From opinion lexicons to sentiment classification of tweets and vice versa: a transfer learning approach

Felipe Bravo-Marquez, Eibe Frank and Bernhard Pfahringer
Department of Computer Science, The University of Waikato
Hamilton, New Zealand

***Abstract*—Message-level and word-level polarity classification are two popular tasks in Twitter sentiment analysis. They have been commonly addressed by training supervised models from labelled data. The main limitation of these models is the high cost of data annotation. Transferring existing labels from a related problem domain is one possible solution for this problem. In this paper, we propose a simple model for transferring sentiment labels from words to tweets and vice versa by representing both tweets and words using feature vectors residing in the same feature space. Tweets are represented by standard NLP features such as unigrams and part-of-speech tags. Words are represented by averaging the vectors of the tweets in which they occur. We evaluate our approach in two transfer learning problems: 1) training a tweet-level polarity classifier from a polarity lexicon, and 2) inducing a polarity lexicon from a collection of polarity-annotated tweets. Our results show that the proposed approach can successfully classify words and tweets after transfer.**

I. Introduction

Twitter[1] is a widely-used microblogging service in which users post short messages, or tweets, limited to 140 characters to express their opinions and thoughts. Automatic analysis of sentiment in tweets has potential applications in a wide range of fields such as business, sports, and politics. However, the brevity of tweets and the range of informal expressions frequently used in them, including slang words, hashtags, and emoticons, make sentiment analysis of tweets a difficult task.

There are two sentiment analysis tasks for tweets that have received substantial attention:

1) **Message-level polarity classification** (MPC) [1], which is the task of classifying tweets into sentiment categories such as positive and negative.
2) **Polarity lexicon induction** (PLI) [2], which is the task of classifying words from a corpus of tweets into sentiment categories.

These two tasks have been successfully tackled using supervised machine learning algorithms by representing the target tweets or words as vectors of features and using hand-crafted sentiment labels for training. A major limitation of supervised approaches is that the annotation of words or tweets based on polarity classes is a time-consuming and labor-intensive task.

Transfer learning refers to the process of improving the learning of a predictive function for a target domain $\mathcal{D}_{\mathcal{T}}$ using knowledge obtained from a related source domain $\mathcal{D}_{\mathcal{S}}$ [3]. Inspired by this principle, we present a tweet centroid model for transferring sentiment knowledge from the word domain $\mathcal{D}_{\mathcal{W}}$ to the message domain $\mathcal{D}_{\mathcal{M}}$ and vice versa.

In our model, we represent tweets and words by feature vectors of the same dimensionality. Tweets are represented using standard natural language processing (NLP) features such as unigrams and part-of-speech (POS) tags, and words are represented by the centroids of the tweet vectors in which they occur. A noteworthy aspect of this approach is its simplicity; yet, despite its simplicity, it yields promising classification performance, as we show in Section IV.

The tweet centroid model (TCM) allows classifiers trained from one of the two above domains to be deployed on data from the other one because both tweets and words can be labelled according to the same sentiment categories, e.g, positive and negative ($\mathcal{Y}_{\mathcal{W}} = \mathcal{Y}_{\mathcal{M}}$). Therefore, a word-level classifier trained from a polarity lexicon and a corpus of unlabelled tweets can be used for classifying the sentiment of tweets (MPC). Likewise, we can train a message-level classifier from a corpus of sentiment-annotated tweets and use it for classifying words into sentiment classes (PLI). Hence, this transfer learning approach is useful in scenarios where either MPC or PLI needs to be solved but it is easier to obtain annotated data from the other domain.

The model is based on the hypothesis that there is a sentiment interdependence relation between words and tweets. This relation, which was first observed in [4] in the case of larger text documents, is defined by the following two statements:

1) The polarity of a tweet is determined by the polarity of the words it contains.
2) The polarity of a word is determined by the polarity of the tweets in which it occurs.

This article is organised as follows. In Section II, we provide a review of related work. The proposed transfer learning approach is described in Section III. In Section IV, we present the experiments we conducted to evaluate the proposed approach and discuss results. The main findings and conclusions are discussed in Section V.

II. Related Work

Previous work on transfer learning for sentiment analysis focuses on adapting document-level sentiment classifiers trained on labelled reviews from a source domain, e.g., movie reviews, to a target domain where a different vocabulary is used, e.g., kitchen appliances [5].

[1] http://www.twitter.com

A recursive neural tensor network for learning the sentiment of pieces of texts of different granularities, such as words, phrases, and sentences, was proposed in [6]. The network was trained on a sentiment annotated treebank[2] of parsed sentences for learning compositional vectors of words and phrases. This method is difficult to apply to Twitter data because of the lack of Twitter-specific sentiment treebanks and robust PCFG constituency parsers for Twitter [7].

There is a family of models that incorporate lexical knowledge provided by opinion lexicons for training document-level sentiment classifiers. In [4], words and documents are jointly represented by a bipartite graph of labelled and unlabelled nodes. The sentiment labels of words and documents are propagated to the unlabelled nodes using regularised least squares. In [8], the term-document matrix associated with a corpus of documents is factorised into three matrices specifying cluster labels for words and documents using a constrained non-negative tri-factorisation technique. Sentiment-annotated words and documents are introduced into the model as optimisation constraints. A generative naive Bayes model based on a polarity lexicon, which is then refined using sentiment-annotated documents, is proposed in [9].

Regarding the MPC task for tweets, state-of-the art solutions are based on supervised models such as logistic regression models and support vector machines trained from hand-annotated polarity corpora. Some of the features used for describing the tweets are: n-grams, POS tags, Brown clusters [10], and features derived from polarity lexicons [1], [11]. Satisfactory results have also been reported using convolutional neural networks and word embeddings [12], [13].

Distant supervision is a popular strategy for addressing the label sparsity problem of supervised models in MPC. In these methods, raw tweets gathered from the Twitter API[3] are automatically labelled into positive and negative classes using strong sentiment signals such as positive and negative emoticons, e.g., :), :([14]–[16], or emotional hashtags [11], e.g., #joy, #sadness. The signals are normally discarded from the content for feature extraction. However, these approaches are ill-suited to domains such as politics where emoticons or emotional hashtags are rarely used to express positive and negative opinions.

Another approach for tackling MPC in Twitter is proposed in [17]. This approach is based on distant supervision and lexical prior knowledge. The authors build a graph that has users, tweets, words, hashtags, and emoticons as its nodes. A subset of these nodes is labelled by prior sentiment knowledge provided by a polarity lexicon, the known polarity of emoticons, and a message-level classifier trained with emoticons. These sentiment labels are propagated throughout the graph using random walks.

A common approach for addressing the PLI task is to calculate a word-level sentiment score based on how frequently a word occurs in positive and negative messages. This measure, referred to as PMI semantic orientation, is calculated as the difference between the point-wise-mutual information (PMI) of a word occurring in positive and negative messages [18]. The message-level sentiment labels can be obtained through distant supervision [1], or using a self-training approach. In the latter case, a message-level classifier trained from a small corpus of hand-annotated tweets is used to classify a large collection of unlabelled messages from which the word-level sentiment scores are computed [19].

Another approach is to induce the lexicon by representing Twitter words from a corpus of tweets as vectors that are used together with a small group of labelled words for training a word-level polarity classifier. The resulting classifier is then deployed on the remaining unlabelled words for performing the induction. In [20], PMI-based semantic orientation was used together with other associations between words and emoticon-annotated tweets for building the classifier's feature space. Other types of word-level features, used in [2], [21], are low-dimensional dense word embeddings.

The results in these papers indicate that the sentiment-interdependence relation between words and messages can be helpful in the MPC and PLI tasks. Sentiment-annotated words can be used as prior knowledge for MPC, and the message-level sentiment distribution of words can be used for PLI. In this paper we propose a unified representation that allows the bidirectional transfer of sentiment classifiers between words and tweets. The main benefit of our approach is that it only requires labelled data in one of the two domains (words or messages) for transferring sentiment knowledge into the other one.

III. Tweet-Centroids for Transfer learning

In this section, we formalise the MPC and PLI problems and define the tweet centroid model (TCM) for sentiment transfer learning between words and messages. Following the notation proposed in [3], a domain $\mathcal{D}$ consists of two components: a feature space $\mathcal{X}$ and a probability distribution $P(X)$, where $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$ and each x_i is a numeric feature. Given a particular domain $\mathcal{D}$, a task $\mathcal{T}$ consists of a label space $\mathcal{Y}$ and a predictive function f that can be learned from training data consisting of pairs $\{x, y\}$ where $x \in X$ and $y \in \mathcal{Y}$. The function f can be used for predicting the corresponding label $f(x)$ of a new instance x.

In the Twitter sentiment analysis context, a tweet or message m is formed by a sequence of words. A tweet is represented by a *k*-dimensional vector $\overrightarrow{x}$ residing in a feature space $\mathcal{X}_\mathcal{M}$ that belongs to the message domain $\mathcal{D}_\mathcal{M}$. Different NLP features can be used to form $\mathcal{X}_\mathcal{M}$. In this paper, we consider three type of features that have proven to be useful for sentiment analysis of tweets [1]:

1) Word unigrams (UNI): a vector space model based on unigram frequency counts.
2) Brown clusters (BWN): a vector space model based on counting the frequency of word clusters trained with the Brown clustering algorithm [10]. This algorithm

[2]http://nlp.stanford.edu/sentiment/treebank.html

[3]https://dev.twitter.com/streaming/overview

produces hierarchical clusters of words by maximising the mutual information of bigrams.

3) Part-of-speech tags (POS): a vector space model based on counting the frequency of each POS tag in the message.

The message-level sentiment label space $\mathcal{Y}_{\mathcal{M}}$ corresponds to the different sentiment categories that can be expressed in a tweet, e.g., positive, negative, and neutral. For simplicity, we will only consider the two-class (positive and negative) case. Because sentiment is a subjective judgment, the ground-truth sentiment category of a tweet must be determined by a human evaluator.

Given a corpus of sentiment-annotated tweets $\mathcal{C}_L$, a message-level polarity classifier f_M can be trained using standard supervised learning methods and then be used for the MPC task. Annotated corpora are commonly not available for creating domain-specific sentiment classifiers due to the high costs involved in the annotation process. On the other hand, a large corpus of unlabelled public tweets $\mathcal{C}_U$ can be freely obtained from the Twitter API. Tweets restricted to a specific language, geographical region, or set of key words can also been collected for creating domain-specific collections.

Words can be annotated according to the same sentiment categories as messages ($\mathcal{Y}_{\mathcal{W}} = \mathcal{Y}_{\mathcal{M}}$) to indicate their prior sentiment. Examples of positive words are *happy* and *great*, and examples of negative ones are *sad* and *miserable*. Again, the ground-truth sentiment of a word is a subjective judgment determined by a human. We refer to a list of words annotated by sentiment as a polarity lexicon $\mathcal{L}$.

Distributional models [22] are used for representing lexical items such as words according to the context in which they occur. They are based on the hypothesis that words occurring in similar contexts tend to have similar meaning [23]. The tweet centroid model we apply in this paper is a distributional representation proposed in [24] that exploits the short nature of tweets by treating them as the whole contexts of words. This is done by representing words as the centroids of the tweets in which they occur within a corpus of tweets.

Let $\mathcal{V}$ be the vocabulary formed by the distinct words found in a corpus of unlabelled tweets $\mathcal{C}_U$, where the tweets from $\mathcal{C}_U$ are represented by the feature space $\mathcal{X}_{\mathcal{M}}$ described above. For each word w, we define the word-tweet set $\mathcal{M}(w)$ as the set of tweets in which w is observed:

$$\mathcal{M}(w) = \{m : w \in m\} \tag{1}$$

We define the tweet centroid word vector $\overrightarrow{w}$ as the centroid of all tweet vectors in which w is used. In other words, $\overrightarrow{w}$ is a k-dimensional vector in which each dimension w_j is calculated as follows:

$$w_j = \sum_{t \in \mathcal{M}(w)} \frac{x_j^{(t)}}{|\mathcal{M}(w)|} \tag{2}$$

Another interpretation of the tweet centroid model is that words are treated as the expected tweet in which they might occur. The word-level vectors can be used to form a word domain $\mathcal{D}_{\mathcal{W}}$ with the same feature space as the one used for representing the messages ($\overrightarrow{w} \in \mathcal{X}$) in the message domain $\mathcal{D}_{\mathcal{M}}$. Taking the words from the vocabulary that match a given polarity lexicon ($\mathcal{V} \cap \mathcal{L}$), a word-level polarity classifier f_W can be trained and used for classifying the remaining unlabelled words, thus solving the PLI task [24].

Transfer learning requires the source and the target tasks to be related to each other. We hypothesise that there is a strong relationship between MPC and PLI because the sentiment of a tweet is associated with the sentiment of the words it contains and the sentiment of a word is associated with the sentiment of the tweets that use it.

Assuming that this hypothesis is true, we can apply the tweet centroid model for addressing MPC and PLI by taking labels from the respective other domain. Considering that both tweets and words reside in the same feature space, given a collection of unlabelled tweets $\mathcal{C}_U$, we can classify the sentiment of messages using a word-level classifier f_W trained with tweet centroids labelled by a polarity lexicon $\mathcal{L}$.

It is important to note that the number of labelled words for training f_W is limited to the number of words from $\mathcal{L}$ occurring in $\mathcal{C}_U$. Most of exiting hand-annotated polarity lexicons are formed by less than $10,000$ words [25]. This means that our method is not capable of exploiting large collections of unlabelled tweets for producing training datasets larger than the size of $\mathcal{L}$. We propose a modification of our method for increasing the number labelled instances it produces. The modification is based on partitioning the word-tweet sets. The word-tweet set $\mathcal{M}(w)$ for each word from the lexicon ($w \in \mathcal{L}$) is partitioned into smaller disjoint subsets $\mathcal{M}(w)_1, \ldots \mathcal{M}(w)_z$ of a fixed size determined by a parameter p. We calculate one tweet centroid vector $\overrightarrow{w}$ for each partition labelled according to $\mathcal{L}$. As is shown in Section IV-B, this modification leads to substantial improvements when transferring sentiment knowledge from words to tweets.

The reverse transfer of sentiment knowledge is also possible. Given a message-level polarity classifier f_M trained from a corpus of sentiment annotated tweets $\mathcal{C}_L$, a polarity lexicon can be induced by applying f_M to the words from $\mathcal{C}_L$, simply by representing the words with the tweet centroid approach. Alternatively, considering that sentiment-annotated corpora are usually small and word-level distributional representations capture richer semantic information when calculated from large document corpora, it is also possible to perform the induction by applying f_M to word vectors calculated from a larger corpus of unlabelled tweets $\mathcal{C}_U$.

Our transfer learning approach is novel in the sense that both the source domain and target domain are represented with the same feature space ($\mathcal{X}_{\mathcal{M}} = \mathcal{X}_{\mathcal{W}}$). In most previous transfer learning models for text classification the features spaces of the two domains are different [3].

It is important to clarify that the message domain $\mathcal{D}_{\mathcal{M}}$ and the word domain $\mathcal{D}_{\mathcal{W}}$ do not have the same probability distribution. The probability distribution of the tweet domain, $P(X_m)$, is formed by sparse features such as unigrams and Brown clusters, whereas the distribution of the word domain, $P(x_w)$, is formed by averaging vectors from the tweet domain,

which yields dense vectors with lower variance. Moreover, the conditional distributions of the two sentiment classification tasks are not the same either. $P(Y_w|X_w)$ encodes the relation between the prior polarity of a word and its distributional representation, whereas $P(Y_m|X_m)$ represents the relation between the polarity of a message and its sparse feature vector. Hence, normally, $P(Y_w|X_w) \neq P(Y_m|X_m)$. The two domains are clearly different, and transfer learning is required.

IV. Experiments

In this section, we conduct an experimental evaluation of the proposed approach. The evaluation is divided into three parts. First, we empirically study the interdependence relation between tweets and words. Second, we evaluate how to transfer sentiment labels from words to tweets. Finally, we evaluate how to induce a polarity lexicon from sentiment-annotated tweets.

A. The word-tweet sentiment-interdependence relation

We start by studying the sentiment-interdependence relation between documents and words in Twitter: the sentiment of documents determines the sentiment of words, while the polarity of words determines the sentiment of tweets.

We describe positive and negative tweets based on the polarity of their words, and likewise, describe positive and negative words from a given polarity lexicon according to the polarity of the tweets in which they occur. We expect to observe clear differences between elements of different polarities based on these descriptions. The annotated data we use for this is taken from the *SemEval*[4] corpus of sentiment annotated tweets and the AFINN lexicon [26] of positive and negative words.

The *SemEval* [27] corpus is formed by 5232 positive tweets and 2067 negative tweets annotated by human evaluators using the crowdsourcing platform Amazon Mechanical Turk[5]. Each tweet is annotated by five Mechanical Turk workers and the final label is determined based on the majority of the labels.

The AFINN lexicon is formed by 1176 positive words and 2204 negative words, annotated by Finn Årup Nielsen[6], and includes informal words commonly found in Twitter such as slang, obscene words, acronyms and Web jargon. AFINN does not include any emoticons.

We describe each tweet from *SemEval* by a message-level polarity variable calculated as the difference between the number of positive and negative words from the AFINN lexicon found in the message. This variable is normalised by the total number of words in the tweet. The tweets that do not have words from the lexicon are discarded, resulting in 1638 negative and 4193 positive tweets. The median of this variable for negative and positive tweets is -0.04 and 0.05, respectively. The polarity of positive and negative categories is also compared using a Wilcoxon rank sum test obtaining a p-value less than $2.2e^{-16}$. This shows that there is statistical evidence that negative tweets are more likely to be formed by negative words than positive ones, and likewise positive tweets are more likely to contain positive words than negative ones. These results support the first part of the proposed tweet-word sentiment-interdependence relation: the sentiment of a tweet is determined by the polarity of its words.

We also describe each word from the AFINN lexicon by a word-level polarity variable calculated as the difference between the number of positive and negative tweets that contain it. This variable is normalised by the total number of tweets in which the word is used. To reduce the noise induced by infrequent words, we discard words occurring in fewer than three tweets, resulting in 259 positive and 250 negative words. The median of the word-level polarity for positive and negative classes is 0.76 and -0.33 respectively. We compare this variable for both sentiment classes using a Wilcoxon rank sum test and the resulting p-value is again less than $2.2e^{-16}$. This indicates that there is also statistical evidence that positive and negative words occur more frequently in tweets with the same polarity than in tweets with the opposite one. These results support the second part of the tweet-word sentiment-interdependence relation: the sentiment of a word is determined by the sentiment of tweets in which it occurs.

The distribution of the message-level and word-level polarity variables for each corresponding sentiment category is shown in the violin plots in Figure 1.

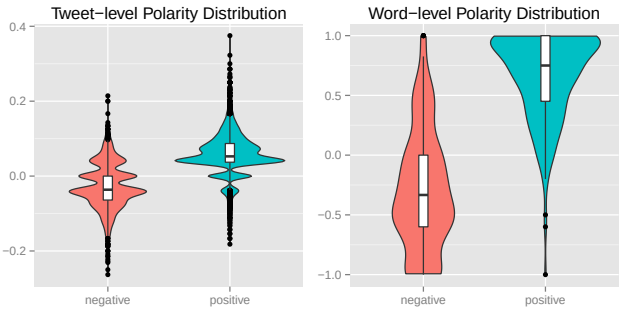

Fig. 1. Violin plots of the polarity of tweets and words.

From the plot we can observe that the interquartile range of the tweet-level polarity lies below zero for the negative class and above zero for the positive one, suggesting that tweets of different sentiment classes have different distributions when considering the sentiment of their words. Regarding the words we can again observe that the interquartile ranges lie below and above zero for negative and positive words respectively. Note that the gap between the positive and negative interquartile range is larger than the corresponding gap in the case of tweets. We believe that this is because there is more information available for describing words according to the polarity of the tweets in which they occur than for describing tweets according to the polarity of their words. In one case, the sentiment labels of the tweets in which opinion words occur are fully given by the sentiment-annotated corpus. In the other case, we only have the polarity of the words from a tweet that

[4]http://www.cs.york.ac.uk/semeval-2013/task2/
[5]http://www.mturk.com
[6]http://neuro.imm.dtu.dk/wiki/AFINN

match the lexicon but do not have sentiment information for the other words in the tweet.

B. From opinion words to sentiment tweets

In this subsection, we evaluate the transfer of sentiment labels from words to tweets for solving MPC. We train a word-level classifier f_W on tweet centroids calculated from a collection of unlabelled tweets $\mathcal{C}_U$, where these centroids are labelled according to a polarity lexicon $\mathcal{L}$. We also study the effect of partitioning the word-tweet sets to increase the number of training instances obtained with our tweet centroid method.

The collection of unlabelled tweets is taken from the Edinburgh corpus [28], which is a general purpose collection of 97 million unlabelled tweets in multiple languages collected with the Twitter streaming API between November 11th 2009 and February 1st 2010. Tweets written in languages different from English are discarded, resulting in a corpus of around 50 million English tweets. We use AFINN as the polarity lexicon for the centroid labels.

The features used for representing the tweets and the words from $\mathcal{C}_U$ are: unigrams, POS tags, and Brown clusters. The tweets are lowercased, and user mentions and URLs are replaced by special tokens. The tokenisation of the tweets, the calculation of the POS tags, and the Brown clusters are taken from the **TweetNLP** library[7].

We only consider word vectors of words that are included in the lexicon, and we also discard words occurring in fewer than ten tweets to avoid learning spurious relationships from infrequent words. The classifier is trained using an L_2-regularised logistic regression taken from LIBLINEAR[8], with the regularisation parameter C set to 1.0. We compare our model with classifiers trained using two distant supervision methods for obtaining training instances from unlabelled corpora: the emoticon-annotation approach (EAA) and the lexicon-annotation approach (LAA).

In EAA we use the following positive and negative emoticons for labelling tweets from the source collection: ":)", ":D", "=D", "=)", ":]", "=]", ":-)", ":-D", ":-]", ";)", ";D", ";]", ";-)", ";-D", and ";-]" for positive tweets and ":(", "=(", ";(", ":[", "=[", ":-(", ":-[", ":(", ":[", and "D:" for negative tweets. Tweets without emoticons and tweets containing both positive and negative emoticons are discarded. The emoticons are removed from the content after labelling.

In LAA the tweets from $\mathcal{C}_U$ are labelled using the AFINN lexicon. The tweets with at least one positive word and no negative word are labelled positive, and analogously, tweets with at least one negative word and no positive word are labelled negative.

It is important to recall that the training examples produced with the three methods reside in the same feature space. We study different configurations of TCM. The first configuration is the original version of TCM, in which we obtain one instance per word. The other configurations correspond to partitioned versions of TCM, in which the tweet-word sets of each word from the lexicon are partitioned into disjoints subsets of size p. The centroids are calculated from the partitions, and hence, multiple training instances are produced for words occurring in more than p tweets. The partitioning is implemented by enumerating the tweets in each word-tweet set and creating consecutive sublists of size p. The last partition of the set will be smaller than p if there is a remainder when dividing the size of the set by the value of p.

	Avg. Positive	(%)	Avg. Negative	(%)	Avg. Total	(%)
EAA	130,641	(6.5%)	21,537	(1.1%)	152,179	(7.6%)
LAA	681,531	(34.1%)	294,177	(14.7%)	975,708	(48.8%)
TCM	1537	(0.05%)	951	(0.08%)	2488	(0.12%)
TCM (p=5)	276,696	(13.8%)	149,989	(7.5%)	426,684	(21.3%)
TCM (p=10)	138,596	(6.9%)	75,390	(3.8%)	213,986	(10.7%)
TCM (p=20)	69,518	(3.5%)	38,044	(1.9%)	107,563	(5.4%)
TCM (p=50)	32,231	(1.6%)	17,950	(0.9%)	50,181	(2.5%)
TCM (p=100)	14,338	(0.7%)	8357	(0.4%)	22,695	(1.1%)

TABLE I
AVERAGE NUMBER OF POSITIVE AND NEGATIVE INSTANCES GENERATED BY DIFFERENT MODELS FROM 10 COLLECTIONS OF 2 MILLION TWEETS.

The evaluation of the classifiers is carried out on three manually annotated collections of tweets represented by the same features as the tweets from the corresponding partition: *SemEval*, *6HumanCoded*[9], and *Sanders*[10]. As was described in the previous subsection, the *SemEval* corpus is formed by 5232 positive and 2067 negative hand-annotated tweets. The *6HumanCoded* dataset is a collection of tweets scored according to positive and negative numeric scores by six human evaluators. The ratings are averaged and we use the difference of these scores to create polarity classes and discard messages where this difference is zero. The resulting dataset has 1340 positive and 949 negative tweets. The *Sanders* dataset consists of 570 positive and 654 negative tweets evaluated by a single human annotator.

We study the average performance obtained by classifiers trained on labelled instances generated by different configurations of TCM, EAA, and LAA, using ten independent subsamples of 2 million tweets from the Edinburgh corpus as the source data. The average number of positive and negative instances obtained by each model from the ten subsamples is shown in Table I.

We can see from the table that LAA produces the largest training dataset and that the original version of TCM produces the smallest one. Regarding the partitioned version of TCM, we observe that the lower the value of p, the larger the number of instances produced.

From the ten training sets, we compare the average area under the ROC curve (AUC) obtained on the three target collections of tweets for TCM and the two baselines EAA and LAA using a paired Wilcoxon signed-rank test with the significance value set to 0.05. AUC is a useful metric for comparing the performance of classifiers because it is independent of any specific value for the decision threshold. The comparisons are

[7]http://www.ark.cs.cmu.edu/TweetNLP/

[8]http://www.csie.ntu.edu.tw/~cjlin/liblinear/

[9]http://sentistrength.wlv.ac.uk/documentation/6humanCodedDataSets.zip

[10]http://www.sananalytics.com/lab/twitter-sentiment/

	6HumanCoded		Sanders		SemEval	
EAA	0.805 ± 0.005	= -	0.800 ± 0.017	= +	0.802 ± 0.006	= -
LAA	0.809 ± 0.001	+ =	0.778 ± 0.002	- =	0.814 ± 0.000	+ =
TCM	0.776 ± 0.004	- -	0.682 ± 0.024	- -	0.779 ± 0.008	- -
TCM (p=5)	0.834 ± 0.002	+ +	0.807 ± 0.008	= +	0.833 ± 0.002	+ +
TCM (p=10)	0.845 ± 0.003	+ +	**0.817** ± 0.006	+ +	0.841 ± 0.002	+ +
TCM (p=20)	**0.850** ± 0.003	+ +	0.815 ± 0.011	+ +	**0.844** ± 0.003	+ +
TCM (p=50)	0.844 ± 0.004	+ +	0.785 ± 0.010	- +	0.836 ± 0.004	+ +
TCM (p=100)	0.829 ± 0.003	+ +	0.752 ± 0.019	- -	0.821 ± 0.004	+ +

TABLE II
MESSAGE-LEVEL POLARITY CLASSIFICATION AUC VALUES. BEST RESULTS PER COLUMN ARE GIVEN IN BOLD.

done for the three target collections of tweets and the results are given in Table II. The statistical significance tests of each configuration of TCM with respect to EAA and LAA are indicated by a sequence of two symbols. Improvements are denoted by a plus (+), degradations by a minus (-), and cases where no statistically significant difference is observed by an equals (=). The baselines are also compared amongst each other.

Regarding the baselines, we observe that LAA is better than EAA in *6HumanCoded* and *SemEval* but worse in *Sanders*. The original version of TCM is statistically significantly worse than the two baselines. We believe that this is because non-partitioned TCM generates too few training instances (Table I). In contrast, the partitioned TCM achieves statistically significantly improvements over the two baselines in the three datasets when p equals 10 and 20. We also observe a degradation in performance when the value of p is decreased further (p=5). This suggests a trade-off in the value of p. If p is too large, TCM will generate too few training instances, and conversely, if p is too small, the instances will be calculated by averaging very few tweets, and the resulting distributional word vectors will lack contextual information.

Regarding the performance on the different datasets, we observe a lower performance for *Sanders* in comparison to the other two datasets. Considering that this is the only dataset in which labels are not obtained by averaging multiple human evaluations, we believe that this dataset contains noisier sentiment labels because it reflects the subjective judgement of a single evaluator.

The results obtained in this subsection indicate that opinion words can be successfully transferred to the message level using tweet centroids when the centroids are obtained from partitioned data. Additionally, we conclude that the partitioned tweet centroid method is capable of extracting better information from unlabelled tweets than EAA and LAA.

C. From tweets to opinion words

The research question evaluated in this subsection is whether it is possible to transfer the sentiment knowledge obtained from a sentiment-annotated corpus of tweets for solving PLI. To address this question, we train a message-level classifier f_W from a corpus of sentiment annotated tweets $\mathcal{C}_L$ and deploy it on words found in a corpus of unlabelled tweets, where the words are represented by tweet centroids. Considering that in this task we need to have a single instance per word, we do not partition the word-tweet sets here.

Instead of calculating the target tweet centroids from $\mathcal{C}_L$, we calculate them from a larger corpus of unlabelled tweets $\mathcal{C}_U$ that corresponds to one of the collections of 2 million tweets used in the previous subsection. This is done because of the following reasons:

1) There is empirical evidence that distributional semantic models of words tend to generalise better when calculated from large corpora [29].
2) By classifying the words from a larger corpus of unlabelled tweets we can induce the polarity of words that do not necessarily occur in the annotated corpus.

We use the three annotated collection of tweets that were previously used as testing data for training three message-level classifiers: *Sanders*, *6HumanCoded*, and *SemEval*. We build the feature space with the same features used before: unigrams, POS tags, and Brown clusters. We also use an L_2-regularised logistic regression model with the same parameters for learning the classifier. We only consider labelled words from the AFINN lexicon for evaluation purposes.

We compare the word-level classification AUC of a message-level classifier deployed on words represented by TCM with the AUC obtained by PMI semantic orientation (PMI-SO) [18], a popular method for inducing polarity lexicons from a corpus of polarity annotated tweets $\mathcal{C}_L$. PMI-SO corresponds to the difference between the PMI of a word with the positive class and the PMI of the same word with the negative one. Let *count* be a function that counts the number of times that a word w or a sentiment label y occurs in $\mathcal{C}_L$. The PMI-SO score for each word in $\mathcal{C}_L$ is calculated as follows:

$$\text{PMI-SO}(w) = log_2 \left(\frac{\text{count}(w \wedge y = pos) \times \text{count}(y = neg)}{\text{count}(y = pos) \times \text{count}(w \wedge y = neg)} \right)$$

The words classified by TCM and PMI-SO are not necessarily the same. TCM classifies the words from a larger corpus of unlabelled tweets $\mathcal{C}_U$ rather than classifying the words from $\mathcal{C}_L$. Therefore, the words induced by TCM are independent of the words from $\mathcal{C}_L$. On the other hand, PMI-SO classifies the words from the labelled corpus $\mathcal{C}_L$. In order to produce a fair comparison between TCM and PMI-SO, we compare the classification performance obtained for the words from AFINN that are classified by both methods. The number of positive and negative words classified by PMI-SO for each source corpus, the number of words classified by TCM for $\mathcal{C}_U$, and the number of words in the intersection, are all shown in Table III.

Set of Words	Pos	Neg	Total
PMI-SO (SemEval)	522	617	1139
PMI-SO (Sanders)	196	231	427
PMI-SO (6HumanCoded)	333	352	685
TCM	961	1554	2515
PMI-SO (SemEval) ∩ TCM	517	602	1119
PMI-SO (Sanders) ∩ TCM	194	227	421
PMI-SO (6HumanCoded) ∩ TCM	332	349	681

TABLE III
NUMBER OF POSITIVE AND NEGATIVE WORDS FROM AFINN.

The AUC values for the intersection of words classified by both PMI-SO and TCM are displayed in Table IV. From the table we can observe that TCM outperforms PMI-SO for solving PLI when trained on any of the three collections of sentiment annotated tweets. This is a noteworthy result, considering that PMI-SO is a widely-used approach for lexicon induction. We can also observe that classifiers trained from *6HumanCoded* and *SemEval* achieve satisfactory results on the AFINN words, and we observe a substantially lower performance for the classifier trained from *Sanders*.

AUC		
Source Dataset	PMI-SO	TCM
Sanders	0.757	**0.864**
6HumanCoded	0.861	**0.930**
SemEval	0.858	**0.916**

TABLE IV
WORD-LEVEL POLARITY CLASSIFICATION RESULTS FOR THE AFINN LEXICON. BEST RESULTS PER ROW ARE GIVEN IN BOLD.

These results suggest that the performance of the tweet centroid model for transferring sentiment knowledge from tweets to words can vary substantially depending on the quality of the corpus of sentiment-annotated tweets. We observe that corpora in which the labels are obtained by averaging the judgments of multiple annotators such as *6HumanCoded* and *SemEval* are preferable to corpora annotated by one single individual such as *Sanders*. The size of the corpus could also be a relevant factor, considering that *Sanders* is the smallest collection. It is worth mentioning that when an appropiate source corpus is used, the word-level performance obtained after transfer can be even better than for the reverse transfer learning task.

The probabilistic output of the logistic regression model applied to tweet centroids can be used to explore the sentiment intensities or semantic orientations of Twitter words. We calculate the log odds ratio of the positive and negative probabilities returned by the logistic regression model ($\log_2(\frac{P(pos)}{P(neg)})$) for all the words found in the corpus of unlabelled tweets (here we also include words that are not part of AFINN). In this way, we obtain a sentiment score for each word in which the polarity and the intensity of a word are determined by the sign and the absolute value of the score, respectively.

In Figure 2, we use word clouds to visualise the sentiment intensities of positive and negative words classified with the message-level classifier trained from the SemEval dataset.

The left-side word cloud corresponds to positive words in which the log odds are greater than zero ($\log_2(\frac{P(pos)}{P(neg)}) > 0$) and the size of each word is proportional to its score. Analogously, in the right-side word cloud we show negative words in wich the score is less than zero and the size of the words is proportional to the score multiplied by -1. We observe from the figure that the word-level sentiment intensities transferred from message-level sentiment knowledge are plausible.

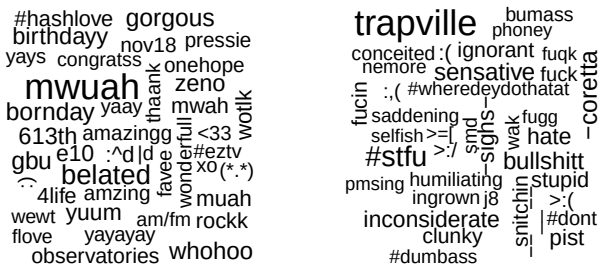

Fig. 2. Word clouds of positive and negative words obtained from a message-level classifier.

V. CONCLUSIONS

In this article, we have presented a transfer learning model for transferring sentiment knowledge between words and tweets by representing both tweets and words with the same features and deploying classifiers trained from one domain on data from the other one[11]. We studied the word-tweet sentiment interdependence relation on which the proposed tweet centroid model is based, showing that the sentiment of tweets is strongly related to the sentiment of their words and that the sentiment of a word is strongly related to the sentiment of the tweets in which it occurs.

We observed that the partitioned version of the tweet centroid model allows for accurate classification of the sentiment of tweets using a word-level classifier trained from a corpus of unlabelled tweets and a polarity lexicon of words. The partitioned tweet centroid model (with an appropiate partition size) outperformed the classification performance of the popular emoticon-based method for data labelling and also produced better results than a classifier trained from tweets labelled based on the polarity of their words (LAA). The partitioned tweet centroid model can be used for training message-level classifiers when no tweets annotated by sentiment are available and for domains in which emoticons are not frequently used. Considering that opinion lexicons are usually easier to obtain than corpora of sentiment-annotated tweets, the tweet centroid model can save significant labelling efforts when solving the message-level polarity classification problem.

Our results also show the feasibility of the reverse transfer process, where a polarity lexicon is induced by a message-level polarity classifier. We found that TCM produces more accurate lexicons than the well-known PMI-SO measure. The quality of the induced lexicon depends on the reliability of the sentiment-annotated Twitter data. An important aspect of TCM for lexicon induction is that the word centroids can be calculated from any collection of unlabelled tweets. Hence, the method can be used for creating domain-specific opinion lexicons by collecting tweets associated with the target domain.

A noteworthy aspect of the tweet centroid model is its flexibility: it can be used with any kinds of features for representing tweets. For example, paragraph vector-embeddings [30], which have shown to be powerful representations for sentences, could be trained from large corpora of unlabelled tweets and included in the message-level feature space.

[11]The source code of the model is available for download at http://www.cs.waikato.ac.nz/ml/sa/ds.html#ptcm.

The model is also sufficiently flexible to be used with any type of sentiment label for tweets or words. For future work, we will study the transferability of other sentiment information such as subjectivity or neutrality, numerical scores indicating sentiment strength, and multi-label emotions.

REFERENCES

[1] S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Journal of Artificial Intelligence Research*, pp. 723–762, 2014.

[2] S. Amir, W. Ling, R. Astudillo, B. Martins, M. J. Silva, and I. Trancoso, "Inesc-id: A regression model for large scale twitter sentiment lexicon induction," in *Proceedings of the 9th International Workshop on Semantic Evaluation*. Denver, Colorado: Association for Computational Linguistics, June 2015, pp. 613–618.

[3] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. on Knowl. and Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.

[4] V. Sindhwani and P. Melville, "Document-word co-regularization for semi-supervised sentiment analysis," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*. Washington, DC, USA: IEEE Computer Society, 2008, pp. 1025–1030.

[5] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proceedings of the 28th International Conference on Machine Learning*, 2011, pp. 513–520.

[6] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, vol. 1631, 2013, p. 1642.

[7] J. Foster, O. Cetinoglu, J. Wagner, J. Le Roux, J. Nivre, D. Hogan, and J. van Genabith, "From news to comment: Resources and benchmarks for parsing the language of web 2.0," in *Proceedings of 5th International Joint Conference on Natural Language Processing*. Chiang Mai, Thailand: Asian Federation of Natural Language Processing, November 2011, pp. 893–901.

[8] T. Li, Y. Zhang, and V. Sindhwani, "A non-negative matrix tri-factorization approach to sentiment classification with lexical prior knowledge," in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2009, pp. 244–252.

[9] P. Melville, W. Gryc, and R. D. Lawrence, "Sentiment analysis of blogs by combining lexical knowledge with text classification," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2009, pp. 1275–1284.

[10] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra, and J. C. Lai, "Class-based n-gram models of natural language," *Computational linguistics*, vol. 18, no. 4, pp. 467–479, 1992.

[11] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-canada: Building the state-of-the-art in sentiment analysis of tweets," in *Proceedings of the seventh international workshop on Semantic Evaluation Exercises*, 2013, pp. 321–327.

[12] D. Tang, F. Wei, B. Qin, T. Liu, and M. Zhou, "Coooolll: A deep learning system for twitter sentiment classification," in *Proceedings of the 8th International Workshop on Semantic Evaluation*. Dublin, Ireland: Association for Computational Linguistics and Dublin City University, August 2014, pp. 208–212.

[13] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: ACM, 2015, pp. 959–962.

[14] J. Read, "Using emoticons to reduce dependency in machine learning techniques for sentiment classification," in *Proceedings of the ACL Student Research Workshop*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 43–48.

[15] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Project Report, Stanford*, 2009.

[16] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," *In Proceedings of the Seventh Conference on International Language Resources and Evaluation*, 2010.

[17] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge, "Twitter polarity classification with label propagation over lexical links and the follower graph," in *Proceedings of the First Workshop on Unsupervised Learning in NLP*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 53–63.

[18] P. D. Turney, "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2002, pp. 417–424.

[19] L. Becker, G. Erhart, D. Skiba, and V. Matula, "Avaya: Sentiment analysis on twitter with self-training and polarity lexicon expansion," in *Proceedings of the seventh international workshop on Semantic Evaluation Exercises*, 2013, pp. 333–340.

[20] F. Bravo-Marquez, E. Frank, and B. Pfahringer, "Positive, negative, or neutral: Learning an expanded opinion lexicon from emoticon-annotated tweets," in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, 2015, pp. 1229–1235.

[21] D. Tang, F. Wei, B. Qin, M. Zhou, and T. Liu, "Building large-scale twitter-specific sentiment lexicon : A representation learning approach," in *Proceedings 25th International Conference on Computational Linguistics*, 2014, pp. 172–182.

[22] P. D. Turney and P. Pantel, "From frequency to meaning: Vector space models of semantics," *J. Artif. Int. Res.*, vol. 37, no. 1, pp. 141–188, 2010.

[23] Z. Harris, "Distributional structure," *Word*, vol. 10, no. 23, pp. 146–162, 1954.

[24] F. Bravo-Marquez, E. Frank, and B. Pfahringer, "From unlabelled tweets to twitter-specific opinion words," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. New York, NY, USA: ACM, 2015, pp. 743–746.

[25] F. Bravo-Marquez, M. Mendoza, and B. Poblete, "Meta-level sentiment models for big social data analysis," *Knowledge-Based Systems*, vol. 69, no. 0, pp. 86 – 99, 2014.

[26] F. Årup Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," in *Proceedings of the Workshop on 'Making Sense of Microposts': Big things come in small packages*, 2011, pp. 93–98.

[27] P. Nakov, S. Rosenthal, Z. Kozareva, V. Stoyanov, A. Ritter, and T. Wilson, "Semeval-2013 task 2: Sentiment analysis in twitter," in *Proceedings of the seventh international workshop on Semantic Evaluation Exercises*. Atlanta, Georgia, USA: Association for Computational Linguistics, June 2013, pp. 312–320.

[28] S. Petrović, M. Osborne, and V. Lavrenko, "The edinburgh twitter corpus," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 25–26.

[29] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111–3119.

[30] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31th International Conference on Machine Learning, Beijing, China, 21-26 June 2014*, 2014, pp. 1188–1196.