Weakly Supervised Stance Learning Using Social-Media Hashtags

Sumeet Kumar
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213, USA

Abstract

Extracting stance from a text is fundamental to automate opinion mining on the web. Recently, many models have been proposed to advance the state-of-the-art. However, most of these models are designed for small hand-labeled datasets that reference a single topic or only a few topics. These models do not generalize to new topics and unseen data. On social media, we have observed that certain hashtags (e.g. 'climatehoax') carry stance information. In this research, we propose an approach that uses such hashtags for weak supervision to learn stance in the text. First, we use text sentiment to identify potential *hashtags* that carry stance information. We call theses hashtags *stance-tags*. Then, we collect a large amount of Twitter data using these *stance-tags*. Using the stance of *stance-tags*, we train two kinds of deep-learning models to predict stance in text that doesn't necessarily contain hashtags. Further, given a few labeled examples, we model the task of finding stance-tags that are most suitable for stance learning as a multi-armed-bandit problem and find solutions using four optimization strategies. We validate our approach with experiments on a hand-labeled dataset that contains stance on five topics, obtaining results comparable to the state-of-the-art. As many hahstags are used together, they are not completely independent. In the future work section, we propose a search strategy that exploits that similarity between hashtags.

Introduction

Peoples attitude have consistently strong relationship with their behaviour when directed at the same target (Ajzen and Fishbein1977). Because people express their opinions on blogs and other social media platforms, it is expected that consistent attitudes can be observed in posts made on social media. Automated ways to understand attitudes in such user-generated corpus is of immense value. It is especially essential to understand the stance – which involves finding people's attitude about a topic of interest – of users in polarized communities, which are increasingly of interest to researchers. Therefore, it's not surprising that many researchers have explored automated ways to learn stance given a text (Hasan and Ng2013; Mohammad et al.2017; Joseph et al.2017). However, most recent research on stance learning is constrained to a few small hand-curated datasets.

Building manually labeled datasets is of great value as it allows for the easy comparison of various learning algorithms with respect to humans expectations. However, in the real-world models built on these small datasets do not generalize, especially to unseen data. Also, these hand labeled datasets are only available for a few topics, and models trained on these datasets are not appropriate for use with new topics. In this research, we propose a more flexible approach of learning stance that primarily uses data collected using specific hashtags. Our method allows for training stance-learning models on a new topic with minimal human supervision.

Neural-network models have revolutionized many areas of artificial intelligence(Krizhevsky et al.2012). In particular, the best performing models on most text-processing tasks are deep-learning neural networks (e.g. Felbo et al.2017). However, the training of deep-learning models requires a large amount of data. In fact, much of the recent improvements in computer-vision (Krizhevsky et al.2012) and text-understanding (Mikolov et al.2013; Kiros et al.2015) can be attributed to the presence of a large amount of data for these tasks. While it's is possible to build large labeled-dataset for other tasks, the stance of a text can be very context dependent (different topics) and the required dataset for stance-learning may not be economically feasible to produce.

Though past success in deep learning was made possible because of large labeled datasets, more recently, training models using weak labels has been shown achieve comparable performance (Felbo et al.2017; Mahajan et al.2018). For example, recently researchers at Facebook achieved the state-of-the-art accuracy on object detection by using images annotated with hashtags (Mahajan et al.2018). So far this approach has been applied to a limited set of tasks. Stance mining, where hand labled datasets are limited but there is a large amount of weakly labeled data in the form of social-media posts, may be another type of task that could be improved upon by using weak supervision. In weak supervision, our objective remains the same, however, instead of ground-truth data for training, data with noisy labels are used.

The use of hashtags on social-media is ubiquitous and the text associated with some of these hashtags could provide a useful signal for weak supervision. In this research, we propose to use hashtags for stance learning. Prior work

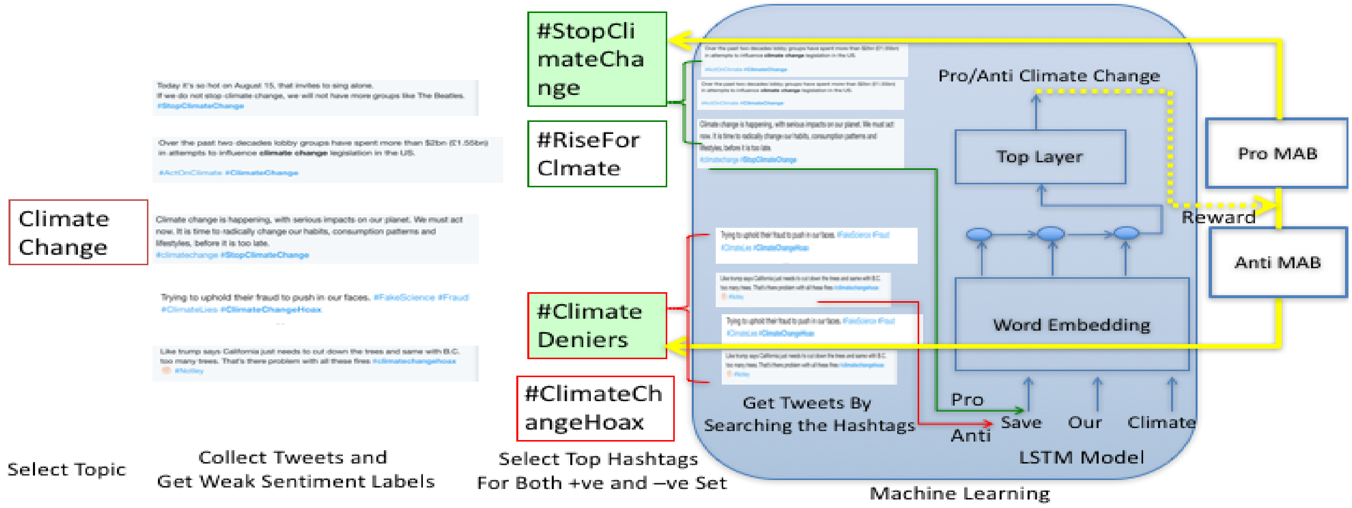

Figure 1: Stance learning workflow: Given a topic, using sentiment analysis we first extract the top hashtags having 'pro' and 'anti' stance on that topic. Using these *stance hashtags*, we download more data that are available on Twitter. This large dataset is then used to train neural-network models for stance classification. Moreover, given a few hand-labeled examples, a multi-armed-bandit (MAB) optimization is used to pick the best pro and anti hashtag for classification.

has shown that hashtags added at the end of the text in social-media posts often carry stance (Evans2016). Researchers have explored using hashtags for emotion (Davidov et al.2010; Qadir and Riloff2013) and topic (Wang et al.2014) labeling tasks. Searching for such hashtags, on Twitter, we observed that certain hashtags carry stance information and are popular (e.g. 'climatehoax'). In this research, we propose an approach that uses data obtained by searching hashtags to train deep-learning models that can predict stance in a text (see Fig. 1). We use text sentiment to identify potential hashtags that can carry stance information. Given a topic, we pick a few *hashtags* that have either higher mean positive sentiment or mean negative sentiment. We call these hashtags '*stance-tags*' as these hashtags can carry stance information. Then, we collect a large number of tweets using these *stance-tags*. Using weak supervision from the stance of *stance-tags*, we train two deep-learning models to predict stance in the text that don't necessarily contain hashtags. We observe that depending on hashtags and the timing of data collection, the accuracy of stance learning could fluctuate. The fluctuation is because of the noisy nature of social-media where hashtags could be used for multiple purposes. To address this issue, given a few labeled examples, we use the multi-armed-bandit (MAB) optimization to find stance-tags that are most suitable for stance learning in the long run. We validate our approach with experiments on a hand-labeled dataset with five different topics, obtaining results comparable to the state-of-the-art. To the best of our knowledge, this is the first work that suggests a principled approach for finding and using hashtags as labels for stance learning.

In this paper, we make three significant contributions to this research area: 1)we propose a way to generate a more extensive dataset for training neural networks (NNs) for stance learning and suggest a method to find specific hashtags (stance-tags) that carry stance information; 2) we find that not all stance-tags are useful for such learning depending on the time of data collection and the models used; and 3) we suggest a multi-armed bandit (MAB) based reinforcement learning approach that maximizes the long-term performance of these models by finding stance-tags that are most suitable for stance learning. In the future work section, we propose a search strategy that exploits that similarity between hashtags.

This paper is organized as follows. In section 'Related Work', we discuss prior work that influenced this research. In section 'Dataset', we discuss the dataset that is used in this study to evalute the performance of our models. Here we also describe how we split the human-labeled dataset for tuning and validation. Then, in section 'Methodology', we discuss our research methodology. This section is further sub-divided into three subsections explaining three parts of this research. Each subsection has its own results paragraph. Finally, at the end, we discuss the conclusions of this study and describe some ideas for future work.

Related Work

Stance learning lies at the intersection of many different fields. In particular, we discuss the relation between stance learning and sentiment mining and show how they are different. In addition, we also discuss some relevant work in the area of *weakly supervised learning*, which is the main approach taken in this research. Finally, we discuss *multi-armed-bandit* optimization approach, which is also another topic relevant to this work.

Stance Learning and Sentiment Mining

Du Bois2007 defines the act of stance taking as 'Stance is a public act by a social actor, achieved dialogically through overt communicative means, of simultaneously evaluating objects, positioning subjects (self and others), and aligning with other subjects... ' (page 163). Most data-mining research on stance considers communicative means (text) for learning stance of users in debates and social-media conversations. Using a manually annotated corpus, Somasundaran and Wiebe2010 constructed an arguing lexicon to recognize stance in on-line debates. They used both the arguing-lexicon based features and sentiment based features to build an SVM classifier resulting in above 60% accuracy on debate posts in four domains 'Gun rights', 'Gay rights', 'Abortion' and 'Creationism'. Ozer et al.2016 tried to cluster politically motivated users into communities. They used structural balance theory to build a user-connectivity network using endorsements, and tried three non-negative matrix factorization approaches to detect communities. Tu et al.2017 proposed context-aware embeddings that attempt to use semantic relationships between users to preserve the diverse roles of interacting users.

For social-media conversations, Mohammad et al.2017 built a stance dataset using Tweets and organized a SemEval competition in 2016 (Task 6). Many researchers (Augenstein et al.2016; Liu et al.2016; Wei et al.2016) used the dataset and proposed algorithms to learn stance from data. However none of them exceeded the performance achieved by a simple algorithm (Mohammad et al.2017) that uses word and character n-grams, sentiment, parts-of-speech (POS) and word embeddings as features. The authors used an SVM classifier to achieve 0.59 as the mean f1-macro score.

One of the topics that is closely related to stance learning is sentiment mining (Pang et al.2008). Prior work has shown that sentiment could be used as one of the features for stance learning (Mohammad et al.2017). However, users can convey stance in text that does not contain any sentiment. As described in (Du Bois2007), in stance detection, we learn stance towards a target but stance learning also involves source of stance. Often source of stance is not evident and only text is available to learn stance. Though we utilize sentiment in our approach, the focus of this work is on stance learning from text data.

Weakly Supervised Machine Learning

Nearly all current state-of-the-art models for visual perception rely on supervised training e.g. (Krizhevsky et al.2012) for object detection. The standard approach in supervised-training uses large datasets like Imagenet (Deng et al.2009) for pre-training a neural-network and then fine tuning the model for task-specific goal. Recent works have explored the use of weakly-supervised training using social-media hashtags (Mahajan et al.2018). Mahajan et al.2018 shows that supervised pre-training can lead to state-of-the-art results that are comparable to supervised training.

In the text domain, weakly-supervised training is more common. Go et al.2009 used positive and negative smiley emoticons to train a sentiment classifier. Another, active area of research is learning word-embeddings in which unsupervised learning is used to learn word-vectors that preserve syntactic and semantic similarities in words (Mikolov et al.2013). Kiros et al.2015 used unsupervised learning to train sentence encoders. Using sentences for training a model that tries to reconstruct the surrounding sentence, the authors learn an sentence-encoder that was useful in many downstream tasks. Researchers have also tried hashstags for weak-supervision. As discussed earlier, (Mahajan et al.2018) recently used hahstgas related to images for object detection. In text analysis, hashtags have been used in sub-event discovery (Xing et al.2016) and aspect-based opinion mining (Lim and Buntine2014).

Though weak labels and unsupervised learning are common in many areas of text mining like sentiment mining, for stance-learning most known work use labeled datasets. Mohammad et al.2017 briefly discussed the idea of using unlabeled tweets for improving their classification results. They explored two methods: first using additional data for training and second using word-association as additional features. They found that using data from certain hashtags as additional training data, they could improve their f-score by up to 4 percentage points (for one target, but minimal for others). In comparison to their work, the focus of our current work is stance learning using weakly labeled data.

Multi-Armed Bandit Problem

In simple terms, a multi-armed Bandit (MAB) problem involves allocating a fixed set of resources between competing choices to maximize the long-term gain (Sutton et al.1998) (chapter 2). Such problems often involve exploration vs exploitation trade-off (Audibert et al.2009). For example, a gambler playing with a number of slot machines ('one bandit with many arms') has to decide which machines to play and how many times to maximize his gain. This problem appears in different forms like the contextual-bandit and the collaborative-bandit. In this research, we design our hashtags selection problem as a multi-armed bandit problem. In each iteration, a k-armed bandit has to choose between the arms to use. For this type of problem, researchers have explored many optimization approaches: e.g. Epoch-greedy (Langford and Zhang2008) , upper-confidence-bound (UCB) values (Auer and Ortner2010), softmax (Sutton et al.1998) (chapter 2) and Bayesian (Scott2010) optimization strategies.

Dataset

In this section, we describe a human-labeled datatset that was built in prior research (Mohammad et al.2017) and was used for a competition in SemEval 2016. The dataset is summarized in Tab. 1.

This dataset has no development/validation set. As we needed data for validation and tuning, we split the training data into two equal sets: one for tuning and another for validation. In the rest of the paper, whenever we refer to validation set, we mean this validation data. Our training set is the data we collected from Twitter using hashtags.

Topic	Train	Test
Atheism	513	220
Climate Change	395	169
Feminist Movement	664	285
Hillary Clinton	689	295
Legal. Abortion	653	280

Table 1: Human labeled datasets used in this research

Methodology

Our research can be divided in four parts:

1. Stance-Tag selection: We find potential hashstags that can be used for weak-supervision by exploiting sentiment information present in text. This step results in a list of pro hashtags and anti hashtags. We call these hashtags stance-tags.
2. Stance classification using deep-learning models: Using data collected by searching stance-tags on Twitter, we train two state-of-the-art neural-network architectures, a long-short term memory (LSTM) and a convolutional neural network (CNN), for stance classification. We evaluate the trained models by verifying their performance on an existing human-labeled datasets.
3. Multi-armed bandit optimization for the optimal stance-tag selection: Because of inconsistency in results for models trained on data collected by searching stance-tags, we propose a strategy to find the most suitable stance-tags for training the neural-network models. This step further improves the performance of the models learned in step 2.
4. Because hashtags are often used together, we propose a search optimization approach that can exploit the similarity between hashtags. This idea is further elaborated in the Future Work section.

Next, we elaborate on each of the parts in details.

Stance-Tag Selection

The research question in this part can be formulated as:

Given a *Topic T* and a large amount of *text data D* (Tweets) collected for *T*, find the *Top-Hashtags* present in D that are relevant to T and whose associated text have high positive or negative *mean sentiment*.

Evans2016 studied the usage of hashtags on social media. By analyzing conversations on Twitter, the author found that users of social media use hastags to take stance and stance carrying hashtags are commonly used at the end of the posts. We further explore the idea of using sentiment signal to find hashtags that contain stance. Because sentiment is useful in stance in stance learning (Mohammad et al.2017), it is expected that higher sentiment in text would relate to stance that is in favor or against the topic. Thus, if we search for a topic (say 'abortion'), retrieve all end-tags (hashtags present at the end of text in posts), then analyze sentiment for the text associated with each endtag, and plot them in order of mean-sentiment score (as shown in Fig. 2), we expect the stance-tags to be have higher mean sentiment score. It's not always the case that a pro-stance-tag has +ve sentiment as it depends on how hashtags are getting used. But, in general, we find that stance-tags have higher mean sentiment (+ve or -ve).

We conducted this experiment for all five topics in the SemEval dataset. For each topic, we downloded data from Twitter [1]. We performed basic cleaning of data which involved excluding very-short sentences and excluding all sentences that have more than three hastags. We then obtained all endtags in the data. For each end-tag, we randomly selected 200 text posts to evaluate their sentiment score using a sate-fo-the-art neural network as proposed in Radford et al.2017. Before evaluating sentiment , we cleaned the text by removing all hashtags, @ mentions and URLs. Figure 2 shows mean sentiment trend for the 'abortion' topic. As we can observe, hashtags with higher absolute sentiment score also carry stance. Note that not all hastgas that have higher mean-sentiment carry stance about the topic. In fact, we found a number of hahstags that were off-topic. Hence, by visual inspection, we removed many hashtags that were not relevant. In Tab. 2, we list the top pro and anti hashtags for each topic. We refer these hashtags as *stance-tags* in the rest of the paper.

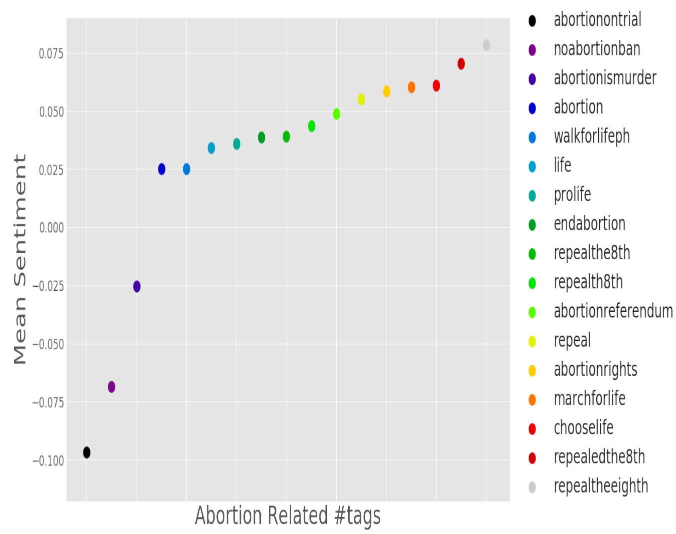

Figure 2: Mean sentiment score for various hashtags related to 'abortion' topic. As explained in this paper, twitter data obtained by searching some of these hashtags can be used for training stance-learning models.

Stance Classification using Deep-learning Models

The research question in this part is formulated as:

Given a *Topic T* and a large amount of *text data D* (Tweets) collected by searching for *stance-tags* on Twitter, build and evaluate *deep-learning models M* that uses data D

[1] https://developer.twitter.com/en/docs/tweets/filter-realtime/guides/connecting.html

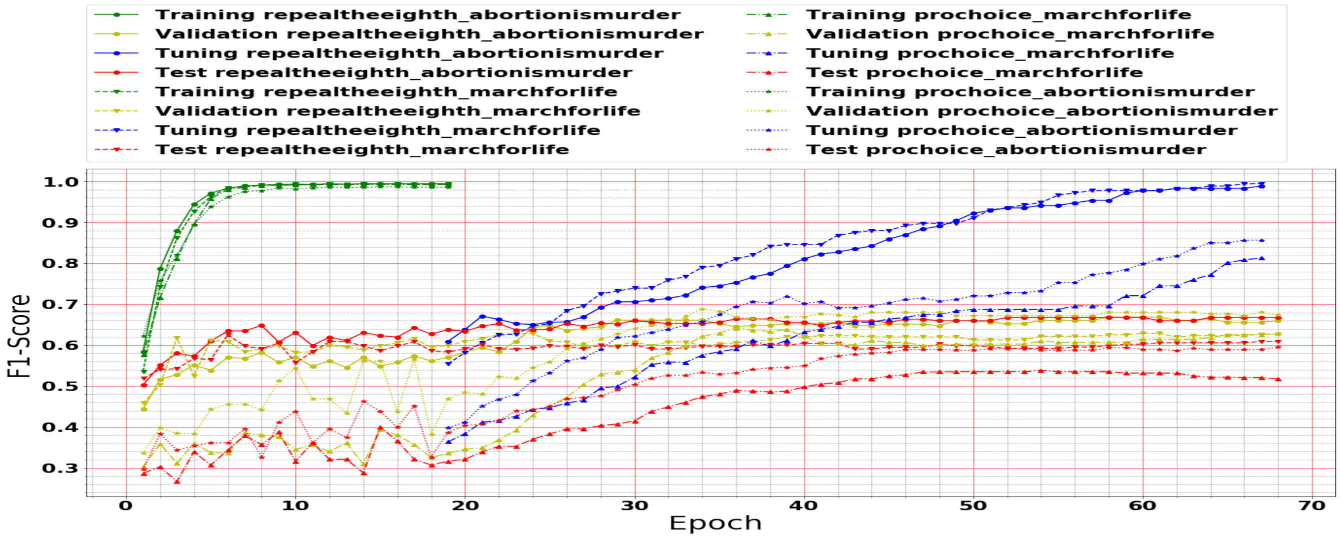

Figure 3: CNN training for Abortion: Trend of training, tuning, validation and test f1-score for different combination of pro and anti stance-tags. Note that training uses unlabeled Twitter data and is run for 20 epochs, then tuning is performed for another 50 epochs using a subset of SemEval training dataset.

Topic	Top Pro stance-tags	Top Anti stance-tags
Atheism	GodIsGood, IBelieveInGod, GodIsLove	Humanist, GoodWithoutGod, Atheist
Climate Change	DefendClimate, RiseForClimate, ClimateChangeIsReal	ClimateChangeHoax, ClimateDeniers, ClimateCrisis
Feminist Movemt.	Feminism, HeForShe, MeToo	WomenAgainstFeminism, AntiFeminist, FuckFeminism
Hillary Clinton	LoveHillary, IMWithHer, StillWithHer, IMStillWithHer	LockupHillary, IndictHillary, CrookedHillary, HillaryForPrison
Legality of Abortion	ChooseLife, MarchforLife, AbortionIsMurder	AbortionOnTrial, ProChoice, RepealTheEighth

Table 2: List of a few stance-tags for different topics. The list of topics in the first column is taken from Mohammad et al.2017

for training. In addition, also explore the effects of using SemEval training data for tuning the performance the models.

In this step, we first use stance-tags to retrieve more data from Twitter. We again clean the downloaded tweets, and filter them by endtags. If a tweet has two or three stance-tags at the end of tweet, we consider the tweet as a part of all of its stance-tags. We exclude any tweets containg more than three end-tags. This process results in one text file for each stance-tag, where each line is cleaned text from a single tweet. The data is then used to train two deep-learning models, a Bi-Directional Long Short Term Memory (LSTM) and a Convolutional Neural Network (CNN). Finally, the models are evaluated on the hand-labled datasets described earlier.

Next, we discuss LSTM and CNN model in detail and present their performance on SemEval dataset.

$$L(y,p) = -\frac{1}{n}\sum_{i,j} y_{ij}\log(p_{ij}) \tag{1}$$

For the classification task, we define our overall objective function using binary cross entropy loss as can be seen in Equation 1, where $i \in n$ samples, $j \in 2$ classes (Pro/Anti), y is the (one-hot) true label, p is the prediction label.

F1- score is used as the metric for evaluating the performance on the models. We choose F1-score as that is the metric of choice in Mohammad et al.2017, and also because F1-score is more representative of the actual performance when the dataset classes are not balanced. We define F1 score as the average of F1 score for two classes. F1-score is defined as the harmonic mean of precision and recall.

$$F1_{score} = \frac{F1_{pro} + F1_{anti}}{2} \tag{2}$$

Convolutional Neural Networks Kim2014 used convolution neural networks (CNN) for sentence classification, and showed that a simple CNN could be used to get sate-of-the art results in sentiment analysis. Since stance learning is closely associated with sentiment learning, we built a similar CNN model. The model is built using PyTorch library [2], and used one dimensional convolution layer with Glove embedding [3] as input, adds a max-pooling operation, adds a flattern, a droupout (0.5) and then adds a fully connected

[2]https://pytorch.org/

[3]https://nlp.stanford.edu/projects/glove/

layer for the final classification. The model is trained using Adam optimization with learning rate of 10^-3. For tuning, we again used Adam optimization but with a lower learning rate of 10^-4. For training, we used the cleaned data collected using stance-tags as described earlier. After training, we also tune the model by using SemEval training dataset.

Bi-Directional LSTM To build an LSTM (Hochreiter and Schmidhuber Hochreiter and Schmidhuber1997) model, we again used PyTorch library. Our LSTM model takes Glove embedding as input, adds 50 dimension Bi-LSTM layer, a linear dense activation layer and uses binary-cross-entropy loss function. The model's training, tuning and test followed the same steps as described for the CNN model.

Results Using the training data collected from Twitter, we first train the neural-networks that predicts 'Pro' and 'Anti' stance. The model is then tuned on the SemEval training dataset. In Fig. 3, we show the f1-score after each iteration. In the figure, we fist trained the model for 20 iterations, then tune the model for 50 iterations. For different pro and anti stance-tags combinations, we plot the results for training, tuning, validation and test. Figure 4 shows the trend of F1 score for training, tuning, validation and test performance with epochs for the LSTM model with a few combinations of stance-tags relevant to 'Abortion'. We summarize the best results for 'Abortion' in Table 3 which shows the top test f1-score for different stance-tags combinations. Results for different topics are shown in Table 4 which includes pre-tuning and post-tuning f1-scores. Note that for all topics, only unlabeled Twitter data is used for training which is run for the first few epochs till training converges. Then tuning is performed using a subset of SemEval training dataset for another few epochs (until it converges). Tuning does not necessarily improve the performance on validation/test set (as seen in Fig. 4) indicating that weakly supervised training was useful. In fact, in this plot, the best f1-score on test data (red lines) is observed while training on the unlabeled Twitter data. Fig. 3 also highlight that not all hashtags are useful for stance learning e.g. the combination of 'prochoice' and 'marchforlife' barely improved on traing and made minor gains while tuning. For the same combination, we can also see an inconsistency in the performance with training. The inconsistency though parlty because of the noisy nature of social-media, but also because of multiple possible usage of certain hashtags.

Pro s-tag	Anti s-tag	CNN F1 (valid/test)	LSTM F1 (valid/test)
repealtheeighth	abortionismurder	0.67/0.67	0.66/0.62
repealtheeighth	marchforlife	0.63/0.60	**0.72/0.68**
prochoice	marchforlife	0.64/0.49	0.72/0.62
prochoice	abortionismurder	0.64/0.49	0.72/0.60

Table 3: Classification f1-score on SemEval dataset 'Abortion' topic. Pro s-tag implies pro (in favor) stance-tag and Anti s-tag implies anti (against) stance-tag used for training. Test f1-score is not the best f1-score we obtained on the test set but rather the f1-score on test set corresponding to the best f1-score on the validation set.

Stance-tag Selection as Multi-armed Bandit Problem

In the last part of this research, we explored a way to find stance-tags and used the data collected using stance-tags to train two deep-learning models. While evaluating the models, we found that data from only some stance-tags are useful for stance-learning. Moreover, we also found that data collected from Twitter show a lot of temporal variation. For example, we show the variation of F1-score for different hashtags combination the topic 'Hillary Clinton' for multiple training episodes based on data with different collection timing (Fig. 8). As we can observe, the distribution for hashtags suggest that the tweets using these hashtags are noisy. This is likely because political discussions on Twitter are often very noisy and similar hashtgas are used for variety of discussions. Depending on the timeframe of data collection, models trained of this data may get inconsistent performance. This is expected as the usage of hashtags change over time and some hashtags could be used for multiple purposes. Given this variability, can we still use social-media data and, more importantly, can we find the stance-tags that are optimal for training the models. Unlike a supervised learning setting in which the correct labels are known before training, we don't know the stance-tags that would lead to good performance. However, after selecting a pair of pro and anti stance-tags, we can evaluate the performance of the learned models given some labeled data. To maximize the long-term performance of these models, we explore a reinforcement learning approach.

For this part, we formally define the research question as:

Given *k pro and k anti stance-tags* and some labeled examples, select one *pro stance-tag* and one *anti stance-tag* that minimizes regret (defined later) over multiple training episodes.

Solving a reinforcement learning problem needs finding a policy that leads to maximum reward (or lower the regret) in the long run based on actions taken. In our problem setting, we use the classification performance on the validation set of data as the reward. Action, which is the choice of selecting a stance-tag-pair (one +ve and one -ve hashtags), is independent of the state of the agent i.e. we always strive to maximize the validation-set performance and hence our problem is a non-associative reinforcement learning problem where there are no states. Therefore, we use the multi-armed bandit (MAB) optimization approach. MAB problems can be solved by many possible strategies. These strategies attempt to strike a balance between exploration and exploitation. In exploration, the algorithm searches for the best choice i.e. in our problem the optimal stance-tag to use. In exploitation, the algorithm uses the current best choice (current best hashtag) to get the maximum reward. Exploration and exploitation are competing goals, and often finding the optimal

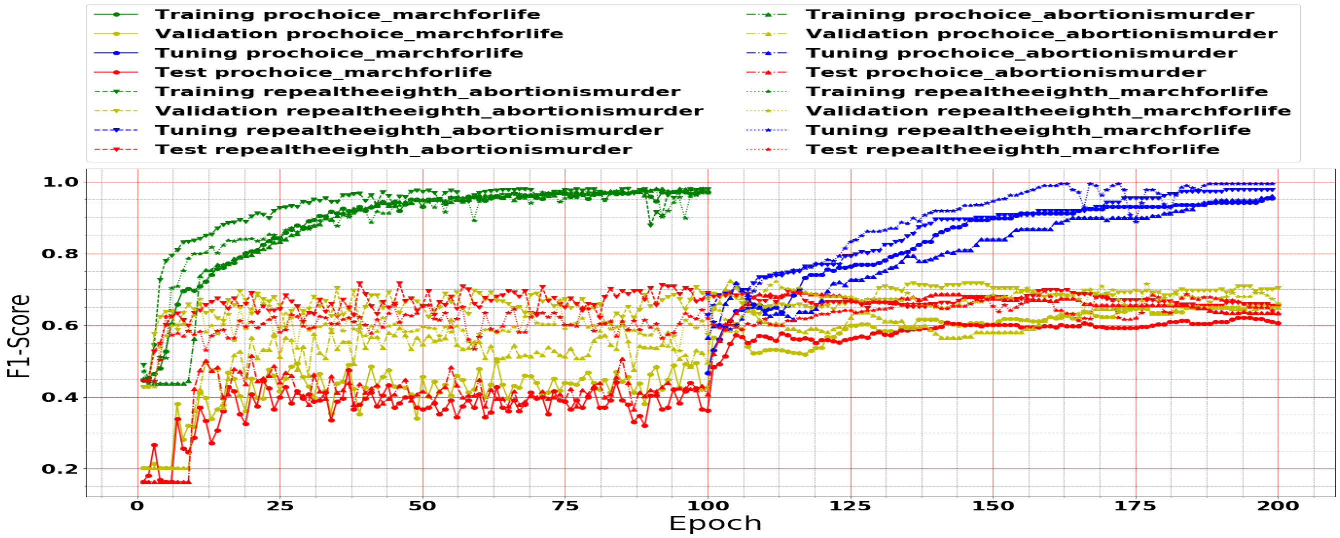

Figure 4: LSTM training for Abortion: Trend of training, tuning, validation and test f1-score for different combination of pro and anti stance-tags. Note that training uses unlabeled Twitter data and is run for 100 epochs, then tuning is performed for another 100 epochs using a subset of SemEval training dataset. Tuning does not necessarily improve the performance on validation/test set indicating that weakly supervised training was useful. In fact, in this plot, the best f1-score on test data (red lines) is observed while training on the unlabeled Twitter data.

choices is not trivial. Regret is commonly used as a metric to evaluate the choices made over multiple episodes. We try four different strategies that have shown promising results in prior research. Through multiple training episodes, we aim to minimize the regret in long run.

In a simple case, when there are k possible actions, MAB optimization involves choosing an arm of a k-armed bandit with action a that leads to reward r. k arms have reward probabilities defined as $\{\theta_1, ...\theta_k\}$ which is typically not known in advance. The optimization leads to minimization of regret defined as:

$$Regret = E\Big[\sum_{t=1}^{T}(\theta^* - Q(a_t)\Big] \tag{3}$$

where $\theta^* = Q(a^*)$ is the reward for the optimal action a^* and $Q(a) = E[r|a]$ is the expected reward for action a. In our problem, we need to select one +ve stance-tag and one -ve stance-tag. Given, the pro-stance-tag and anti-stance-tag, we can evaluate the performance of a deep-learning model on validation data which could be used as reward. We design an algorithm that simultaneously tries various combinations of pro and anti stance-tags to find the optimal. The algorithm (see Algorithm. 1) , involves two MABs. The first MAB learns the best pro-stance-tag and the second MAB learns the best anti-stance tag. The algorithm uses the validation performance (by training the classifiers discussed in the last section) as reward. Based on the strategy used, MAB suggests an action (i.e. which stance-tag to use) in each timestep. Total regret in each iteration is the sum of regrets estimated by the two MABs.

Next, we describe four strategies that are commonly used in MAB optimization.

Epsilon (ϵ) greedy Policy ϵ-greedy (Langford and Zhang2008) selects the arm with highest reward with probability $1 - \epsilon$ and a random arm with probability ϵ. Thus, action

$$a_t = \begin{cases} argmax_{a \in A}\hat{Q}_t(a) & \text{with prob.}(1-\epsilon) \\ a \in A, & \text{with prob.}\epsilon \end{cases}$$

where $\hat{Q}_t(a)$ is the mean reward associated with action a.

Upper-Confidence-Bound (UCB) Action Selection Policy UCB (Auer and Ortner2010) strategy allows for better exploration by giving higer probability to actions for which reward estimate is not available.

$$a_t = argmax_{a \in A}\hat{Q}_t(a) + c\sqrt{\frac{\ln t}{N_t(a)}} \tag{4}$$

where $\hat{Q}_t(a)$ is the mean reward associated with action a, $N_t(a)$ denotes the total number of times a has been selected before time t and c is a parameter to control exploration.

Softmax Policy In softmax policy , a preference H_t vector is defined for each action a.

The probability of taking an action is defined in terms of preference as:

$$\pi_t(a) = Pr\{A_t = a\} = \frac{\exp H_t(a)}{\sum_{b=1}^{k}\exp H_t(b)}$$

Topic	Pro s-tag	Anti s-tag	CNN F1 Pre-tuning (valid/test)	CNN F1 Post-tuning (valid/test)	LSTM F1 Pre-tuning (valid/test)	LSTM F1 Post-tuning (valid/test)
Atheism	godisgood	atheist	0.484/0.479	0.484/0.479	0.434/0.453	0.434/0.451
Climate Change	actonclimate	climatehoax	0.736/0.564	0.714/0.612	0.611/0.479	0.686/0.479
Feminist Movemt.	heforshe	womenagainst-feminism	0.281/0.194	0.528/0.424	0.522/0.54	0.624/0.534
Hillary Clinton	imwithher	neverhillary	0.56/0.498	0.56/0.498	0.483/0.459	0.483/0.459
Legal. of Abortion	repealtheeighth	abortionismurder	0.594/0.647	0.669/0.668	0.708/0.667	0.717/0.669

Table 4: Pre-tuning and after-tuning classification mean f1-score on SemEval dataset using data from top +ve and -ve stance-tags for training the models. +ve s-tag imples positive stance tag and -ve s-tag implies negative stance tag used for training. No SemaEval dataset is used for training. Evaluation is on validation and test set of SemEval. Best performance on test set is the performance obtained using the model that performs the best on the validation set.

Algorithm	Atheism	Climate Change	Feminist Movemt.	Hillary Clinton	Legal. of Abortion
a. Rf1 n-grams	65.2	42.4	57.5	58.6	66.4
b. a + Sentiment	65.2	40.1	54.5	60.6	61.7
c. Rf1 Best	68.3	43.8	**58.4**	**64.7**	66.9
d. Double-MAB	**69.1**	**61.2**	53.4	49.8	**68.1**

Table 5: Results. F1 score for different algorithms. Rf1 refers to Mohammad et al.2017. The top performers for each topic is highlighted in bold font. Double-MAB is the algorithm that we proposed that selects the best combination oh stance-tags to improve the classification accuracy.

To find the optimal policy, we update the preference in each episode using stochasitic gradient ascent:

$$H_{t+1}(A_t) = H_t(A_t) + \alpha(Q_t - \tilde{Q}_t)(1 - \pi_t(A_t)) \quad (5)$$

$$H_{t+1}(a) = H_t(a) + \alpha(Q_t - \tilde{Q}_t)(\pi_t(a)), \text{for } a \neq A_t \quad (6)$$

where α is step size parameter. We refer readers to (Sutton et al.1998) (chapter 2) for details.

Bayesian Bandit Policy Bayesian bandit strategy assumes a prior rewards distribution and updates the distribution with trials. We use beta distribution for exploration and exploitation. We refer readers to (Scott2010) for details on bayesian bandits.

Results For all five topics, we used the Double MAB optimization algorithm to find the best stance-tag-pair. In the process of optimization, for each time step, we store the actions (choice of stance-tags), maximum arm probability (confidence in stance-tag selection) and regret (as described earlier). Because of limited space, we only show the trend of regret (Fig. 5) , arm selection probability (Fig. 6) and actions (Fig. 7) for 'Legal. of abortion' topic. For other topics, we present the results for the best stance-tag-pair selected by MAB in Tab. 5. As shown in Fig. 7, for the 'Abortion' topic the Anit-MAB returned a single hahstag 'AbortionIsMurder' as the optimal hashtags. However, the Pro-MAB returned different stance-tags depending on the strategy used. In case of conflicts like this, we can use the arm selection probability as shown in Fig. 6 that reflects the confidence in selecting the hashtag. In most cases, MAB returns a single optimal pair of pro and anti stance-tags. With the optimal-pair stance-tags, we again train our CNN and LSTM models to find the best performance on the test dataset. In Table. 5, we summarize the final results obtained by training models based on optimal stance-tag-pair. As we can observe, there is an improvement in the f1-score for two cases ('Atheism' and 'Abortion') compared to random choices for pro and anti stance-tags made in the last part. This is expected as we have an optimization routine to select the better stance-tag pairs. For 'Hillary Clinton' our algorithm is under-performing by a large margin, so we further anlysed it. We think this poor preformance could be due two two reasons: a) The time difference in SemEval data collection (in 2016) and our data collection (in 2108) b) The noisy nature of political conversations on Twitter (see 8). For three topics, we are able to get better than the performance of Mohammad et al.2017. Note that Mohammad et al.2017 algorithm has better performance that all seventeen teams that participated in SemEval 2016 on this task.

Conclusion

We started with the goal of using social-media hashtags for stance learning. To prove that certain hashtags can be used as weak labels for stance-learning, we first used sentiment in the text to identify a few potentially useful hashtags, which we call as stance-tags. Stance-tags can be used to train classifiers that are used to predict pro/anti stance in the hand-labeled dataset. However, a random combination of a pro-stance-tag and an anti-stance-tag may not result in a good performance (measured as f1 score). To resolve this, we pro-

Algorithm 1 Double MAB Problem: Given the set of pro-stance tags and anti-stance tags, Pro-MAB find the best pro-stance-tag and Anti-MAB finds the best anti-stance-tag,

Require: MAB $mabPro$, MAB $mabAnti$, proHashtags H_p, antiHashtags h_a, timesteps T
for $i = 1 \rightarrow T$ **do**
  # Get Pro-MAB strategy
  $strategyPro = mabPro.strategy$
  # Choose pro-hashtag
  $h_{pi} = H_p[strategyPro.choice]$
  # Get Anti-MAB strategy
  $strategyAnti = mabAnti.strategy$
  # Choose anti-hashtag
  $h_{ai} = H_a[strategyAnti.choice]$
  # Get f1 for the selected pro- and ati- hashtags
  $f1_{validation}$ = *CNNClassification(* $h_{pi}.data, h_{pi}.labels, h_{ai}.data, h_{ai}.labels)$
  # Set MABs reward
  $mabPro.reward = f1_{validation}$
  $mabAnti.reward = f1_{validation}$
  # Estimate regret
  $mabPro.estimateRegret()$
  $mabAnti.estimateRegret()$
  # Calculate total regret
  totalRegeret = $mabPro.estimateRegret() + mabAnti.estimateRegret()$
end for
=0

posed a multi-armed bandit (MAB) optimization approach that uses two MABs to find the optimal pair of positive and negative stance-tag-pair. Using SemEval hand-labeled dataset on five topics, we demonstrate that our approach can achieve performance that is comparable to the state-of-the-art. We propose an approach that allows collecting a large amount of weakly-labeled data for new topics in an inexpensive way. Thus, we don't just highlight the feasibility of using hashtags for stance-learning, we also show that weak labels could be used to build models that perform as good as or better than models trained on a human-labeled dataset. Consequently, we transformed stance learning problem from a supervised learning to weakly-supervised learning that require human-labels only for validation. We hope that this approach will extend the stance learning to newer topics, thereby, enabling researchers to study topics like polarized communities and echo-chambers in a new light.

Limitations

There are several limitations to this work. In particular, not all tweets have hashtgas. Since our method is based on extracting stance signal from hashtgas, such tweets can't be used for training using this approach. At the same time, the test tweets do not need to have hashtags as the learned classifiers do not need hashtgas. If hashtags appear in test tweets, they are treated as another word in tweet text. The other limitation of this work is the manipulation of hashtag, which is commonly called hashtag high jacking - which

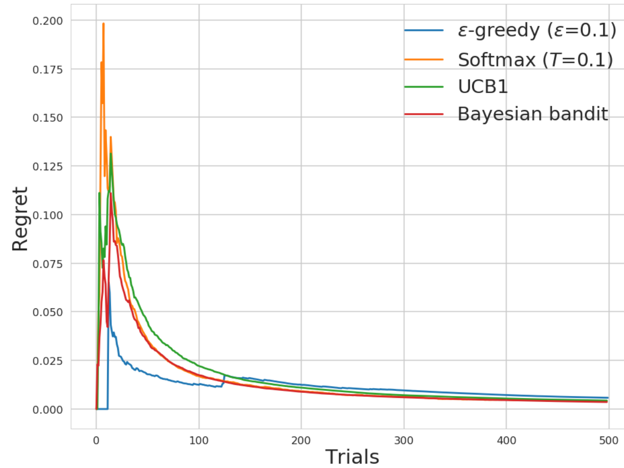

Figure 5: Total Regret of Double-MAB with trials. For the Double-MAB, we sum up the regret estimated by Pro and Anti MAB to get the total regret.

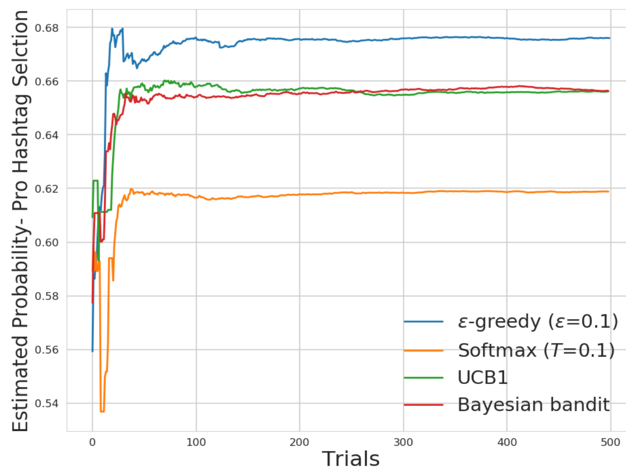

Figure 6: Arm selection probability of Pro-MAB for different MAB strategies.

bots and trolls do frequently on Twitter. In this research, we don't handle such phenomenon. In addition, we have assumed that there is sufficient data associated with each hashtag for training our CNN and LSTM models. Because the usage of hashtgas in a given time interval depends on a lot of factors, it is possible to have insufficient training examples for some hashtags. In future, we would like to provide a table that compares the classification performance of different hahstags and the number of examples used for training.

Future Work

In this research, we assumed that hashtags are independent. However, hashtags are often related to each other. For example, many tweets contain more than one hashtags, which implies hashtags are not completely independent. The hashtags that appear together are likely to be similar to each other in

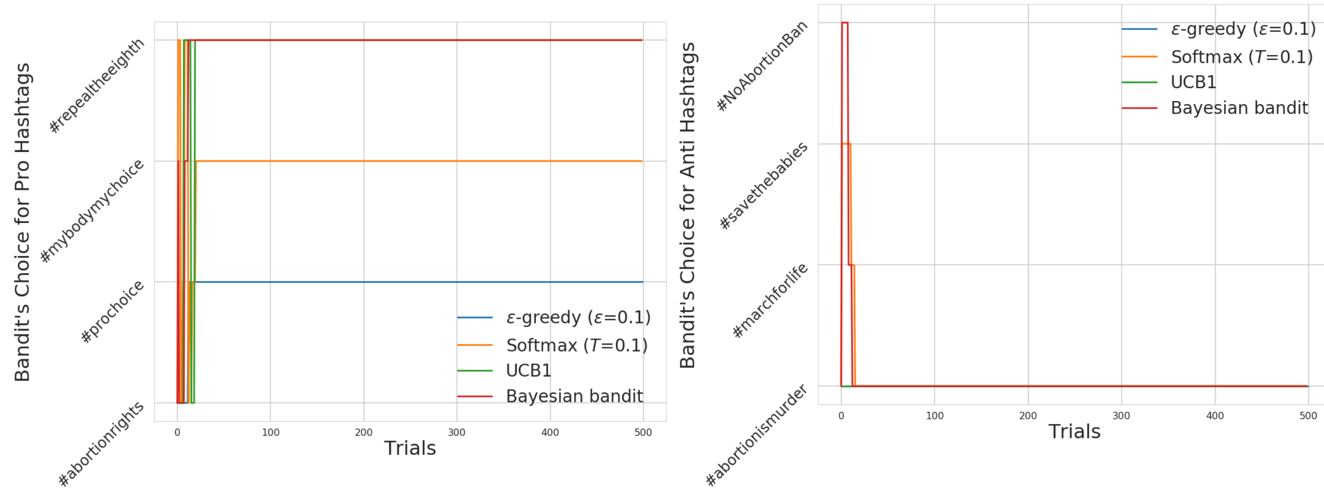

Figure 7: Choice of stance-tags made by MAB over trials for 'Abortion' topic. The left figure shows the choices of pro stance-tags and the right figure shows the choices of anti stance-tags.

their meanings. We can use this similarity measure to create hashtags embeddings space which can then be used to find top pro-hashtag and top anti-hashtag using some stochastic search in the embedding space. Here we outline the steps that we plan to take in this approach.

Create Hashtags Embedding

The goal is to create a hashtag embedding space in which two hashatags that are more similar in usage are also nearby in the embedding space. For this, we first create a hashtags graph (9) where edge-weight reflect the occurrence frequency. We then used this graph to create an embedding space using the Wrod2vec algorithm. The Word2vec algorithm takes sentences as input which is our case is the series of hashtags used at the end of tweets. The embeddings are visualized in Fig. 10 for the topic 'climate change'. As we can observe, many embeddings appear to concentrate on one one side of the plot meaning many hashtags were used in a similar way.

Cross Entropy Method for Searching in the Embedding Space

After building the embeddings space, our goal is to use a search algorithm that can search in the embedding space for the top pro and anti hashtags pair. For this, we plan to use cross-entropy method for this search. The problem can be formulated as:

$$ph^*, ah^* = \underset{(ph,ah)}{\operatorname{argmax}} S(ph, ah), ph, ah \in hashtags \quad (7)$$

where S is a function that maps ph (pro-hahstag) and ah (anti-hashtags) data to the classifier model's F1-score on a validation set.

Cross entropy method is a monte-carlo method for sampling and and optimization. In this approach, we first draw samples from a source distribution and minimize the cross-entropy between the source distribution and the target distribution. In our problem, the source distribution is the word distribution of tweets obtained by selecting a pro-hashtag and an anti-hashtag. The target distribution is the word distribution in human labeled examples used for validation.

Besides, the chnages proposed above, we would also like to make some modifications to the problem at hand. In our study, we considered only two-class (pro/anti) classification. In the future, we would like to include a 'neutral' class in classification. Even better, if we can model the problem as a regression in which the stance ranges from -1 to +1. A varying score from -1 to +1 best reflects stance that people take on topics. We also plan to apply our approach to find and study polarized communities.

Acknowledgement

I would like to thanks Prof. Zico Kolter and Prof. Kathleen M. Carley for their valuable advice throughput the project. This work was supported in part by the MURI Award No. N000140811186, MURI Award No. N000141712675 and the Center for Computational Analysis of Social and Organization Systems (CASOS). The views and conclusions contained in this document are those of the authors only and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Research, or the U.S. government.

References

Icek Ajzen and Martin Fishbein. 1977. Attitude-behavior relations: A theoretical analysis and review of empirical research. *Psychological bulletin* 84, 5 (1977), 888.

Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. 2009. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science* 410, 19 (2009), 1876–1902.

Peter Auer and Ronald Ortner. 2010. UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica* 61, 1-2 (2010), 55–65.

Isabelle Augenstein, Andreas Vlachos, and Kalina Bontcheva. 2016. Usfd at semeval-2016 task 6: Any-target stance detection on twitter with autoencoders. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. 389–393.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd international conference on computational linguistics: posters*. Association for Computational Linguistics, 241–249.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. Ieee, 248–255.

John W Du Bois. 2007. The stance triangle. *Stancetaking in discourse: Subjectivity, evaluation, interaction* 164, 3 (2007), 139–182.

Ash Evans. 2016. Stance and identity in Twitter hashtags. *Language@ internet* 13, 1 (2016).

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1, 12 (2009).

Kazi Saidul Hasan and Vincent Ng. 2013. Stance classification of ideological debates: Data, models, features, and constraints. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*. 1348–1356.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

Kenneth Joseph, Lisa Friedland, William Hobbs, David Lazer, and Oren Tsur. 2017. ConStance: Modeling Annotation Contexts to Improve Stance Classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1115–1124. `http://aclweb.org/anthology/D17-1116`

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *Advances in neural information processing systems*. 3294–3302.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

John Langford and Tong Zhang. 2008. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in neural information processing systems*. 817–824.

Kar Wai Lim and Wray Buntine. 2014. Twitter opinion topic model: Extracting product opinions from tweets by leveraging hashtags and sentiment lexicon. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*. ACM, 1319–1328.

Can Liu, Wen Li, Bradford Demarest, Yue Chen, Sara Couture, Daniel Dakota, Nikita Haduong, Noah Kaufman, Andrew Lamont, Manan Pancholi, et al. 2016. Iucl at semeval-2016 task 6: An ensemble model for stance detection in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. 394–400.

Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. 2018. Exploring the Limits of Weakly Supervised Pretraining. *arXiv preprint arXiv:1805.00932* (2018).

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. 2017. Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)* 17, 3 (2017), 26.

Mert Ozer, Nyunsu Kim, and Hasan Davulcu. 2016. Community detection in political Twitter networks using Nonnegative Matrix Factorization methods. In *Advances in Social Networks Analysis and Mining (ASONAM), 2016 IEEE/ACM International Conference on*. IEEE, 81–88.

Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval* 2, 1–2 (2008), 1–135.

Ashequl Qadir and Ellen Riloff. 2013. Bootstrapped learning of emotion hashtags# hashtags4you. In *Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis*. 2–11.

Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. 2017. Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444* (2017).

Steven L Scott. 2010. A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry* 26, 6 (2010), 639–658.

Swapna Somasundaran and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*. Association for Computational Linguistics, 116–124.

Richard S Sutton, Andrew G Barto, et al. 1998. *Reinforcement learning: An introduction*. MIT press.

Cunchao Tu, Han Liu, Zhiyuan Liu, and Maosong Sun. 2017. Cane: Context-aware network embedding for relation modeling. In *Proceedings of ACL*.

Yuan Wang, Jie Liu, Jishi Qu, Yalou Huang, Jimeng Chen, and Xia Feng. 2014. Hashtag graph based topic model for tweet mining. In *Data Mining (ICDM), 2014 IEEE International Conference on*. IEEE, 1025–1030.

Wan Wei, Xiao Zhang, Xuqin Liu, Wei Chen, and Tengjiao Wang. 2016. pkudblab at semeval-2016 task 6: A specific convolutional neural network system for effective stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. 384–388.

Chen Xing, Yuan Wang, Jie Liu, Yalou Huang, and Wei-Ying Ma. 2016. Hashtag-Based Sub-Event Discovery Using Mutually Generative LDA in Twitter.. In *AAAI*. 2666–2672.

Appendix

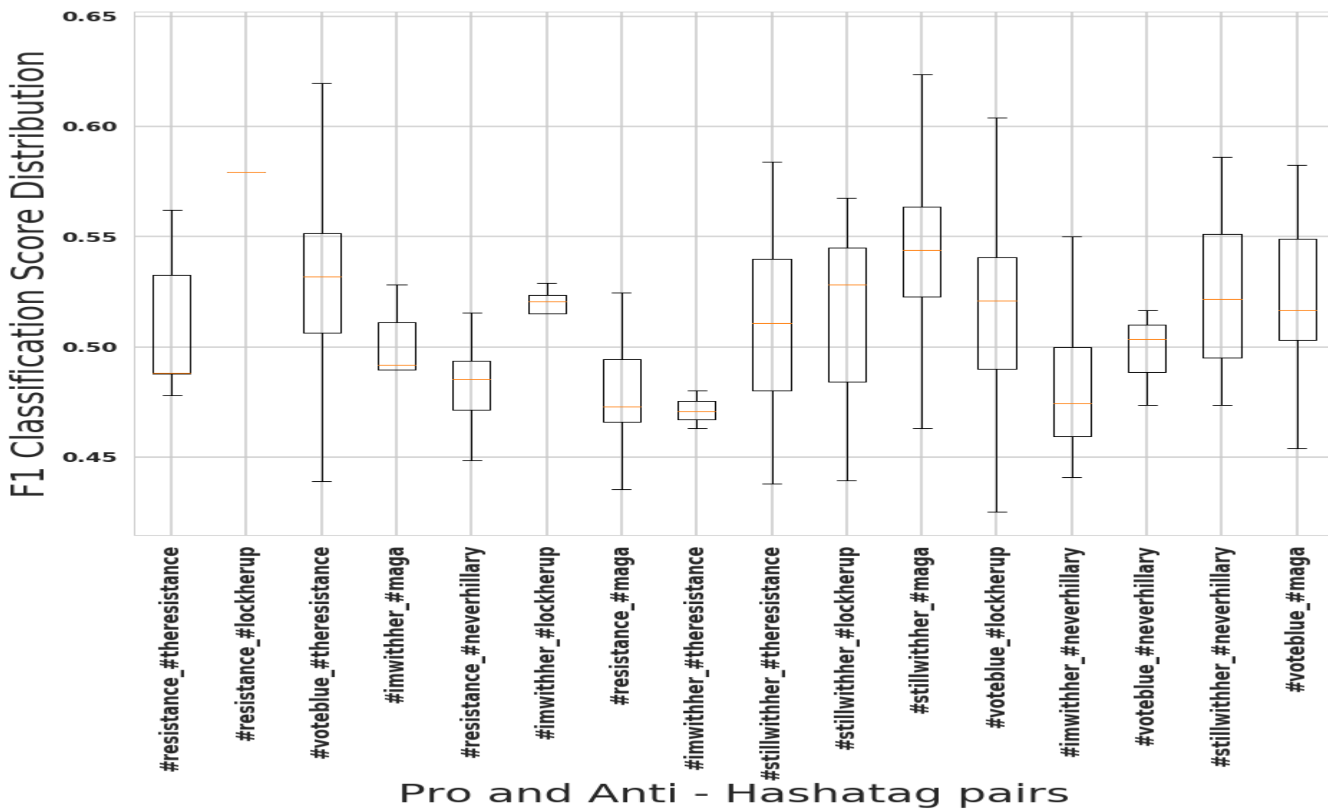

Figure 8: Box plot for F1 classification scores for different pairs of pro and anti stance tags for the topic 'Hillary Clinton'. This is one of the topics that had shown to perform poorly on the human-labeled (test) dataset. Median F1-score is shown as horizontal lines at the median of each box. The boxes show the quartiles and the median's confidence intervals. As we can observe, the distribution for hashtags suggest that the tweets using these hashtags are noisy. This is likely because political discussions on Twitter are often very noisy and similar hashtgas are used for variety of discussions. For example, these days #maga' can be found in any popular US based discussions.

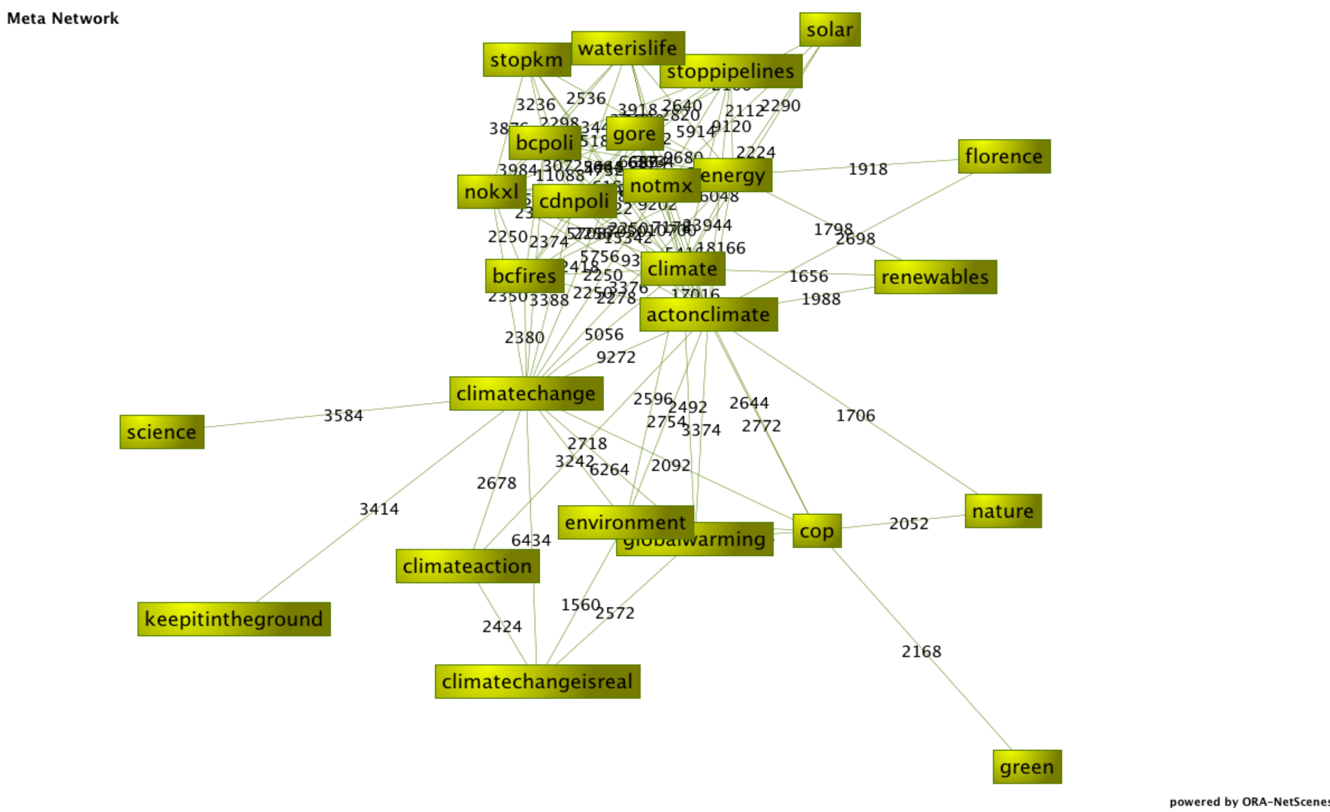

Figure 9: A network of top 25 hashtags (top 100 edges) related to the topic 'climate change' in our dataset. The count of times these hashtags appear together is shown as edge labels.

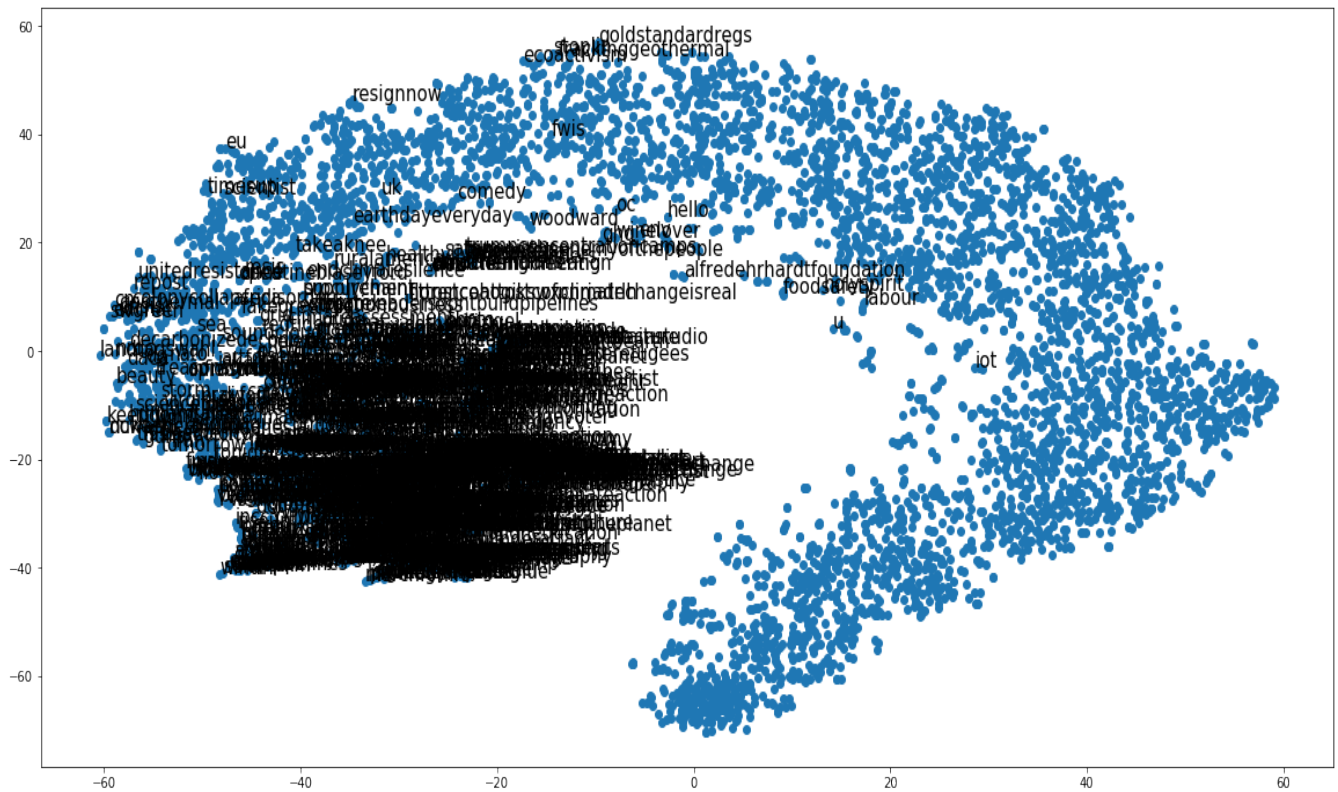

Figure 10: TSNE visualization of hashtags used in the topic ‘climate change’. Here we only show labels for hahastgas that appeared more than 200 times in the dataset. As in other parts of the research. we only used the hashtags that appear at the end of the tweets. We applied Wrod2vec to generate embeddings which internally used sentence were made of only end-hashtags. As we can observe, many embeddings appear to concentrate on one one side of the plot, which means that it is likely that many hashtags are used in a similar way.