Twitter Stance Detection - A Subjectivity and Sentiment Polarity Inspired Two-Phase Approach

Kuntal Dey*†, Ritvik Shrivastava‡ and Saroj Kaushik†
*IBM Research India, New Delhi, India. Email: kuntadey@in.ibm.com
†Indian Institute of Technology, New Delhi, India. Email {anz138579,saroj}@cse.iitd.ac.in
‡Netaji Subhas Institute of Technology, New Delhi, India. Email: ritviks.it@nsit.net.in

***Abstract*—The problem of stance detection from Twitter tweets, has recently gained significant research attention. This paper addresses the problem of detecting the stance of given tweets, with respect to given topics, from user-generated text (tweets). We use the SemEval 2016 stance detection task dataset. The labels comprise of positive, negative and neutral stances, with respect to given topics. We develop a two-phase feature-driven model. First, the tweets are classified as neutral vs. non-neutral. Next, non-neutral tweets are classified as positive vs. negative. The first phase of our work draws inspiration from the subjectivity classification and the second phase from the sentiment classification literature. We propose the use of two novel features, which along with our streamlined approach, plays a key role deriving the strong results that we obtain. We use traditional support vector machine (SVM) based machine learning. Our system (F-score: 74.44 for SemEval 2016 Task A and 61.57 for Task B) significantly outperforms the state of the art (F-score: 68.98 for Task A and 56.28 for Task B). While the performance of the system on Task A shows the effectiveness of our model for targets on which the model was trained upon, the performance of the system on Task B shows the generalization that our model achieves. The stance detection problem in Twitter is applicable for user opinion mining related applications and other social influence and information flow modeling applications, in real life.**

I. Introduction

The user stance detection problem is one where, based upon user-generated content, a system needs to understand the opinion polarity of the user with respect to a topic addressed in the content. The opinion polarity is often expressed in form of discrete class labels, e.g., *positive* or *favor*, *negative* or *against*, and *neutral* or *none*. Thus, stance detection is an integral part of the bigger research problem of opinion mining. Stance detection on Twitter has multiple real-life applications, such as, early detection of stances of individuals towards social, economic, political and other events, towards commercial products, and in information diffusion and influence modeling problems, to name a few among many. This makes the problem important to solve.

While sentiment detection of users has been a long-standing problem [34] [33], the stance detection problem has only recently started to gain the research attention it requires. This was instigated by a seminal work by [29], and the corresponding SemEval 2016 task [28]. The challenge resulted in a spur of first-level research works from the participants. Given the problem novelty, the absence of any literature prior to the SemEval 2016 task, and the absence of prior baselines apart from the one given by the task setters (released post-challenge), the performances delivered by the systems built by the participating teams varied wildly on the test dataset. The overall average values of F-scores ranged from 46.19 at the lower end to all the way up to 67.82 at the higher end. Different models, including deep learning approaches such as convolutional neural networks (CNN) [16] [21], recurrent neural networks (RNN) [7] and long short-term memory (LSTM) [14] [15], traditional machine learning and genetic algorithms, were tried.

However, no system could outperform the overall average F-score (68.98) of the baseline. Further, all the baseline methods proposed by the task setters, including the most successful baseline having an F-score of 68.98, follow the traditional machine learning approach, mostly relying upon Support Vector Machines (SVM). And yet, the baseline outperforms all the advanced modeling efforts including the deep learning ones, as noted by the task setters [28]. The intuition is stark: research around stance detection with respect to given topics, from user-generated content on social media such as Twitter, requires more research attention.

Our work, thereby, aims to improve the first-level understanding, as well as produce an improved baseline, providing the future researchers a robust grounding to improve upon. Interestingly, Igarashi *et al.* [17], a participant in the SemEval 2016 task, observed that, the feature-based model they developed, had outperformed the deep learning (CNN) model that they had also developed. Given limited availability of labeled data, with $2,914$ labeled training data instances spanning over 5 targets in the SemEval 2016 training data, and the first-level observation made by Igarashi *et al.* [17], we employ a traditional machine learning approach, instead of taking a deep neural network based approach. The aim is to also obtain a first-level insight into the impact of the several aspects, modeled by features, that would also convey the intuition behind the process, and act as a robust platform for future research.

We propose a feature-driven two-phase approach, and use SVM based learning. The target dataset, provided by SemEval 2016 [28], comprises of three classes - *favor* (or, "positive"), *against* (or, "negative") and *neither* (or, "none" or "neutral", or "other"). We hypothesize that, messages with neutral stances are likely to have a frame of non-subjectiveness, while those with positive (favor) and negative (against) are likely to con-

2375-9259/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDMW.2017.53

stitute elements with non-neutral sentiments. Based upon this hypothesis, in the first phase, we classify tweets into neutral and non-neutral (favor/against) stance classes, borrowing from the subjectivity detection literature. In the second phase, we use the non-neutral tweets, and classify these tweets into the favor and against classes, making use of the sentiment detection literature. Our approach delivers an average F-score of 74.44 for the Task A, where the target topics are included in the training data. This largely outperforms the state-of-the-art, having an F-score of 68.98, by a huge margin of $74.44 - 68.98 = 5.46$.

The immediate question that arises is - how amenable is our model, where the target topics are not seen in advance? Does it generalize well? This is precisely the problem specified in Task B of the SemEval 2016 challenge. Our system (F-score 61.57) again largely outperforms the literature (F-score 56.28 [37]), improving the F-score by $61.57 - 56.28 = 5.29$. This demonstrates the effectiveness of our approach. Our simplistic but highly effective system, thus, establishes a baseline benchmark for developing more advanced systems in future.

The contributions of our work are the following.

- We propose a simplistic two-phase approach, with intuitive features and traditional SVM learning, to solve the problem of social media user stance detection towards given topics.
- In the first phase, we borrow from the subjectivity literature, and propose a novel syntactic feature, to classify the *neutral* vs. other (non-neutral) tweets.
- In the second phase, we use features from the sentiment polarity detection literature that apply in the current context, and propose a novel semantic feature, to classify the non-neutral tweets into *favor* vs. *against*.
- Empirically, on the SemEval 2016 benchmark dataset, we demonstrate the effectiveness of our system, where the target topic in the test data is part of the training data (SemEval 2016 Task A), as well as where it is not (SemEval Task B). For both tasks, we outperform the literature by large F-score improvements, 5.46 for Task A and 5.29 for Task B.

The rest of our paper is organized as follows. In Section II, we provide an insight into the related work. Section III provides the details of our technical approach. The results and observations are presented in Section IV. We present a discussion in Section V, and also include the possible future directions to take the work forward. Finally, we conclude in Section VI.

II. Related Work

Detecting stance of users towards target topics on the online social media, is a problem fundamental to opinion mining. While other forms of solutions towards the sentiment detection problem exists in multiple settings, only limited research has been carried out till date towards stance detection on social media - namely Twitter. The first-ever benchmark dataset was released by Mohammad *et al.* [28], instigating research in this space. This dataset constitutes of target topics, and tweets to be labeled as *favor* (favorable stance), *against* (unfavorable stance) and *neither* (neutral stance). The task setters had, in parallel, conducted an independent study [29], where the dataset was annotated for both stance and sentiment polarity. A SVM-based approach was used. They used manually annotated sentiment labels as an input to their system, making it different from the SemEval 2016 stance detection task. However, real-life social media data will not have sentiment polarity annotated a-priori. Thus, their approach is not practicable.

As part of the SemEval 2016 task, the task setters had released baselines using the challenge data [28]. Four baselines were released for Task A, where the test set targets are a subset of training set targets, including a majority class based classifier and three SVM-based n-gram models. The task setters' baseline for Task A, with an F-score of 68.98, outperformed the winning work, namely MITRE [40], which produced F-score of 67.82. Further, two baselines were released for Task B, where the test set targets are not a subset of training set targets, including a majority class based classifier and a SVM-based n-gram model.

MITRE [40] provides the best-known deep learning based solution to this problem. The authors use a RNN-based two-layered approach. At the first layer, they pre-train a projection layer, initializing weights from a 256-dimensional word embeddings learned using the word2vec skip-gram algorithm [25]. The second layer is composed of 128 Long Short-Term Memory (LSTM) units [15]. This layer receives as input, a sequence of up to 30 embeddings, folding each into its hidden state in turn. It is initialized with weights, pre-trained using the distant supervision of a hashtag prediction auxiliary task. Among the other works, *pkudblab* [37] and DeepStance [36] use CNN models.

Some works employ two-step solutions. ECNU [41] deploys a two-step learning system. The first step determines whether a given tweet is relevant to the given target topic. The second step addresses orientation detection, where the stance polarity (favor/against) is detected. The work by ltl.uni-due [39] also uses a two-level stacked classifier approach. Their first layer classifier identifies the neutral stances, and the second classifier distinguishes between the favor/against stances. While our work improves upon this (as well as outperforms all the other works in the literature) via providing two novel features that deliver significant impact on the system performance, the philosophy underlying our work is aligned to this work. In a recent work, Du *et al.* [9] propose a target-specific neural attention model, and augment the embeddings of the constituent words of the tweets with the embeddings of the stance target.

Several other works, mostly using traditional machine learning, exist in the literature, all from the SemEval 2016 task. TakeLab [6], fine-tunes off-the-shelf machine learning algorithms with genetic algorithms. Other works, such as CU-GWU [10], IUCL-RF [23], IDI@NTNU [5], JU_NLP [30] and NLDS-UCSC [27], all use variants of traditional machine learning classifiers. CU-GWU [10] uses a lexical,

sentiment, semantic dictionaries and latent and frame semantic features, and performs SVM based learning. IUCL-RF [23] uses a random forest model. It uses gradient boosting decision trees and SVM, and merges all classifiers into an ensemble method. IDI@NTNU [5] builds a supervised system that combines shallow features and pre-trained word vectors as word representation. JU_NLP [30] uses SVM, learning from features built upon target-specific words, sentiment words and dependency information. NLDS-UCSC [27] learns using a maximum entropy classifier, using surface-level, sentiment and domain-specific features. USFD [2] uses feature autoencoders, and solves only for Task B. INF-UFRGS [8] also solves only Task B, combining sentiment detection solutions, n-grams representing opinion targets and common terms to denote stance.

Tohoku [17] conducts a comparative study between traditional and deep learning models, towards the stance detection task, for the given benchmark dataset. They develop a feature-based model with features such as PoS tags, sentiment, mutual information and bag features, and apply logistic regression, on one hand. On the other hand, they also perform a CNN-based learning. They make the remarkable observation that, the traditional feature-based method outperforms CNN for the given test data. They observe that, although CNN outperforms the traditional feature-based machine learning for cross-validation, but the traditional feature-based method outperformed CNN for test data, which also motivates us to strengthen the baseline using traditional feature-based machine learning.

In summary, many works have been carried out for stance detection using the SemEval 2016 benchmark dataset; however, all of the participating teams were comprehensively outperformed by the task setters baseline work for Task A, and the performance attained in Task B also has left much to be desired. While some works such as ECNU [41] and ltl.uni-due [39] have used two-step solutions using several features, their solutions approaches are different from ours. We build our solution using the subjectivity features known to be effective in the first phase, and the sentiment features known to be effective in the second phase.

We make use of features drawing from the literature of sentiment and subjectivity analysis. An abundant volume of related works exist in the literature. The works by [1], [4] and [20], and Opinion Finder at University of Pittsburgh [38], are some examples. A number of recent works, such as Khan *et al.* [18], Kolchyna *et al.* [19], Le and Nguyen [22], Severyn and Moschitti [35] and Zimbra *et al.* [42], have also attempted to look into the sentiment analysis problem. Some of these works also perform subjectivity detection, as this helps in filtering objective (non-subjective) entries away from the classifier, which in turn leads to an improved performance of the overall system. Works, such as [24] and [31], address related problems such as personality detection and sarcastic opinion detection in tweets. A dedicated SemEval task is set often enough, where several research works investigating sentiment polarity are published, such as, Rosenthal *et al.* [34] [33].

Our work borrows from this rich literature, making our solution more effective compared to the current state of the art, outperforming the state of the art significantly and consistently.

III. Our Approach

To provide an intuitive "feel" of the data at hand to the reader, we provide a few examples randomly chosen from the training set, from a few of the given topics, in Table I. The stance data comprises of three classes: *favor* ("positive"), *against* ("negative") and *neither* ("neutral"). We note that, the *favor* and *against* tweets are often subjective in nature, while the *neutral* tweets often are non-subjective. Inspired by this observation, we perform a two-phase SVM based machine learning, following a preprocessing phase. This approach allows us to construct features, drawing from rich literature, namely the subjectivity detection literature in the first phase, and sentiment polarity classification literature in the second phase. The architecture of our system is presented on Figure III.

A. Preprocessing

We perform traditional Twitter data preprocessing, in order to improve our system performance. Our preprocessing comprises of the tweet normalization, stopword removal and stemming. The details of the preprocessing is given below.

1) Tweet normalization: We normalize the tweets normalized using net slang and Han-Baldwin normalization dictionary knowledge [12]. This helps in resolving on-the-net expressions that are colloquial in nature and do not exist in standard dictionaries. For instance, the term *aaf* is resolved as *as a friend*. For net slang normalization, we use an online dictionary[1].

2) Stopword removal: Stopword removal is an essential step of our preprocessing. This step ensures that the superfluous words with practically no information content for the task under consideration are discarded (such as prepositions, article *etc.*). We make use of an online resource to perform stopword removal, which is a part of the Stanford NLP resources[2].

3) Stemming: We perform stemming on the tokens identified in the user-generated Twitter content strings. We use both the main tokens as well as the stemmed tokens, for the n-gram and other features that we use in our work. The stemming is carried out using the well-accepted Porter stemmer [32].

B. Feature Construction and Training

The challenges to perform the task of stance detection, exist at different NLP layers. This includes (a) lexical, (b) syntactic, (c) semantic and (d) pragmatic challenges. For a robust and intuitive solution, we break the task up into two phases. In the first phase, the objective is to identify the tweets with neutral stances with respect to the target topics, and segregate the non-neutral (favor/against) stances from the neutral ones. In the subsequent (second) phase, the aim is

[1]http://www.noslang.com/dictionary

[2]https://nlp.stanford.edu/IR-book/html/htmledition/ dropping-common-terms-stop-words-1.html

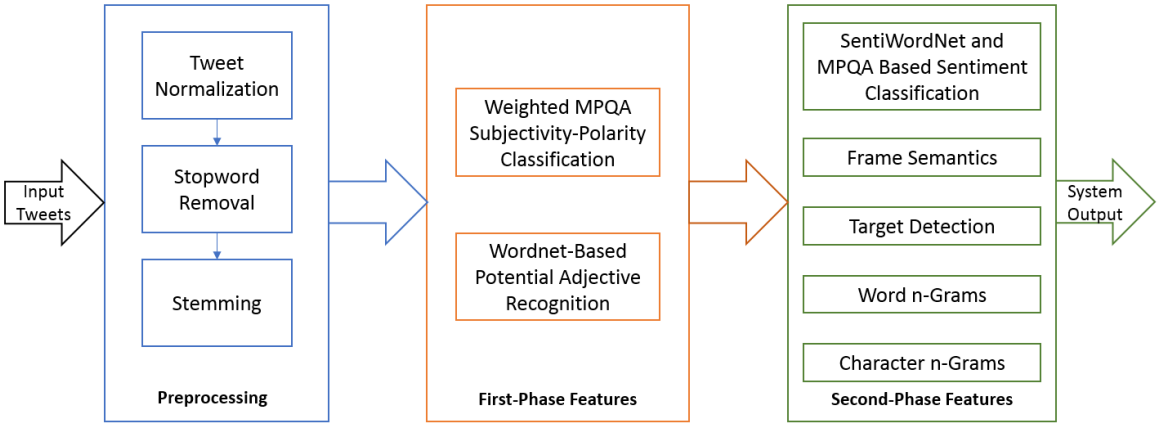

Fig. 1. System Architecture Diagram

ID	Target	Tweet	Stance
Examples from the **favor** *stance*			
111	Atheism	Everyone is able to believe in whatever they want. #Freedom #SemST	FAVOR
1220	Feminist Movement	@OliviaJeniferx it's not always the guys job. #equality #SemST	FAVOR
1921	Hillary Clinton	@vacanti @samglover Maybe a woman should be President. #SemST	FAVOR
Examples from the **against** *stance*			
586	Atheism	Be still. Be patient. Watch and let God work. #SemST	AGAINST
1359	Feminist Movement	Friendly reminder that the "Gender Pay Gap" is a myth. #SemST	AGAINST
2124	Hillary Clinton	Insurgent. What will happen if Hillary becomes dictator in chief. #SemST	AGAINST
Examples from the **none** *stance*			
180	Atheism	Alot of angry people in this world. Peace to all. #love #SemST	NONE
1381	Feminist Movement	@sass_unicorn lol! Young male children for #SemST	NONE
1922	Hillary Clinton	Today LOVE won and Hate was defeated. #MorningJoe #UnitedBlue #SemST	NONE

TABLE I
RANDOM EXAMPLES OF TWEETS OF THE DIFFERENT STANCES, FOR A FEW OF THE GIVEN TARGET TOPICS

to perform a classification between the positive (favor) and negative (against) stances, from the non-neutral tweets. Our system closely follows the subjectivity and sentiment literature respectively, for the feature construction process over the two phases, and we empirically observe our system to be more effective compared to the rest of the literature.

We perform SVM-based learning from these features. We first train a model using the first phase features, and the entire dataset. We subsequently train the second phase model, using the second phase features separately, using only the data with non-neutral tweets present within the dataset. For testing, we initially run the first model, and then on the non-neutral output we run the second model, to obtain our final output.

1) First Phase Features: The first phase is used to segregate the neutral-stance tweets from the non-neutral ones. Here, we draw from the subjectivity detection literature, and use other related features to create our feature set. The features we construct for the first phase of our system, are as follows.

- **Weighted MPQA Subjectivity-Polarity Classification:** The tokens and the stemmed tokens, from the tweets, are measured against the MPQA subjectivity lexicon, with weights and polarities being taken into account. Each tweet has a cumulative sum of the scores, obtained as a sum of the individual tokens it comprises. The positive and negative polarity tweets are assigned positive and negative weights respectively. Tokens matching the strong subjective set are assigned a magnitude of 2, whereas the weak subjective words are assigned a magnitude of 1. Each tweet has a cumulative weight as the sum of the scores of its constituent tokens. We construct a pragmatic boolean feature where, Tweets with an overall positive sum beyond +2 or a negative sum below -2 are termed as subjectively inclined, whereas the intermediate values are marked otherwise.
- **Wordnet Based Potential Adjective Recognition:** Hatzivassiloglou and Wiebe [13] show that different kinds of adjectives, such as dynamic adjectives, semantically oriented adjectives, and gradable adjectives are strong predictors of presence of subjectivity. Motivated by this, we aim to analyze the presence of such adjectives in a syntactically extended set. We extensively use Wordnet [26], to enable the construction of this syntactic feature, wherein, we aim to detect whether a token given in the tweet content, exists in Wordnet and is marked as an adjective there, and construct a boolean feature accordingly. Note that, words that at all can be used as an adjective (as per Wordnet), are used to form this feature, and not the actual usage of the word in text (which would be tagged by the PoS tagger). This is a novel feature in context of the problem at hand, and is syntactic in nature.

2) Second Phase Features: The second phase is used to differentiate between the tweets where the stance is positive (favor) and those where the stance is negative (against), from the set of subjective tweets. In this phase, we make use of the sentiment detection literature. The input to the training are all the tweets labeled with a non-neutral stance. In the testing phase, the tweets labeled as non-neutral in the first phase, will be passed to this phase. The features that we use in the second phase are as follows.

- **SentiWordNet and MPQA Based Sentiment Classification:** This pragmatic feature provides an overall positive or negative score to the sentence, to interpret the sentiment behind it. Using positive and negative word lists from SentiWordNet, tokens are assigned a polarity score of +1 or -1, for belonging to these lists respectively. The sum of every token is taken as the feature for classification. Further, MPQA subjectivity is also used, as an integer sum of the polarity score of all the tokens present in the tweet content.
- **Frame Semantics:** "Frame semantics assemble the meanings of different elements in a given piece of text to model the meaning of the whole text" [3]. In our setting, we use a primitive but effective approach for identifying the different elements of the text, namely, *connector words*. If a tweet (often akin to compound sentences in nature) comprises of connector tokens, such as *but*, *although*, *also*, *therefore*, then the two clauses it tends to combine have a varying impact on the semantics of the sentence as a whole. Connectors with the 'opposite' feel, such as *but* and *although*, give more importance to the latter clause, while those with the 'apposite' feel, give the first clause support by using an appropriate complimenting second clause. For small multi-sentence tweets, this applies across the sentence boundaries as well. We assign more weightage to the more important clause, in case connector words are present in the sentence. This is a novel feature in context of the problem at hand, and is semantic in nature.
- **Target Detection:** This is a boolean feature. The value of this feature is set to be true if the given target (as a whole) is present in the content of the tweet, and is set to false otherwise.
- **Word n-grams:** We construct a set of word n-gram features (lexical). These boolean features are used to mark the presence of word n-grams, holding true if any n-gram present in the target string also belongs to the tweet content, and false otherwise. For instance, if both the target string as well as the tweet text contain the bigram "Hilary Clinton", then the word bigram feature will be set to *true*. Specifically, we use unigrams, bigrams and trigrams.
- **Character n-grams:** Similar to word n-grams, we also construct a set of character n-gram (lexical) boolean features. These features are used to check the presence of character bigrams, trigrams and 4-grams, containing any character-token of the target string in the tweet. Note that, special characters and whitespaces are excluded while the character n-gram features are constructed.

The above completes the list of our simplistic but intuitive features. We now perform our two-phase SVM based approach, for our experiments.

IV. Experiments

In this section, we present the experiments we conducted. We illustrate the observations we make on the performance of our system, and compare it with other existing systems.

A. Data Description

As mentioned earlier in the paper, we use the benchmark training and test data provided by the SemEvam 2016 stance detection task setters [28]. For the purpose of self-containment, we reproduce the data statistic shared in their paper, in Table II. Note that, in order to ensure appropriate comparisons, we use the evaluation scripts shared by the task setters, instead of creating separate evaluation scripts. Further note that, we use the Weka [11] tool to perform our machine learning activities.

B. Results

We perform preprocessing, followed by a two-phase SVM based machine learning using Weka [11] with a linear kernel and default parameters (cost function $C = 1$ and $L2$ regularizer), with 10% held-out data for the purpose of model development.

Table III shows the performances delivered by each of the systems for Task A, as reported by the task setters. Our system outperforms all the other systems for all the F-scores that have been reported in the literature - the positive (favor), the negative (against) and the overall F-scores. It outperforms the state-of-the-art *favor* F-score by $69.53 - 62.98 = 6.55$, the *against* F-score by $79.36 - 78.44 = 0.92$ and the overall F-score by $74.44 - 68.98 = 5.46$. Further, when one notes on Table III that, the F-scores obtained by our system exceeds all best-performance points delivered by all the models (with a lone exception), the significance of our results becomes even more evident.

Our system also delivers commendable performance for detecting the user stances towards the individual topics. The IDI@NTNU system outperforms ours for the *climate* topic, but it does not perform anywhere nearly as well in any of the other topics, settling at a rank of #10 amongst the participants of the SemEval 2016 challenge, not counting the baselines, and not counting our work. Thus, for the topic *climate*, our performance rank is #2, second to the IDI@NTNU system. Our system is outperformed by an F-score of $54.86 - 53.59 = 1.27$. For all the remaining topics, we massively outperform all the other systems. For the topics *atheism*, *feminist movement*, *Hillary Clinton* and *legalization of abortion*, our system outperforms the respective best performing systems by massive F-scores of $72.5 - 67.25 = 5.25$, $78.77 - 62.09 = 16.68$, $79.7 - 67.12 = 12.58$ and $83.6 - 66.42 = 17.18$ respectively.

Target	#total	#train	% of instances in Train favor	against	neither	#test	% of instances in Test favor	against	neither
Data for Task A									
Atheism	733	513	17.9	59.3	22.8	220	14.5	72.7	12.7
C.C.C.	564	395	53.7	3.8	42.5	169	72.8	6.5	20.7
Feminist Movement	949	664	31.6	49.4	19.0	285	20.4	64.2	15.4
Hillary Clinton	984	689	17.1	57.0	25.8	295	15.3	58.3	26.4
L.A.	933	653	18.5	54.4	27.1	280	16.4	67.5	16.1
All	4,163	2,914	25.8	47.9	26.3	1,249	24.3	57.3	18.4
Data for Task B									
Donald Trump	707	0	-	-	-	707	20.93	42.29	36.78

TABLE II
DATA FOR THE SEMEVAL 2016 STANCE DETECTION TASK. TARGET C.C.C. → CLIMATE CHANGE IS CONCERN. TARGET L.A. → LEGALIZATION OF ABORTION. TABLE COURTESY: [28].

	Overall			Atheism	Climate	Feminism	Hillary	Abortion
	F_{favour}	$F_{against}$	F_{avg}	F_{avg}	F_{avg}	F_{avg}	F_{avg}	F_{avg}
Our System	**69.53**	**79.36**	**74.44**	**72.5**	53.59	**78.77**	**79.7**	**83.6**
Baselines given by the SemEval 2016 Task Setters								
Majority class	52.01	**78.44**	65.22	42.11	42.12	39.10	36.83	40.30
SVM-unigrams	54.49	72.13	63.31	53.25	38.39	55.65	57.02	60.09
SVM-ngrams	**62.98**	74.98	**68.98**	65.19	42.35	57.46	58.63	**66.42**
SVM-ngrams-comb	54.11	70.01	62.06	53.27	47.76	52.82	56.50	63.71
Participants' Performances in SemEval 2016								
MITRE	59.32	76.33	67.82	61.47	41.63	**62.09**	57.67	57.28
pkudblab	61.98	72.67	67.33	63.34	52.69	51.33	64.41	61.09
TakeLab	60.93	72.73	66.83	**67.25**	41.25	53.01	**67.12**	61.38
PKULCWM	56.96	74.55	65.76	56.39	40.39	51.32	62.26	61.56
ECNU	60.55	70.54	65.55	61.97	41.32	56.21	57.85	61.25
CU-GWU	54.99	72.21	63.60	55.68	39.41	53.88	51.19	59.38
IUCL-RF	52.61	74.59	63.60	57.93	39.06	51.06	49.84	57.61
DeepStance	58.44	68.65	63.54	52.90	40.40	52.34	55.35	63.32
UWB	57.41	69.42	63.42	57.88	46.90	51.82	59.82	61.98
IDI@NTNU	58.97	65.97	62.47	59.59	**54.86**	48.59	57.89	54.47
Tohoku	49.25	75.18	62.21	58.90	39.51	52.41	39.81	37.75
ltl.uni-due	48.71	74.75	61.73	52.47	35.50	55.12	44.23	57.25
LitisMind	50.67	72.20	61.44	52.36	39.15	57.16	42.08	45.88
JU_NLP	46.68	74.53	60.60	38.99	42.60	45.65	50.25	41.83
NEUSA	49.03	71.20	60.12	48.90	41.95	52.14	48.53	61.89
nldsucsc	50.90	67.81	59.36	57.19	42.10	48.97	57.27	61.66
WFU/TNT	47.55	70.89	59.22	46.16	42.07	47.91	45.88	45.34
INESC-ID	50.58	64.57	57.58	52.67	44.92	49.00	50.64	49.93
Thomson Reuters	30.16	62.23	46.19	44.79	35.86	39.37	34.98	38.89

TABLE III
COMPARING PERFORMANCE OF OUR SYSTEM FOR TASK A OF SEMEVAL 2016, WITH THE LITERATURE

The SemEval 2016 stance detection task setters also evaluate the submitted systems to determine, whether the systems are capable of detecting stances when opinion is expressed towards some other entity. To this, they split the test set into two: (a) one subset constitutes data where the opinion is expressed towards the target topic, and (b) another subset where the opinion expressed is towards an entity different from the target entity (target topic). The results are presented in Table IV. The task setters explicitly note that, *"the stance task is markedly more difficult when stance is to be inferred from a tweet expressing opinion about some other entity (and not the target of interest)"*. Our system outperforms the best-performing system where the target entity is known, by an F-score of $79.89 - 74.54 = 5.35$. And in the more difficult task where the opinion is expressed about some other entity, our system outperforms the best-performing system, by $52.45 - 49.34 = 3.11$. Thus, across the topics, our system delivers major improvement over the state of the art. This further highlights the generalization of our simplistic

C. Model Generalization

The amenability of our system towards generalization, becomes even more evident, when one examines the results of Task B, presented in Table V. Task B addresses the scenario of model generalization, where a model is tested with target topics, that are not part of the training target topics. Note that, we use the model trained with the given training data (the same data that is used to train for Task A), to perform testing for Task B. Here, our system outperforms the best-performing *favor* detector by an F-score of $59.72 - 57.39 = 2.33$, the best-performing *against* detector by $63.41 - 59.44 = 3.97$ and the overall best performing system in the state of the art by $61.57 - 56.28 = 5.29$. Thus, our system generalizes better than the state of the art, indicating a potentially larger practical usability compared to the literature. This further indicates that our model does not overfit to the training data provided.

D. Impact of the Features

Table VI presents the results of ablation test over Task A, demonstrating the impact of the features. Note that, the

Team	Opinion Towards: Target	Opinion Towards: Other	All
Our System	**79.89**	**52.45**	**74.44**
Baselines			
Majority class	71.27	41.33	65.22
SVM-unigrams	69.39	38.96	63.31
SVM-ngrams	**74.54**	43.20	**68.98**
SVM-ngrams-comb	66.60	38.05	62.06
Participating Teams			
MITRE	72.49	44.48	67.82
pkudblab	71.07	46.66	67.33
TakeLab	73.66	37.47	66.83
PKULCWM	70.62	45.89	65.76
ECNU	70.29	44.25	65.55
CU-GWU	67.89	45.28	63.60
IUCL-RF	67.77	41.96	63.60
DeepStance	67.81	44.00	63.54
UWB	67.60	44.54	63.42
IDI@NTNU	66.25	42.26	62.47
Tohoku	66.44	44.09	62.21
ltl.uni-due	67.23	42.45	61.73
LitisMind	66.42	41.27	61.44
JU_NLP	62.55	**49.34**	60.60
NEUSA	65.39	39.48	60.12
nldsucsc	65.71	34.64	59.36
WFU/TNT	67.28	34.89	59.22
INESC-ID	63.99	36.63	57.58
Thomson Reuters	49.98	32.43	46.19

TABLE IV
RESULTS FOR SEMEVAL 2016 TASK A (THE OFFICIAL COMPETITION METRIC F_{avg}), ON DIFFERENT SUBSETS OF THE TEST DATA, FOR SOME KEY PARTICIPANTS, AND OUR SYSTEM PERFORMANCE

Team	F_{favor}	$F_{against}$	F_{avg}
Our System	**59.72**	**63.41**	**61.57**
Baselines given by SemEval 2016 Task Setters			
Majority class	0.00	**59.44**	29.72
SVM-ngrams-comb	18.42	38.45	28.43
Participating Teams			
pkudblab	**57.39**	55.17	**56.28**
LitisMind	30.04	59.28	44.66
INF-UFRGS	32.56	52.09	42.32
UWB	34.26	49.78	42.02
ECNU	17.96	50.20	34.08
USFD	10.93	54.46	32.70
Thomson Reuters	14.39	50.39	32.39
ltl.uni-due	46.56	05.71	26.14
NEUSA	16.59	34.87	25.73

TABLE V
COMPARING PERFORMANCE OF OUR SYSTEM FOR TASK B OF SEMEVAL 2016, WITH THE LITERATURE

two features that are novel in the context of the task at hand, namely Wordnet-based potential adjective recognition and the frame semantics feature, deliver strong impacts, as is clear from the improvement these deliver in the average F-score values. These features play a decisive role in lifting the performance of our system over and beyond the present state-of-the-art literature.

Features	Avg. F-score
Senti WordNet + Weighted MPQA + Wordnet Adjective Recognition **(all first phase features)**	61.83
+ Frame semantics	63.89
+ Target matching	67.72
+ Word n-grams	71.68
+ Character n-grams **(all features)**	74.44

TABLE VI
ABLATION TEST FOR OUR FEATURE SET

V. DISCUSSION

Choice of Approach - Traditional Learning over Deep Learning: Our approach is a traditional feature-engineering based one, with SVM-based machine learning. The reason of making this choice is two-fold. One, the traditional machine learning baseline by [28] has outperformed the best deep learning system proposed by [40] so far. Two, the available volume of annotated stance data is limited. The SemEval 2016 task consists of merely $2,914$ training data rows. [17] also experienced better results with traditional machine learning over deep learning, with the given dataset. As and when a sufficiently large dataset for stance detection becomes available for training, a deep-learning model will be likely to deliver stronger performances, and will require a revisit.

Feature Novelty: We used two features that are novel in the context of the task at hand: the Wordnet-based potential adjective recognition feature in the first phase and the frame semantics feature in the second phase. As indicated in Table VI, these features play key roles in obtaining the strong results that our work gets.

Distribution of Errors: A glance at Table III clearly shows that, the proportions errors of the positive and negative classes in our work is much more well-balanced in our system, and more so, compared to all the other high-performing systems. Further, expectedly, we observe errors to be more prevalent, when the target keywords are not part of the user tweet. This is clear from both Table IV as well as Table V. Clearly, other systems also suffer from similar issues; further, our system suffers the least amongst all, helping it deliver the highest performances among all the systems, and in all the aspects.

Future Work: The present work can be enhanced in the future in the following possible directions.

- It will be interesting to augment the model, incorporating pragmatic information about the potential targets (where the targets would not be known in advance), for obtaining better system performance.
- Further, it will be interesting to perform a study to understand the human upper bound of stance detection that can be attained, in order to truly appreciate the performance delivered by the computational systems.

VI. CONCLUSION

In this paper, we proposed a two-phase model for detecting user stance with respect to given topics on Twitter. The first phase of our model, where we classified the tweets into two - positive or negative versus neutral-stance - was inspired by subjectivity classification features. The second phase of our model, where we classified the positive versus negative stances, was inspired by sentiment classification features. We proposed two features that have never been explored before in the context of stance detection on Twitter, namely *frame semantics* and *Wordnet based potential adjective recognition*, both of which played key roles in improving the results obtained by the system. We used traditional SVM based learning. We empirically demonstrate the effectiveness of our

model, by testing it against the benchmark SemEval 2016 stance detection dataset. For the task where the testing was done on topics that were seen by the training data (SemEval 2016 Task A), our system delivered an F-score performance of 74.44, outperforming the state of the art at 68.98. For the test data where the training had not seen the test topics (SemEval 2016 Task B), our system delivered an F-score of 61.57, outperforming the literature (F-score 56.28) Our model is easy to implement, reusable, lightweight and practicable. The Twitter stance detection problem is of interest not only to NLP researchers aiming to cross the barriers of the technical challenges associated with the problem, but also in real-life application dynamics modeling, social influence flow modeling, and information diffusion dynamics modeling, among others.

REFERENCES

[1] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau. Sentiment analysis of twitter data. In *Proceedings of the Workshop on Languages in Social Media*, pages 30–38. Association for Computational Linguistics, 2011.

[2] I. Augenstein, A. Vlachos, and K. Bontcheva. Usfd at semeval-2016 task 6: Any-target stance detection on twitter with autoencoders. In *Proceedings of the International Workshop on Semantic Evaluation, SemEval*, volume 16, 2016.

[3] C. F. Baker, C. J. Fillmore, and J. B. Lowe. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1*, pages 86–90. Association for Computational Linguistics, 1998.

[4] L. Barbosa and J. Feng. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 36–44. Association for Computational Linguistics, 2010.

[5] H. Bøhler, P. F. Asla, E. Marsi, and R. Sætre. Idi@ ntnu at semeval-2016 task 6: Detecting stance in tweets using shallow features and glove vectors for word representation. *Proceedings of SemEval*, 2016.

[6] F. Boltuzic, M. Karan, D. Alagic, and J. Šnajder. Takelab at semeval-2016 task 6: Stance classification in tweets using a genetic algorithm based ensemble. *Proceedings of SemEval*, pages 464–468, 2016.

[7] H. B. Demuth, M. H. Beale, O. De Jess, and M. T. Hagan. *Neural network design*. Martin Hagan, 2014.

[8] M. Dias and K. Becker. Inf-ufrgs-opinion-mining at semeval-2016 task 6: Automatic generation of a training corpus for unsupervised identification of stance in tweets. *Proceedings of SemEval*, pages 378–383, 2016.

[9] J. Du, R. Xu, Y. He, and L. Gui. Stance classification with target-specific neural attention networks.

[10] H. Elfardy and M. Diab. Cu-gwu perspective at semeval-2016 task 6: Ideological stance detection in informal text. *Proceedings of SemEval*, pages 434–439, 2016.

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[12] B. Han and T. Baldwin. Lexical normalisation of short text messages: Makn sens a# twitter. In *ACL-HLT, Volume 1*, pages 368–378, 2011.

[13] V. Hatzivassiloglou and J. M. Wiebe. Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th conference on Computational linguistics-Volume 1*, pages 299–305. Association for Computational Linguistics, 2000.

[14] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[15] S. Hochreiter and J. Schmidhuber. Lstm can solve hard long time lag problems. *Advances in neural information processing systems*, pages 473–479, 1997.

[16] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106–154, 1962.

[17] Y. Igarashi, H. Komatsu, S. Kobayashi, N. Okazaki, and K. Inui. Tohoku at semeval-2016 task 6: Feature-based model versus convolutional neural network for stance detection. *Proceedings of SemEval*, pages 401–407, 2016.

[18] A. Z. Khan, M. Atique, and V. Thakare. Combining lexicon-based and learning-based methods for twitter sentiment analysis. *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)*, page 89, 2015.

[19] O. Kolchyna, T. T. Souza, P. Treleaven, and T. Aste. Twitter sentiment analysis. *arXiv preprint arXiv:1507.00955*, 2015.

[20] E. Kouloumpis, T. Wilson, and J. Moore. Twitter sentiment analysis: The good the bad and the omg! *ICWSM*, 11:538–541, 2011.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[22] B. Le and H. Nguyen. Twitter sentiment analysis using machine learning techniques. In *Advanced Computational Methods for Knowledge Engineering*, pages 279–289. Springer, 2015.

[23] C. Liu, W. Li, B. Demarest, Y. Chen, S. Couture, D. Dakota, N. Haduong, N. Kaufman, A. Lamont, M. Pancholi, et al. Iucl at semeval-2016 task 6: An ensemble model for stance detection in twitter. *Proceedings of SemEval*, pages 394–400, 2016.

[24] N. Majumder, S. Poria, A. Gelbukh, and E. Cambria. Deep learning-based document modeling for personality detection from text. *IEEE Intelligent Systems*, 32(2):74–79, 2017.

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

[26] G. A. Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[27] A. Misra, B. Ecker, T. Handleman, N. Hahn, and M. Walker. Nlds-ucsc at semeval-2016 task 6: A semi-supervised approach to detecting stance in tweets. *Proceedings of SemEval*, pages 420–427, 2016.

[28] S. M. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of SemEval*, volume 16, 2016.

[29] S. M. Mohammad, P. Sobhani, and S. Kiritchenko. Stance and sentiment in tweets. *arXiv preprint arXiv:1605.01655*, 2016.

[30] B. G. Patra, D. Das, and S. Bandyopadhyay. Ju nlp at semeval-2016 task 6: Detecting stance in tweets using support vector machines. *Proceedings of SemEval*, pages 440–444, 2016.

[31] S. Poria, E. Cambria, D. Hazarika, and P. Vij. A deeper look into sarcastic tweets using deep convolutional neural networks. *arXiv preprint arXiv:1610.08815*, 2016.

[32] M. F. Porter. Snowball: A language for stemming algorithms, 2001.

[33] S. Rosenthal, P. Nakov, S. Kiritchenko, S. M. Mohammad, A. Ritter, and V. Stoyanov. Semeval-2015 task 10: Sentiment analysis in twitter. *Proceedings of SemEval-2015*, 2015.

[34] S. Rosenthal, A. Ritter, P. Nakov, and V. Stoyanov. Semeval-2014 task 9: Sentiment analysis in twitter. In *SemEval 2014*, pages 73–80, 2014.

[35] A. Severyn and A. Moschitti. Twitter sentiment analysis with deep convolutional neural networks. In *SIGIR*, pages 959–962. ACM, 2015.

[36] P. Vijayaraghavan, I. Sysoev, S. Vosoughi, and D. Roy. Deepstance at semeval-2016 task 6: Detecting stance in tweets using character and word-level cnns. *arXiv preprint arXiv:1606.05694*, 2016.

[37] W. Wei, X. Zhang, X. Liu, W. Chen, and T. Wang. pkudblab at semeval-2016 task 6: A specific convolutional neural network system for effective stance detection. *Proceedings of SemEval*, pages 384–388, 2016.

[38] T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, C. Cardie, E. Riloff, and S. Patwardhan. Opinionfinder: A system for subjectivity analysis. In *Proceedings of hlt/emnlp on interactive demonstrations*, pages 34–35. Association for Computational Linguistics, 2005.

[39] M. Wojatzki and T. Zesch. ltl. uni-due at semeval-2016 task 6: Stance detection in social media using stacked classifiers. *Proceedings of SemEval*, pages 428–433, 2016.

[40] G. Zarrella and A. Marsh. Mitre at semeval-2016 task 6: Transfer learning for stance detection. *arXiv preprint arXiv:1606.03784*, 2016.

[41] Z. Zhang and M. Lan. Ecnu at semeval-2016 task 6: Relevant or not? supportive or not? a two-step learning system for automatic detecting stance in tweets. *Proceedings of SemEval*, pages 451–457, 2016.

[42] D. Zimbra, M. Ghiassi, and S. Lee. Brand-related twitter sentiment analysis using feature engineering and the dynamic architecture for artificial neural networks. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pages 1930–1938. IEEE, 2016.