecture consists of three main types of components: (i) text-based embedding generation and classification, (ii) three instances of network encoding components for graph-based embedding generation and classification, which are used for feeding followers, friends and likes, and (iii) output aggregation. The first two types of components are executed in parallel to produce isolated stance predictions, after which the final component aggregates the predictions for final output. Figure 1 demonstrates general architecture of our proposed model. In what follows we describe the specifics of these three components.

Component #1: Text-based classification: Contextual Conditional Encoding. This component takes in the textual content of the input posts using bidirectional conditional encoding (conditioning the document representation on the topic) layer followed by a feed-forward neural network. Previous works have shown the advantage of utilising contextual embeddings [26]. We embed the user generated text through the RoBERTa language model [27] to embed a document and topic jointly (768 dimension vector). RoBERTa can take as input either one or two sentences, and uses the special token [SEP] to separate them. In order to input both the textual content and target information associated with the post, we feed the following to the model: "[CLS] + target + [SEP] + context". This component produces an output with its own prediction for the stance of a particular post, as either supporting or opposing.
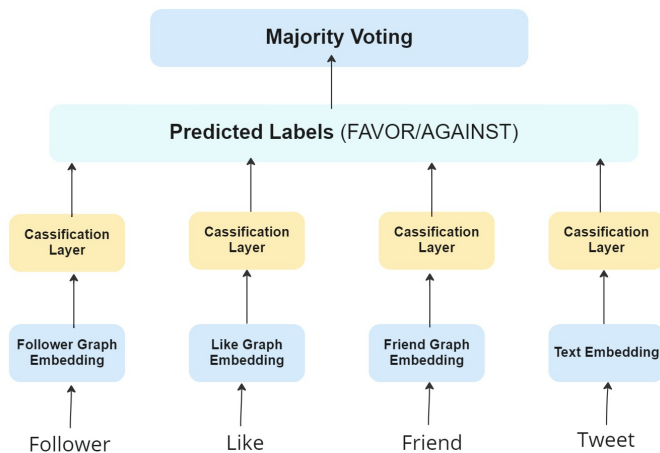
Fig. 1. Architecture of the proposed model, CT-TN.

Component #2: Graph-based classification: Network Encoding. The CT-TN model uses three different instances of the network encoding model for graph-based classification, for representing three types of inputs: followers, friends and likes. To generate embeddings using the Node2Vec architecture [31], we use the PecanPy implementation [33] which optimises its performance. The node embeddings calculated using PecanPy (128 dimension vector) can be used as feature vectors in a downstream task such as node classification. In our case, user IDs are considered as graph nodes and the relationships between users (friends/followers/likes) are provided as graph edges. Each of the three components implemented here through network encodings produce their own predictions on a particular post (supporting or opposing).

Component #3: Output aggregation. The final component takes as input the predictions made by all the above components, i.e. the text-based and three network-based components. To aggregate the predictions of all these four components, the "output aggregation" component implements a voting ensemble (or a "majority voting" strategy) which combines the different predictions, ultimately choose the class with a larger number of votes. While we did test alternative methods for combining different embedding inputs, such as concatenation of embeddings, both with and without normalisation of the input embeddings, the 'majority voting' strategy proved clearly more effective in initial experiments and hence was the option we finally selected to develop CT-TN.

C. Model Hyperparameters

We use the RoBERTa base model (roberta-base-cased) as our pre-trained language model, due to its improved performance over similar transformer models such as BERT [35]. It consists of 12 transformer layers, each of which adopts a hidden state size of 768 and 12 attention headers. Training for RoBERTa text embedding is performed with batch size $b = 128$, dropout probability $d = 0.2$, learning rate= 3e-5 (AdamW optimiser) and 40 training epochs. While we trained graph embedding models as follow: batch size $b = 128$, dropout probability $d = 0.2$, learning rate= 1e-2 (SGD optimiser) and

100 training epochs. For model training, we use multi-class cross-entropy loss.

While previous research has addressed the stance detection task in both 3-class [36], [37] and 2-class [38], [39] settings, our focus here is on the latter, while the extension of our proposed model to 3-class is beyond the scope of this paper.

IV. EXPERIMENTS

Next, we provide the details of the dataset we use in our research, as well as the baseline methods we compare our method against, followed by experiment settings and evaluation metrics used in our work.

A. Dataset

We chose to use the P-Stance dataset [13], given that it is an order of magnitude larger than other publicly available datasets and because it provides more than one target, as required for our research in cross-target stance detection. The P-Stance dataset is originally composed of tweets pertaining to three different political figures as targets: "Donald Trump," "Joe Biden," and "Bernie Sanders." A sample of the P-Stance dataset is provided as Table I.

The original P-Stance dataset, as published by the authors, only contains the tweet texts associated with their stance labels. This original dataset lacked the network information that we needed. Upon request, the authors kindly shared 9,307 tweet IDs, which we use to reconstruct and expand the dataset. This includes retrieving full tweet metadata, from which we can extract the user IDs, which would then allow us to retrieve the user network.

Given the focus of our research in 2-class classification (i.e. favor or against), we retrieve metadata for the tweets associated with these categories. This led to 4,212 tweets with available metadata. The resulting dataset has a distribution of labels as shown in Table II, and a distribution in the number of tweets per target as shown in Figure 2. While the number of tweets across targets is very similar, we observe some differences in the number of favor and against tweets, with Donald Trump having the largest ratio of against tweets, and Bernie Sanders having the largest ratio of favor tweets. For these available tweets, we further complement the data retrieval as described below.
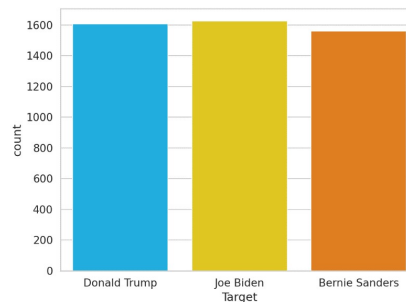

Fig. 2. Distribution of targets in the P-Stance dataset.

Having the collection of tweet IDs, tweet metadata and user IDs, we proceeded with the retrieval of additional data

TABLE I
A SAMPLE OF THE P-STANCE DATASET.

Tweet	Target	Stance
How Joe Biden would make community college free and fix student loans via @politico	Joe Biden	FAVOR
Glad our GREAT President called out the so called whistleblower. If there is a Senate trail, they may call the whistleblower to testify. BTW Trump is not impeached until crazy Nancy send the articles over to the Senate. Trump will not be convicted. Vote Trump 2020	Donald Trump	FAVOR
#Bernie Sanders says he's 'one of the poorer members of the #UnitedStatesSenate' #BetOil is A Multimillionaire,#Warren has A 5 Million dollar Home,#Hillary HAS several Mansions plus A Super Millionaire! Whats Your Point?	Bernie Sanders	AGAINST

TABLE II
THE STATISTICS OF THE RESULTING DATASET.

Target	Favor	Against	Avg. length
Donald Trump	519	947	34.7
Joe Biden	702	716	33.7
Bernie Sanders	776	553	31.5

including networks of the users (followers and friends) and likes (tweets they liked from other users). We detail each of these additional data collection steps next:

- **Retrieval of followers:** Followers include the set of users who follow a particular user. For the user IDs in the dataset, we retrieve the complete list of followers for each user. This leads to a list of user IDs followed by each user, which allows us to build a network of followers.
- **Retrieval of friends:** Friends constitute the set of users followed by a user. Similar to the list of followers, this provides a list of user IDs per user, with which we can build a network.
- **Retrieval of likes:** For each user, we retrieve the tweets they liked from others. Given that we are interested in building networks of users, we obtain the user IDs pertaining to the tweets liked by the users. This again allows us to build a network, very similar to the friend / follower networks above, in this case based on the user IDs whose tweets have been liked by each user.

Hence, for each user, we have four features: (i) the textual content of their tweet, (ii) the network of followers, (iii) the network of friends, and (iv) the network of likes. Each of these is associated with a favor or against label, which we aim to predict.

After aggregating all these four features, we end up with a dataset of 4,144 tweets (posted by 3,871 distinct users) for which we have all features available.

B. Baseline Methods

We evaluate and compare our model with several strong baselines, including two of the main state-of-the-art cross-target stance detection models as well as the widely-used Transformer model RoBERTa:

- **CrossNet [18]:** This model is a variant of BiCond, which leverages a self-attention layer to capture important words in the input text.
- **TGA-NET [20]:** A new model has been proposed for (few-shot) cross-target stance detection that implicitly captures relationships between topics using generalized topic representations.
- **RoBERTa [27]:** The method fine-tunes a pre-trained BERT model to perform cross-target detection. Specifically, we convert the given context and target to "[CLS] + target + [SEP] + context" structure for source and target domain, respectively.

Note that all of the above baselines make use of the textual content of the posts, as opposed to our proposed CT-TN also incorporating network information.

C. Experiment Settings

Experiments for the proposed few-shot cross-target approach are conducted in 100-shot, 200-shot, 300-shot, and 400-shot settings (e.g. injecting N samples of destination target into (source-target based) train data and then predicting the stance on the test data consist of only destination target) with 5 different random seeds: 24, 524, 1024, 1524, and 2024. Then we average the 5 seeds' results per shot.

D. Evaluation Metrics

In line with previous research in stance detection [17], [18], [20], we also adopt the macro-averaged F1 score ($MacF_{avg}$) as the main metric to evaluate the performance in our experiments. In our case with binary classification involving the support and oppose classes, the resulting metric is the arithmetic mean of the F1 scores for each class, as follows:

$$MacF_{avg} = \frac{F1_{support} + F1_{oppose}}{2} \quad (1)$$

where each of $F1_{support}$ and $F1_{oppose}$ is defined as follows:

$$F1_c = \frac{2 * precision_c * recall_c}{precision_c + recall_c} \quad (2)$$

TABLE III
MACRO-AVERAGED F1 SCORES FOR CT-TN VS BASELINE MODELS.

| Source | Trump | Sanders | Sanders | Biden | Trump | Biden | Average |
Test	Sanders	Trump	Biden	Sanders	Biden	Trump	
RoBERTa	0.53	0.59	**0.78**	0.66	**0.77**	0.62	0.66
CrossNet	0.46	0.51	0.69	0.58	0.6	0.54	0.56
TGA-Net	0.57	0.6	0.69	0.6	0.69	0.59	0.62
CT-TN	**0.72**	**0.8**	0.78	**0.73**	0.77	**0.82**	**0.77**

V. RESULTS

We next present results of our proposed CT-TN model. We first discuss results of the model compared to a set of competitive baselines. We then delve into the results by analysing the performance of ablated versions of the model, and by looking at the impact of the number of shots used for training.

A. CT-TN vs Baselines

Table III shows the results of CT-TN for the six combinations of source-destination targets under consideration, compared with the baseline models RoBERTa, CrossNet and TGA-Net. In addition to the results for each of the target pairs, we also show the average performance of each model across all pairs.

We observe that CT-TN consistently outperforms both cross-target stance detection models, CrossNet and TGA-Net, when we look at each target pair independently as well as at the overall average absolute improvements of 0.21 and 0.15, respectively.

CT-TN also performs remarkably better for than RoBERTa for a number of target pairs, not least Trump-Sanders, Sanders-Trump and Biden-Trump, with absolute improvements of 19%, 21% and 20% respectively. This improvement is more modest for Biden-Sanders (7%), with similar performances for the Sanders-Biden and Trump-Biden (0%) target-pairs. On average, CT-TN still outperforms RoBERTa by 0.11, showing that it is more consistent across targets and overall more reliable. We believe that the strongest improvements of CT-TN with respect to the baselines come particularly for targets with significantly different ideology (i.e. those combining Trump and Sanders, and Trump and Biden); this suggests that for more distant targets, textual models may be more limited in capturing these substantial linguistic differences, whereas a network-based model generalises better in these situations.

B. Analysis of Results per Class

We next delve into the performance scores of the models broken down by category: favor and against. Table VI shows the results for the favor and against categories. We see that the improvement of the CT-TN model with respect to the baselines is consistent across both classes, hence showing that CT-TN provides a positive boost that impacts both classes positively. The extent of the improvement across classes is also consistent with the overall results shown above, as we see that the set of target pairs where CT-TN achieves the highest improvement

matches those with the highest improvement in the overall results, i.e. Trump-Sanders, Sanders-Trump and Biden-Trump. Overall, CT-TN achieves improvements of 12% in both the favor and against class over the second best model, RoBERTa.

TABLE IV
CONFUSION MATRICES OF CT-TN MODEL VS. ROBERTA MODEL FOR SIX TARGET PAIRS

Biden-Sanders		
CT-TN		
	Against	Favor
Against	133	47
Favor	63	156
RoBERTa		
	Against	Favor
Against	58	122
Favor	27	192

Sanders-Biden		
CT-TN		
	Against	Favor
Against	123	50
Favor	47	179
RoBERTa		
	Against	Favor
Against	53	120
Favor	51	175

Biden-Trump		
CT-TN		
	Against	Favor
Against	208	9
Favor	66	116
RoBERTa		
	Against	Favor
Against	207	10
Favor	132	50

Trump-Biden		
CT-TN		
	Against	Favor
Against	180	59
Favor	34	126
RoBERTa		
	Against	Favor
Against	183	56
Favor	43	117

Sanders-Trump		
CT-TN		
	Against	Favor
Against	179	10
Favor	71	139
RoBERTa		
	Against	Favor
Against	147	42
Favor	149	61

Trump-Sanders		
CT-TN		
	Against	Favor
Against	162	57
Favor	59	121
RoBERTa		
	Against	Favor
Against	44	175
Favor	36	144

To further delve into the predictions broken down by class and to assess the consistent effectiveness of CT-TN across classes, we look at the confusion matrices of predictions. In the interest of a focused analysis, we present confusion matrices for the text-only baseline model (RoBERTa) as well as our proposed CT-TN model that combines graph and textual embeddings (see Table IV. Looking at the pairwise comparison of predictions of CT-TN and RoBERTa for each

of the source-destination target pairs, we observe that CT-TN is generally better at making more accurate predictions for both the favor and against classes. Exceptions include the favor class in the Biden-Sanders pair, the against class in the Trump-Biden pair and the favor class in the Trump-Sanders pair. CT-TN is consistently better in all the other cases. A look at the aggregate confusion matrix combining all target pairs (see Table V) shows that CT-TN is clearly better overall in making predictions for both classes, with 985 correct 'against' predictions (vs 692 by RoBERTa) and 837 correct 'favor' predictions (vs 739 by RoBERTa).

CT-TN		
	Against	Favor
Against	985	232
Favor	340	837

RoBERTa		
	Against	Favor
Against	692	525
Favor	438	739

TABLE V
AGGREGATE CONFUSION MATRIX COMBINING PREDICTIONS ACROSS ALL SIX TARGET PAIRS.

C. Ablated versions of CT-TN

To evaluate the effectiveness of each of the text and network components of CT-TN, we perform a set of ablation of experiments with different sets of these components removed. Table VII shows the performance scores of the full CT-TN model compared with ablated versions of the model.

We can see that the full CT-TN model achieves top performance in five of the six source-destination target pairs, with the exception of the Sanders-Trump pair where the use of likes only outperforms the full model. For the rest of the target pairs, the full CT-TN either outperforms all ablated models or achieves the same performance as one of the ablated models. Interestingly, however, even if some of the ablated models perform at the same level as the full model on some occasions, there is no consistency on the best ablated model across target pairs. Given the uncertainty on the selection of the best ablated model in each case, it is more reliable to use the full CT-TN model instead, which is more consistent across all target pairs. Indeed, this consistency is also demonstrated in the highest average performance across targets, with an average 0.77 overall. Among the ablated models, those using the likes feature show competitive performance, with the model using only likes and the model combining likes, friends and followers both achieving second-best performance with an average of 0.76. This in turn suggests that, among the network features, the likes are the most useful ones.

D. Reducing the number of shots

Experiments so far have relied on the use of 400 shots associated with the destination target, showing competitive performance. We are further interested in investigating how CT-TN performs with fewer shots, as well as to assess the number of shots the model needs to outperforms other baseline models.

Table VIII shows the results for varying numbers of shots, ranging from 100 to 400. These results show a clear trend where the CT-TN model becomes remarkably effective with 300 shots used for training, after which it starts to outperform baseline models, generally by a margin. Conversely, results also show that using 200 or fewer shots is insufficient for CT-TN, where baseline models CrossNet and TGA-Net can perform better. Hence, CT-TN becomes especially reliable as the number of shots increases; however, performance scores are substantially lower for all models when the number of shots is 200 or fewer, hence suggesting that it is worth labelling some more instances up to 300 to achieve a substantial performance gain.
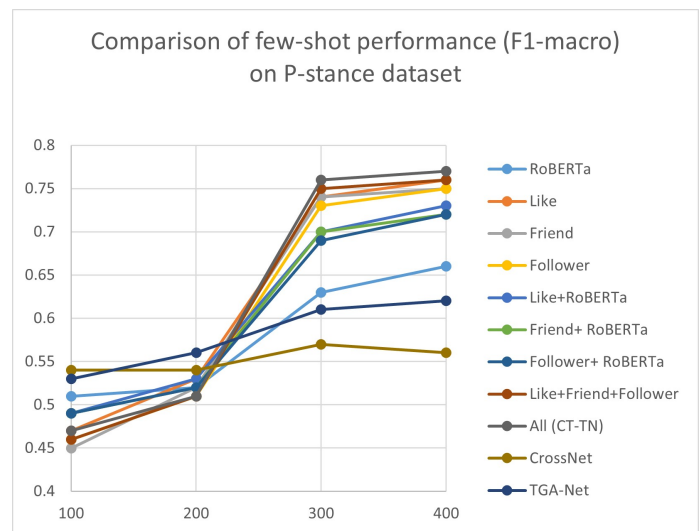

Fig. 3. The overall performance of 11 few-shot cross-target tasks.

Figure 3 shows the performance of the full CT-TN model, ablated models as well as baseline models with different numbers of shots used for training. In addition to the results presented in Table VIII, this figure enables additional visual comparison by also incorporating ablated models. These results reaffirm our previous observations, showing that it is especially after 300 shots that CT-TN and its ablated models become effective. All CT-TN based models achieve a remarkable gain of performance from 200 to 300 shots, which becomes less pronounced when shots are increased from 300 to 400.

VI. DISCUSSION

Through our experiments, CT-TN has proven to be a very competitive model achieving state-of-the-art performance when it is given a moderate number of training data associated with the destination target. In our case, we have seen that the model can outperform all other baselines with 300 instances pertaining to the destination target. We are however interested in further delving into the performance of CT-TN, which we do next by looking at some of its correct predictions as well as further investigating the structure of the network data it uses.

TABLE VI
MACRO-AVERAGED F1 SCORES FOR THE FAVOR AND AGAINST CLASSES WITH CT-TN VS BASELINE MODELS.

Source Test	Trump Sanders	Sanders Trump	Sanders Biden	Biden Sanders	Trump Biden	Biden Trump	Average
'Favor' class							
RoBERTa	0.64	0.5	0.78	0.72	0.72	0.47	0.64
CrossNet	0.4	0.47	0.68	0.54	0.6	0.49	0.53
TGA-Net	0.55	0.56	0.7	0.67	0.65	0.47	0.6
CT-TN	**0.69**	**0.78**	**0.79**	**0.74**	**0.75**	**0.78**	**0.76**
'Against' class							
RoBERTa	0.44	0.68	**0.77**	0.59	**0.8**	0.75	0.67
CrossNet	0.5	0.55	0.7	0.61	0.63	0.6	0.6
TGA-Net	0.59	0.61	0.65	0.54	0.73	0.7	0.64
CT-TN	**0.76**	**0.81**	0.76	**0.75**	**0.8**	**0.84**	**0.79**

TABLE VII
MACRO-AVERAGED F1 ON FULL CT-TN VS ABLATED VERSIONS OF CT-TN.
LI: LIKE, FR: FRIENDS, FL: FOLLOWERS, RB: ROBERTA.

Li	Fr	Fl	Rb	Trump Sanders	Sanders Trump	Sanders Biden	Biden Sanders	Trump Biden	Biden Trump	Average
			x	0.64	0.5	**0.78**	0.72	0.72	0.47	0.64
x				0.71	**0.83**	0.74	**0.73**	0.74	**0.82**	0.76
	x			0.7	0.79	0.75	0.71	0.75	0.81	0.75
		x		0.69	0.8	0.74	0.7	0.74	0.8	0.75
x			x	0.63	0.79	0.77	0.7	0.76	0.74	0.73
	x		x	0.63	0.71	0.76	0.69	**0.77**	0.73	0.72
		x	x	0.63	0.73	**0.78**	0.69	0.76	0.71	0.72
x	x	x		0.71	0.81	0.75	0.72	0.75	**0.82**	0.76
x	x	x	x	**0.72**	0.8	**0.78**	**0.73**	0.77	**0.82**	**0.77**

To better understand the benefits of CT-TN, we delve into some of the examples where CT-TN made a correct prediction and the baseline models made a wrong prediction. We show some of these CT-TN's correct predictions in Table IX. We observe that these are indeed difficult to predict solely from text for an automated model, not least because there are no explicitly positive keywords, often requiring more complex understanding of the text which is not trivial. In situations like these, information derived from the network through CT-TN can be particularly valuable to correct these otherwise challenging predictions.

Looking at the network data, Figure 4 shows a visualisation of the aggregate of follower, friend and like connections of supporters of each of the political candidates in the dataset, i.e. Bernie Sanders (blue), Joe Biden (green) and Donald Trump (red). Interestingly, we can observe three clear clusters of supporters of each candidate, with strong connections within clusters and fewer connections across clusters. Further, we can also observe that clusters associated with the two candidates of the Democrats, namely Joe Biden and Bernie Sanders, are closer and more strongly connected to each other than any of them is with Republican candidate Donald Trump's cluster. Through our experiments with CT-TN, we demonstrate that, while network information alone would not suffice to achieve top performance on stance detection, it is a valuable feature when used in combination with text, indeed outperforming any ablated models that solely use text or network data.

VII. CONCLUSION

To tackle the challenging task cross-target stance detection from social media posts, we have introduced a novel model, CT-TN, which aggregates multimodal text and network embeddings into a model. With a set of experiments across six different source-destination target pairs, we demonstrate the overall effectiveness of CT-TN, outperforming state-of-the-art models such as CrossNet and TGA-Net. While all models struggle with small numbers of shots used for training, CT-TN achieves a noticeable performance gain after 300 shots associated with the destination target are incorporated into the training data. In addition to showing the effectiveness of the novel CT-TN model, we also demonstrate the importance of considering network features for cross-target stance detection, among which the 'likes' feature leads to highest performance gains.

Our work in turn opens a set of avenues for future research. While we demonstrate that we can achieve competitive performance with 300+ shots, future work could look into further improving models that perform competitively when

TABLE VIII
MACRO-AVERAGED F1 SCORES ON MODELS USING DIFFERENT NUMBERS OF TRAINING SHOTS (100-400) FROM THE DESTINATION TARGET.

| | Trump → Sanders | | | | Sanders → Trump | | | |
	#100	#200	#300	#400	#100	#200	#300	#400
RoBERTa	0.24	0.28	0.49	0.53	0.31	0.35	0.54	0.59
CrossNet	0.41	0.46	0.48	0.46	**0.49**	0.45	0.5	0.51
TGA-Net	0.4	**0.47**	0.55	0.57	0.43	**0.48**	0.58	0.6
CT-TN	**0.44**	0.4	**0.68**	**0.72**	0.3	0.33	**0.78**	**0.8**
	Sanders → Biden				Biden → Sanders			
	#100	#200	#300	#400	#100	#200	#300	#400
RoBERTa	**0.76**	**0.77**	0.76	**0.78**	0.5	0.47	0.58	0.66
CrossNet	0.62	0.61	0.68	0.69	**0.57**	**0.56**	0.58	0.58
TGA-Net	0.61	0.65	0.69	0.69	**0.57**	**0.56**	0.59	0.6
CT-TN	0.59	0.73	**0.77**	**0.78**	0.38	0.41	**0.73**	**0.73**
	Trump → Biden				Biden → Trump			
	#100	#200	#300	#400	#100	#200	#300	#400
RoBERTa	**0.75**	**0.76**	**0.78**	0.77	0.5	0.5	0.61	0.62
CrossNet	0.6	0.58	0.62	0.6	**0.52**	**0.56**	0.57	0.54
TGA-Net	0.67	0.68	0.69	0.69	0.49	0.5	0.56	0.59
CT-TN	0.74	**0.76**	0.77	**0.77**	0.38	0.43	**0.8**	**0.82**

TABLE IX
SAMPLES WITH CORRECT PREDICTION ONLY BY CT-TN, WHERE BASELINE MODELS MISPREDICTED. EXAMPLES EXTRACTED FROM EXPERIMENTS FOR THE BIDEN-TRUMP TARGET PAIR.

Tweet	Real label	CT-TN	RoBERTa	CrossNet	TGA-NET
Guess we will have to wait forever!! Were working with heartless #CLOWNS her!! #TRUMP #Trump	FAVOR	FAVOR	AGAINST	AGAINST	AGAINST
What was Nancy Pelosi doing when realDonaldTrump was putting the #Coronavirus task force together? Handing out impeachment pens. Voting #Trump and red down the entire ticket!	FAVOR	FAVOR	AGAINST	AGAINST	AGAINST
People talk about the #GOP being the party of Lincoln and Reagan, well add realDonaldTrump to it because he is a game changing #POTUS with policy like this!	FAVOR	FAVOR	AGAINST	AGAINST	AGAINST

fewer shots available, which is particularly important where there are limited resources for labelling new data. Our research demonstrates the effectiveness of CT-TN for 2-class stance detection, while future research could further look into extending it to 3-class stance detection. While datasets enabling cross-target stance detection are very limited to date, not least datasets for which network data can be gathered, we hope to see more suitable datasets in the future, which would also enable further experiments using a bigger set of target pairs.

Our objective with CT-TN has been to demonstrate that the hybrid combination of graph-based embeddings using network interactions and text-based embeddings based on social media posts can lead to an effective approach for the cross-target stance detection task, ourperforming other state-of-the-art text-only models in the task. We do believe however that there is room for further improvement and investigation with other embedding methods, such as wave based architectures for graph representation [32]. Likewise, we chose to develop CT-TN by using a majority voting strategy to combine different embedding inputs, given its superior performance over other methods, such as concatenation of embeddings. We believe that there is room for further exploration with concatenation strategies to try and make them effective for the task, for example by investigating different normalisation methods for the embeddings.

REFERENCES

[1] A. Perrin, "Social media usage," *Pew research center*, vol. 125, pp. 52–68, 2015.
[2] L. Tian, X. Zhang, Y. Wang, and H. Liu, "Early detection of rumours on twitter via stance transfer learning," in *European Conference on Information Retrieval*. Springer, 2020, pp. 575–588.
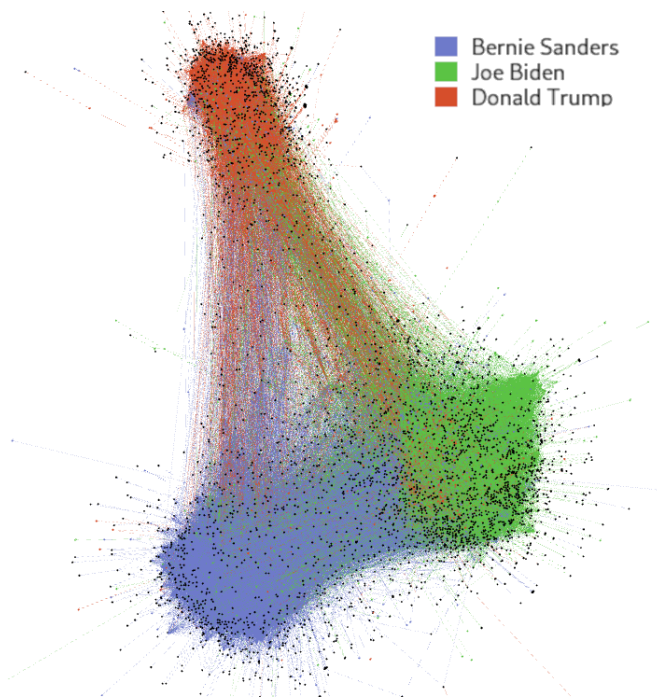
Fig. 4. Network visualisation of followers, friends and likes for users expressing supporting stance towards Bernie Sanders (purple), Joe Biden (green) and Donald Trump (red).

[3] B. Liu, "Sentiment analysis and opinion mining," *Synthesis lectures on human language technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[4] A. Zubiaga, "Mining social media for newsgathering: A review," *Online Social Networks and Media*, vol. 13, p. 100049, 2019.

[5] D. Biber and E. Finegan, "Adverbial stance types in english," *Discourse processes*, vol. 11, no. 1, pp. 1–34, 1988.

[6] A. AlDayel and W. Magdy, "Stance detection on social media: State of the art and trends," *Information Processing & Management*, vol. 58, no. 4, p. 102597, 2021.

[7] P. Jamadi Khiabani, M. E. Basiri, and H. Rastegari, "An improved evidence-based aggregation method for sentiment analysis," *Journal of Information Science*, vol. 46, no. 3, pp. 340–360, 2020.

[8] B. Liu *et al.*, "Sentiment analysis and subjectivity." *Handbook of natural language processing*, vol. 2, no. 2010, pp. 627–666, 2010.

[9] A. Zubiaga, R. Procter, and C. Maple, "A longitudinal analysis of the public perception of the opportunities and challenges of the internet of things," *PloS one*, vol. 13, no. 12, p. e0209472, 2018.

[10] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, 2016, pp. 31–41.

[11] W. Antoun, F. Baly, and H. Hajj, "Arabert: Transformer-based model for arabic language understanding," *arXiv preprint arXiv:2003.00104*, 2020.

[12] B. Zhang, M. Yang, X. Li, Y. Ye, X. Xu, and K. Dai, "Enhancing cross-target stance detection with transferable semantic-emotion knowledge," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 3188–3197.

[13] Y. Li, T. Sosea, A. Sawant, A. J. Nair, D. Inkpen, and C. Caragea, "P-stance: A large dataset for stance detection in political domain," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 2355–2365.

[14] P. Wei and W. Mao, "Modeling transferable topics for cross-target stance detection," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 1173–1176.

[15] D. Küçük and F. Can, "Stance detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 53, no. 1, pp. 1–37, 2020.

[16] R. Cao, X. Luo, Y. Xi, and Y. Qiao, "Stance detection for online public opinion awareness: An overview," *International Journal of Intelligent Systems*, 2022.

[17] I. Augenstein, T. Rocktäschel, A. Vlachos, and K. Bontcheva, "Stance detection with bidirectional conditional encoding," *arXiv preprint arXiv:1606.05464*, 2016.

[18] C. Xu, C. Paris, S. Nepal, and R. Sparks, "Cross-target stance classification with self-attention networks," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 778–783.

[19] Y. Miao, L. Yu, and P. Blunsom, "Neural variational inference for text processing," in *International conference on machine learning*. PMLR, 2016, pp. 1727–1736.

[20] E. Allaway and K. Mckeown, "Zero-shot stance detection: A dataset and model using generalized topic representations," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8913–8931.

[21] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[22] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[23] E. Peters Matthew, N. Mark, I. Mohit, G. Matt, C. Christopher, and L. Kenton, "Deep contextualized word representations.(2018)," 1802.

[24] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[26] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[27] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[28] G. Lample and A. Conneau, "Cross-lingual language model pretraining," *arXiv preprint arXiv:1901.07291*, 2019.

[29] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116*, 2019.

[30] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[31] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[32] M. K. Matlock, A. Datta, N. Le Dang, K. Jiang, and S. J. Swamidass, "Deep learning long-range information in undirected graphs with wave networks," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[33] R. Liu and A. Krishnan, "Pecanpy: a fast, efficient and parallelized python implementation of node2vec," *Bioinformatics*, vol. 37, no. 19, pp. 3377–3379, 2021.

[34] M. Ostendorff, P. Bourgonje, M. Berger, J. Moreno-Schneider, G. Rehm, and B. Gipp, "Enriching bert with knowledge graph embeddings for document classification," *arXiv preprint arXiv:1909.08402*, 2019.

[35] A. F. Adoma, N.-M. Henry, and W. Chen, "Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition," in *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*. IEEE, 2020, pp. 117–121.

[36] R. Xu, Y. Zhou, D. Wu, L. Gui, J. Du, and Y. Xue, "Overview of nlpcc shared task 4: Stance detection in chinese microblogs," in *Natural language understanding and intelligent applications*. Springer, 2016, pp. 907–916.

[37] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "A dataset for detecting stance in tweets," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 2016, pp. 3945–3952.

[38] A. Murakami and R. Raymond, "Support or oppose? classifying positions in online debates from reply activities and opinion expressions," in *Coling 2010: Posters*, 2010, pp. 869–875.

[39] R. Abbott, B. Ecker, P. Anand, and M. Walker, "Internet argument corpus 2.0: An sql schema for dialogic social media and the corpora to go with

it," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 2016, pp. 4445–4452.

Parisa Jamadi Khiabani is a PhD student at Queen Mary University of London. Her research background is mainly focused on Natural Language Processing and Social Media Analysis. Her main field of research is Stance Detection, Computational Linguistics, and Sentiment Analysis. She is passionate about interdisciplinary research and Social Media Analysis in order to gather and extract meaningful information from the posts/interaction between users of social platforms. She also has a journal publication, proposing an improved aggregation method to find a document's sentiment orientation of two review datasets.

Arkaitz Zubiaga is a senior lecturer (associate professor) at Queen Mary University of London, where he co-leads the Social Data Science lab. His research revolves around Social Data Science, interdisciplinary research bridging Computational Social Science and Natural Language Processing. He's particularly interested in linking online data with events in the real world, among others for tackling problematic issues on the Web and social media that can have a damaging effect on individuals or society at large, such as hate speech, misinformation, inequality, biases and other forms of online harm. He serves in the editorial boards of 7 journals, is a regular SPC member of top conferences in computational social science and NLP, and has published over 140 peer reviewed papers, including 50+ journal articles.