SAND2011-4779C

Estimating the Sentiment of Social Media Content for Security Informatics Applications

Kristin Glass
New Mexico Institute of Mining and Technology
Socorro, NM USA
kglass@icasa.nmt.edu

Richard Colbaugh
Sandia National Laboratories
New Mexico Institute of Mining and Technology
Albuquerque, NM USA
colbaugh@comcast.net

***Abstract*—Inferring the sentiment of social media content, for instance blog posts and forum threads, is both of great interest to security analysts and technically challenging to accomplish. This paper presents two computational methods for estimating social media sentiment which address the challenges associated with Web-based analysis. Each method formulates the task as one of text classification, models the data as a bipartite graph of documents and words, and assumes that only limited prior information is available regarding the sentiment orientation of any of the documents or words of interest. The first algorithm is a semi-supervised sentiment classifier which combines knowledge of the sentiment labels for a few documents and words with information present in unlabeled data, which is abundant online. The second algorithm assumes existence of a set of labeled documents in a domain related to the domain of interest, and leverages these data to estimate sentiment in the target domain. We demonstrate the utility of the proposed methods by showing they outperform several standard techniques for the task of inferring the sentiment of online movie and consumer product reviews. Additionally, we illustrate the potential of the methods for security informatics by estimating regional public opinion regarding Egypt's unfolding revolution through analysis of Arabic, Indonesian, and Danish (language) blog posts.**

***Keywords*—**-sentiment analysis, social media, security informatics, machine learning.

I. INTRODUCTION

There is increasing recognition that the Web represents a valuable source of security-relevant intelligence and that computational analysis offers a promising way of dealing with the problem of collecting and analyzing data at Web scale [e.g., 1-4]. As a consequence, tools and algorithms have been developed which support various security informatics objectives [3, 4]. To cite a specific example, we have recently shown that blog network dynamics can be exploited to provide reliable early warning for a class of extremist-related, real-world protest events [5].

Monitoring social media to spot emerging issues and trends and to assess public opinion concerning topics and events is of considerable interest to security professionals; however, performing such analysis is technically challenging. The opinions of individuals and groups are typically expressed as informal communications and are buried in the vast, and largely irrelevant, output of millions of bloggers and other online content producers. Consequently, effectively exploiting these data requires the development of new, automated methods of analysis [3,4]. Although helpful computational analytics have been derived for traditional forms of written content, less has been done to develop techniques that are well-suited to the particular characteristics of the content found in social media.

This paper considers one of the central problems in the new field of social media analytics: deciding whether a given document, such as a blog post or forum thread, expresses positive or negative opinion toward a particular topic. The informal nature of social media content poses a challenge for language-based sentiment analysis. While statistical learning-based methods often provide good performance in unstructured settings like this [e.g., 6-13], obtaining the required labeled instances of data, such as a collection of "exemplar" blog posts of known sentiment polarity, is usually an expensive and time-consuming undertaking.

We present two new computational methods for inferring sentiment orientation of social media content which address these challenges. Each method formulates the task as one of text classification, models the data as a bipartite graph of documents and words, and assumes that only limited prior information is available regarding the sentiment orientation of any of the documents or words of interest. The first algorithm adopts a semi-supervised approach to sentiment classification, combining knowledge of the sentiment polarity for a few documents and a small lexicon of words with information present in a corpus of unlabeled documents; note that such unlabeled data are readily obtainable in online applications. The second algorithm assumes existence of a set of labeled documents in a domain related to the domain of interest, and provides a procedure for transferring the sentiment knowledge contained in these data to the target domain. We demonstrate the utility of the proposed algorithms by showing they outperform several standard methods for the task of inferring the sentiment polarity of online reviews of movies and consumer products. Additionally, we illustrate the potential of the methods for security informatics through a case study in which sentiment analysis of Arabic, Indonesian, and Danish (language) blogs is used to estimate regional public opinion regarding the unfolding revolution in Egypt.

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

II. PRELIMINARIES

We approach the task of estimating the sentiment orientations of a collection of documents as a text classification problem. Each document of interest is represented as a "bag of words" feature vector $x \in \Re^{|V|}$, where the entries of x are the frequencies with which the words in the vocabulary set V appear in the document (perhaps normalized in some way [6]). We wish to learn a vector $c \in \Re^{|V|}$ such that the classifier orient = $\text{sign}(c^T x)$ accurately estimates the sentiment orientation of document x, returning +1 (−1) for documents expressing positive (negative) sentiment about the topic of interest.

Knowledge-based classifiers leverage prior domain information to construct the vector c. One way to obtain such a classifier is to assemble lexicons of positive words $V^+ \subseteq V$ and negative words $V^- \subseteq V$, and then to set $c_i = +1$ if word $i \in V^+$, $c_i = -1$ if $i \in V^-$, and $c_i = 0$ if i is not in either lexicon; this classifier simply sums the positive and negative sentiment words in the document and assigns document orientation accordingly. While this scheme can provide acceptable performance in certain settings, it is unable to improve its performance or adapt to new domains, and it is usually labor-intensive to construct lexicons which are sufficiently complete to enable useful sentiment classification performance to be achieved.

Alternatively, learning-based methods attempt to generate the classifier vector c from examples of positive and negative sentiment. To obtain a learning-based classifier, one can begin by assembling a set of n_l *labeled* documents $\{(x_i, d_i)\}$, where $d_i \in \{+1, -1\}$ is the sentiment label for document i. The vector c then can be learned through "training" with the set $\{(x_i, d_i)\}$, for instance by solving the following set of equations for c:

$$[X^T X + \gamma I_{|V|}]\, c = X^T d, \tag{1}$$

where matrix $X \in \Re^{n_l \times |V|}$ has document vectors for rows, $d \in \Re^{n_l}$ is the vector of document labels, $I_{|V|}$ denotes the $|V| \times |V|$ identity matrix, and $\gamma \geq 0$ is a constant; this corresponds to regularized least squares (RLS) learning [14]. Many other strategies can be used to compute c, including Naïve Bayes (NB) statistical inference [6]. Learning-based classifiers have the potential to improve their performance and to adapt to new situations, but standard methods for realizing these capabilities require that fairly large training sets of labeled documents be obtained and this is usually expensive.

Sentiment analysis of social media content for security informatics applications is often characterized by the existence of only modest levels of prior knowledge regarding the domain of interest, reflected in the availability of a few labeled documents and small lexicon of sentiment-laden words, and by the need to rapidly learn and adapt to new domains. As a consequence, standard knowledge-based and learning-based sentiment analysis methods are typically ill-suited for security informatics. In order to address this challenge, the sentiment analysis methods developed in this paper enable limited labeled data to be combined with readily available "auxiliary" information to produce accurate sentiment estimates. More specifically, the first proposed method is a *semi-supervised* algorithm [e.g., 9,10] which leverages a source of supplementary data which is abundant online: unlabeled documents and words. Our second algorithm is a novel *transfer learning* method [e.g., 11] which permits the knowledge present in data that has been previously labeled in a related domain (say online movie reviews) to be transferred to a new domain (e.g., reviews of consumer products).

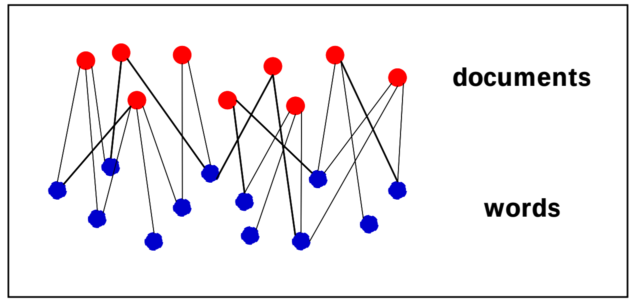

Figure 1. Cartoon of bipartite graph model G_b, in which documents (red vertices) are connected to the words (blue vertices) they contain and the link weights (black edges) can reflect word frequencies.

Each of the algorithms proposed in this paper assumes the availability of a modest lexicon of sentiment-laden words. This lexicon is encoded as a vector $w \in \Re^{|V_l|}$, where $V_l = V^+ \cup V^-$ is the sentiment lexicon and the entries of w are set to +1 or −1 according to the polarity of the corresponding words. The development of the algorithms begins by modeling the problem data as a bipartite graph G_b of documents and words (see Figure 1). It is easy to see that the adjacency matrix A for graph G_b is given by

$$A = \begin{bmatrix} 0 & X \\ X^T & 0 \end{bmatrix} \tag{2}$$

where the matrix $X \in \Re^{n \times |V|}$ is constructed by stacking the document vectors as rows, and each '0' is a matrix of zeros. In both the semi-supervised and transfer learning algorithms, integration of labeled and "auxiliary" data is accomplished by exploiting the relationships between documents and words encoded in the bipartite graph model. The basic idea is to assume that, in the bipartite graph G_b, positive documents will tend to be connected to (contain) positive words, and analogously for negative documents/words.

III. SEMI-SUPERVISED SENTIMENT ANALYSIS

We now derive our first sentiment estimation algorithm for social media content. Consider the common situation in which only limited prior knowledge is available about the way sentiment is expressed in the domain of interest, in the form of small sets of documents and words for which sentiment labels are known, but where abundant unlabeled documents can be easily collected (e.g., via Web crawling). In this setting it is natural to adopt a semi-supervised approach, in which labeled and unlabeled data are combined and leveraged in the analysis process. In what follows we present a novel bipartite graph-based approach to semi-supervised sentiment analysis.

Assume the initial problem data consists of a corpus of n documents, of which $n_l << n$ are labeled, and a modest lexicon V_l of sentiment-laden words, and suppose that this label information is encoded as vectors $d \in \Re^{nl}$ and $w \in \Re^{|Vl|}$, respectively. Let $d_{est} \in \Re^n$ be the vector of estimated sentiment orientations for the documents in the corpus, and define the "augmented" classifier $c_{aug} = [d_{est}^T \quad c^T]^T \in \Re^{n+|V|}$ which estimates the polarity of both documents and words. Note that the quantity c_{aug} is introduced for notational convenience in the subsequent development and is not directly employed for classification. More specifically, in the proposed methodology we learn c_{aug}, and therefore c, by solving an optimization problem involving the labeled and unlabeled training data, and then use c to estimate the sentiment of any new document of interest with the simple linear classifier orient = sign($c^T x$). We refer to this classifier as *semi-supervised* because it is learned using both labeled and unlabeled data. Assume for ease of notation that the documents and words are indexed so the first n_l elements of d_{est} and $|V_l|$ elements of c correspond to the labeled data.

We wish to learn an augmented classifier c_{aug} with the following three properties: 1.) if a document is labeled, then the corresponding entry of d_{est} should be close to this ±1 label; 2.) if a word is in the sentiment lexicon, then the corresponding entry of c should be close to this ±1 sentiment polarity; and 3.) if there is an edge X_{ij} of G_b that connects a document x and a word $v \in V$ and X_{ij} possesses significant weight, then the estimated polarities of x and v should be similar. These objectives are encoded in the following minimization problem:

$$\min_{c_{aug}} \; c_{aug}^T L c_{aug} + \beta_1 \sum_{i=1}^{n_l} (d_{est,i} - d_i)^2 + \beta_2 \sum_{i=1}^{|V_l|} (c_i - w_i)^2 \qquad (3)$$

where $L = D - A$ is the graph Laplacian matrix for G_b, with D the diagonal degree matrix for A (i.e., $D_{ii} = \Sigma_j A_{ij}$), and β_1, β_2 are nonnegative constants. Minimizing (3) enforces the three properties we seek for c_{aug}, with the second and third terms penalizing "errors" in the first two properties. To see that the first term enforces the third property, observe that this expression is a sum of components of the form $X_{ij}(d_{est,i} - c_j)^2$. The constants β_1, β_2 can be used to balance the relative importance of the three properties.

The c_{aug} which minimizes the objective function (3) can be obtained by solving the following set of linear equations:

$$\begin{bmatrix} L_{11} + \beta_1 I_{nl} & L_{12} & L_{13} & L_{14} \\ L_{21} & L_{22} & L_{23} & L_{24} \\ L_{31} & L_{32} & L_{33} + \beta_2 I_{|V_l|} & L_{34} \\ L_{41} & L_{42} & L_{43} & L_{44} \end{bmatrix} c_{aug} = \begin{bmatrix} \beta_1 d \\ 0 \\ \beta_2 w \\ 0 \end{bmatrix} \qquad (4)$$

where the L_{ij} are matrix blocks of L of appropriate dimension. The system (4) is sparse because the data matrix X is sparse, and therefore large-scale problems can be solved efficiently. Note that in situations where the set of available labeled documents and words is *very* limited, sentiment classifier performance can be improved by replacing L in (4) with the normalized Laplacian $L_n = D^{-1/2} L D^{-1/2}$, or with a power of this matrix L_n^k (for k a positive integer); this modification serves to "smooth" the polarity estimates assigned to the vertices of G_b. It is an interesting open problem to develop criteria for identifying those situations when it is beneficial to replace L with L_n^k.

We summarize this discussion by sketching an algorithm for learning the proposed semi-supervised classifier:

Algorithm SS:

1. Construct the set of equations (4), possibly by replacing the graph Laplacian L with L_n^k.
2. Solve equations (4) for $c_{aug} = [\, d_{est}^T \quad c^T \,]^T$ (for instance using the Conjugate Gradient method).
3. Estimate the sentiment orientation of any new document x of interest as: orient = sign($c^T x$).

The utility of Algorithm SS is now examined through a case study involving a standard sentiment analysis task: estimation of the sentiment polarity of online movie reviews.

IV. Case Study One: Movie Reviews

This case study examines the performance of Algorithm SS for the problem of estimating sentiment of online movie reviews. The data used in this study is a publicly available set of 2000 movie reviews, 1000 positive and 1000 negative, collected from the Internet Movie Database and archived at the website [15]. The Lemur Toolkit [16] was employed to construct the data matrix X and vector of document labels d from these reviews. A lexicon of ~1400 domain-independent sentiment-laden words was obtained from [17] and employed to build the lexicon vector w.

This study compares the movie review orientation classification accuracy of Algorithm SS with that of three other schemes: 1.) lexicon-only, in which the lexicon vector w is used as the classifier as summarized in Section II, 2.) a classical NB classifier obtained from [18], and 3.) a well-tuned version of the RLS classifier (1). Algorithm SS is implemented with the following parameter values: $\beta_1 = 0.1$, $\beta_2 = 0.5$, and k = 10. A focus of the investigation is evaluating the extent to which good sentiment estimation performance can be achieved even if only relatively few labeled documents are available for training; thus we examine training sets which incorporate a range of numbers of labeled documents: n_l = 50, 100, 150, 200, 300, 400, 600, 800, 1000.

Sample results from this study are depicted in Figure 2. Each data point in the plots represents the average of ten trials. In each trial, the movie reviews are randomly partitioned into 1000 training and 1000 test documents, and a randomly selected subset of training documents of size n_l is "labeled" (i.e., the labels for these reviews are made available to the learning algorithms). As shown in Figure 2, Algorithm SS outperforms the other three methods. Note that, in particular, the accuracy obtained with the proposed approach is significantly better than the other techniques when the number of labeled training documents is small. It is expected that this property of Algorithm SS will be of considerable value in security informatics applications that involve social media data.

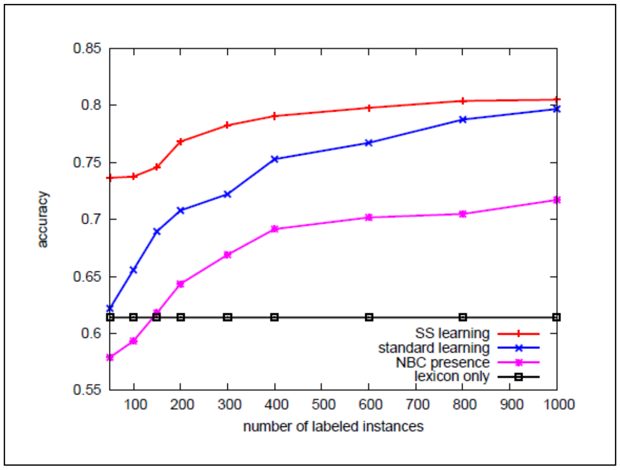

Figure 2. Results for the movie reviews case study. The plot shows how sentiment estimation accuracy (vertical axis) varies with number of available labeled movie reviews (horizontal axis) for four different classifiers: lexicon only (black), NB (magenta), RLS (blue), and Algorithm SS (red).

V. Transfer Learning Sentiment Analysis

This section develops the second proposed sentiment estimation algorithm for social media content. Many security informatics applications are characterized by the presence of limited labeled data for the domain of interest but ample labeled information for a related domain. For instance, an analyst may wish to ascertain the sentiment of online discussions about an emerging topic of interest, and may have in hand a large set of labeled examples of positive and negative posts regarding other topics (e.g., from previous studies). In this setting it is natural to adopt a transfer learning approach, in which knowledge concerning the way sentiment is expressed in one domain, the so-called *source* domain, is transferred to permit sentiment estimation in a new *target* domain. We now present a new bipartite graph-based approach to transfer learning sentiment analysis.

Assume that the initial problem data consists of a corpus of $n = n_T + n_S$ documents, where n_T is the (small) number of labeled documents available for the target domain of interest and $n_S >> n_T$ is the number of labeled documents from some related source domain; in addition, suppose that a modest lexicon V_l of sentiment-laden words is known. Let this label data be encoded as vectors $d_T \in \Re^{nT}$, $d_S \in \Re^{nS}$, and $w \in \Re^{|V|}$, respectively. Denote by $d_{T,est} \in \Re^{nT}$, $d_{S,est} \in \Re^{nS}$, and $c \in \Re^{|Vl|}$ the vectors of estimated sentiment orientations for the target and source documents and the words, and define the augmented classifier as $c_{aug} = [d_{S,est}{}^T \; d_{T,est}{}^T \; c^T]^T \in \Re^{n+|V|}$. Note that the quantity c_{aug} is introduced for notational convenience in the subsequent development and is not directly employed for classification.

In what follows we derive an algorithm for learning c_{aug}, and therefore c, by solving an optimization problem involving the labeled source and target training data, and then use c to estimate the sentiment of any new document of interest via the simple linear classifier orient = sign($c^T x$). This classifier is referred to as *transfer learning-based* because c is learned, in part, by transferring knowledge about the way sentiment is expressed from a domain which is related to (but need not be identical to) the domain of interest.

We wish to learn an augmented classifier c_{aug} with the following four properties: 1.) if a source document is labeled, then the corresponding entry of $d_{S,est}$ should be close to this ±1 label; 2.) if a target document is labeled, then the corresponding entry of $d_{T,est}$ should be close to this ±1 label, and the information encoded in d_T should be emphasized relative to that in the source labels d_S; 3.) if a word is in the sentiment lexicon, then the corresponding entry of c should be close to this ±1 sentiment polarity; and 4.) if there is an edge X_{ij} of G_b that connects a document x and a word $v \in V$ and X_{ij} possesses significant weight, then the estimated polarities of x and v should be similar.

The four objectives listed above may be realized by solving the following minimization problem:

$$\min_{c_{aug}} \; c_{aug}^T L c_{aug} + \beta_1 \| d_{S,est} - k_S d_S \|^2 + \beta_2 \| d_{T,est} - k_T d_T \|^2 + \beta_3 \| c - w \|^2 \quad (5)$$

where L = D − A is the graph Laplacian matrix for G_b, as before, and β_1, β_2, β_3, k_S, and k_T are nonnegative constants. Minimizing (5) enforces the four properties we seek for c_{aug}. More specifically, the second, third, and fourth terms penalize “errors” in the first three properties, and choosing $\beta_2 > \beta_1$ and $k_T > k_S$ favors target label data over source labels. To see that the first term enforces the fourth property, note that this expression is a sum of components of the form $X_{ij} (d_{T,est,i} - c_j)^2$ and $X_{ij} (d_{S,est,i} - c_j)^2$.The constants β_1, β_2, β_3 can be used to balance the relative importance of the four properties.

The c_{aug} which minimizes the objective function (5) can be obtained by solving the following set of linear equations:

$$\begin{bmatrix} L_{11} + \beta_1 I_{nS} & L_{12} & L_{13} \\ L_{21} & L_{22} + \beta_2 I_{nT} & L_{23} \\ L_{31} & L_{32} & L_{33} + \beta_3 I_{|V_l|} \end{bmatrix} c_{aug} = \begin{bmatrix} \beta_1 k_S d_S \\ \beta_2 k_T d_T \\ \beta_3 w \end{bmatrix} \quad (6)$$

where the L_{ij} are matrix blocks of L of appropriate dimension. The system (6) is sparse because the data matrix X is sparse, and therefore large-scale problems can be solved efficiently. In applications with very limited labeled data, sentiment classifier performance can be improved by replacing L in (6) with the normalized Laplacian L_n or with a power of this matrix $L_n{}^k$.

We summarize the above discussion by sketching an algorithm for constructing the proposed transfer learning classifier:

Algorithm TL:

1. Construct the set of equations (6), possibly by replacing the graph Laplacian L with $L_n{}^k$.
2. Solve equations (6) for $c_{aug} = [d_{S,est}{}^T \; d_{T,est}{}^T \; c^T]^T$.
3. Estimate the sentiment orientation of any new document x of interest as: orient = sign($c^T x$).

It is an interesting open problem to develop methods for characterizing *a priori* whether two domains are similar enough to serve as an effective source/target pair for transfer learning.

VI. Case Study Two: Product Reviews

This case study examines the performance of Algorithm TL for the problem of estimating sentiment of online product reviews. The data used in this study is a publicly available set of 1000 reviews of electronics products, 500 positive and 500 negative, and 1000 reviews of kitchen appliances, 500 positive and 500 negative, collected from Amazon and archived at the website [19]. The Lemur Toolkit [16] was employed to construct the data matrix X and vectors of document labels d_S and d_T from these reviews. A lexicon of 150 domain-independent sentiment-laden words was constructed manually and employed to form the lexicon vector w.

This study compares the product review sentiment classification accuracy of Algorithm TL with that of four other strategies: 1.) lexicon-only, in which the lexicon vector w is used as the classifier as summarized in Section II, 2.) a classical NB classifier obtained from [18], 3.) a well-tuned version of the RLS classifier (1), and 4.) Algorithm SS. Algorithm TL is implemented with the following parameter values: $\beta_1 = 1.0$, $\beta_2 = 3.0$, $\beta_3 = 5.0$, $k_S = 0.5$, $k_T = 1.0$, and $k = 5$. A focus of the investigation is evaluating the extent to which the knowledge present in labeled reviews from a related domain, here kitchen appliances, can be transferred to a new domain for which only limited labeled data is available, in this case electronics. Thus we assume that all 1000 labeled kitchen reviews are available to Algorithm TL (the only algorithm which is designed to exploit this information), and examine training sets which incorporate a range of numbers of labeled documents from the electronics domain: $n_T = 20, 50, 100, 200, 300, 400$.

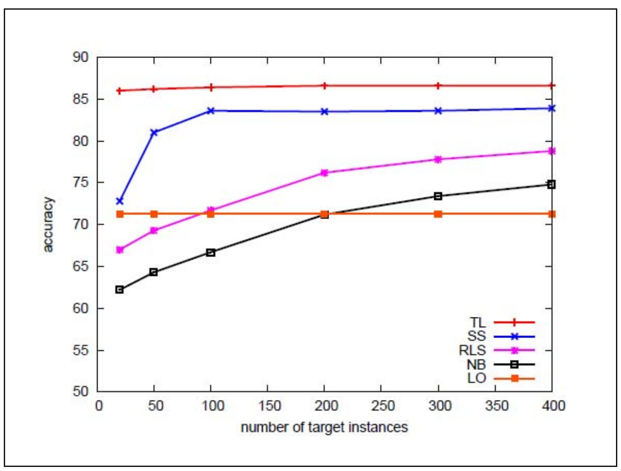

Figure 3. Results for the consumer product reviews case study. The plot shows how sentiment estimation accuracy (vertical axis) varies with number of available labeled electronics reviews (horizontal axis) for five different classifiers: lexicon only (orange), NB (black), RLS (magenta), Algorithm SS (blue), and Algorithm TL (red).

Sample results from this study are depicted in Figure 3. Each data point in the plots represents the average of ten trials. In each trial, the electronics reviews are randomly partitioned into 500 training and 500 test documents, and a randomly selected subset of reviews of size n_T is extracted from the 500 labeled training instances and made available to the learning algorithms. As shown in Figure 3, Algorithm TL outperforms the other four methods. Note that, in particular, the accuracy obtained with the transfer learning approach is significantly better than the other techniques when the number of labeled training documents in the target domain is small. It is expected that the ability of Algorithm TL to exploit knowledge from a related domain to quickly learn an effective sentiment classifier for a new domain will be of considerable value in security informatics applications involving social media data.

VII. Case Study Three: Egyptian Revolution

Beginning on 25 January 2011, a popular uprising swept across Egypt in the form of massive demonstrations and rallies, labor strikes in various sectors, and violent clashes between protestors and security forces, ultimately leading to the resignation of Egyptian President Hosni Mubarak on 11 February. Interest has been expressed in understanding public sentiment regarding the Egyptian revolution generally and Mubarak specifically, especially 1.) in the weeks before the protests and 2.) for different regions of the globe.

To enable a preliminary assessment along these lines, we collected three sets of blog posts which are related to Egyptian unrest and Mubarak and were posted during the two week period immediately before the protests began on 25 January: 1.) 100 Arabic posts, 2.) 100 Indonesian posts, and 3.) 100 Danish posts. We manually labeled the sentiment of a small subset of these documents, and translated into the appropriate language the generic sentiment lexicon used in Case Study One for implementation in this study. Observe that this approach to constructing a sentiment lexicon is far from perfect. However, because our proposed algorithms employ several sources of information to estimate the sentiment of content, it is expected that they will exhibit robustness to imperfections in any single data source. This study therefore offers the opportunity to explore the utility of a very simple approach to multilingual sentiment analysis: translate a small lexicon of sentiment-laden words into the language of interest and then apply Algorithm SS or Algorithm TL directly within that language (treating words as tokens). The capability to perform automated, multilingual content analysis is of substantial interest in many security-related applications.

We used Algorithm SS to estimate the sentiment expressed in the three sets of blog posts noted above, classifying the posts as either 'negative' or 'positive/neutral'. The analysis reveals that, while the sentiment expressed by the bloggers in the sample is largely negative toward Mubarak, the fraction of negative posts varies by post language (and thus possibly by geographic region). In particular, as shown in Figure 4, Arabic language posts are the most negative, followed by Indonesian posts, with the Danish posts in our sample actually being slightly more positive/neutral than negative. Manual inspection of a subset of the comments confirms the results provided by Algorithm SS.

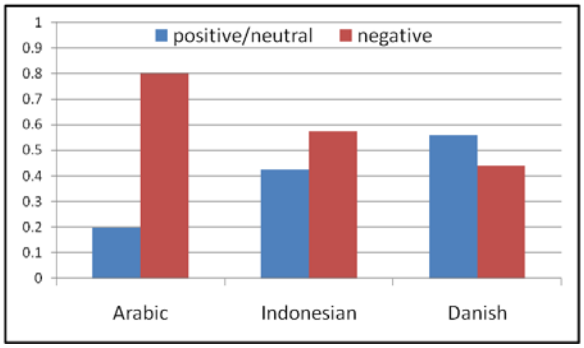

Figure 4. Results for Egyptian revolution case study. The bar chart shows that, while blog sentiment toward former Egyptian President Hosni Mubarak in the weeks leading up to the protests was largely negative, this sentiment varies with the language of posts. More specifically, the fraction of posts expressing negative sentiment regarding Mubarak is 0.80 for Arabic posts, 0.58 for Indonesian posts, and 0.44 for Danish posts.

ACKNOWLEDGEMENTS

This work was supported by the U.S. Department of Defense and the Laboratory Directed Research and Development Program at Sandia National Laboratories.

REFERENCES

[1] US Committee on Homeland Security and Government Affairs, Violent Extremism, the Internet, and the Homegrown Terrorism Threat, 2008.

[2] Bergin, A., S. Osman, C. Ungerer, and N. Yasin, "Countering Internet Radicalization in Southeast Asia", ASPI Special Report, March 2009.

[3] Chen, H., C. Yang, M. Chau, and S. Li (Editors), *Intelligence and Security Informatics*, Lecture Notes in Computer Science, Springer, Berlin, 2009.

[4] *Proc. 2010 IEEE International Conference on Intelligence and Security Informatics*, Vancouver, BC Canada, May 2010.

[5] Colbaugh, R. and K. Glass, "Early warning analysis for social diffusion events", *Proc. 2010 IEEE International Conference on Intelligence and Security Informatics*, Vancouver, BC Canada, May 2010.

[6] Pang, B. and L. Lee, "Opinion mining and sentiment analysis", *Foundations and Trends in Information Retrieval*, Vol. 2 , pp. 1-135, 2008.

[7] Dhillon, I., "Co-clustering documents and words using bipartite spectral graph partitioning", *Proc. ACM International Conference on Knowledge Discovery and Data Mining*, San Francisco, August 2001.

[8] Kim, S. and E. Hovy, "Determining the sentiment of opinions", *Proc. International Conference on Computational Linguistics*, 2004.

[9] Sindhwani, V. and P. Melville, "Document-word co-regularization for semi-supervised sentiment analysis", *Proc. 2008 IEEE International Conference on Data Mining*, Pisa, Italy, December 2008.

[10] Colbaugh, R. and K. Glass, "Estimating sentiment orientation in social media for intelligence monitoring and analysis", *Proc. 2010 IEEE International Conference on Intelligence and Security Informatics*, Vancouver, BC Canada, May 2010.

[11] Pan, S. and Q. Yang, "A survey on transfer learning", *IEEE Trans. Knowledge and Data Engineering*, Vol. 22, pp. 1345-1359, 2010.

[12] Blitzer, J., M. Dredze, and F. Perieia, "Biographies, bollywood, boom-boxes, and blenders: Domain adaptation for sentiment classification", *Proc. 45th Annual Meeting of the ACL*, Prague, June 2007.

[13] He, J., Y. Liu, and R. Lawrence, "Graph-based transfer learning", *Proc. 18th ACM Conference on Information and Knowledge Management*, Hong Kong, November 2009.

[14] Hastie, T., R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Second Edition, Springer, New York, 2009.

[15] http://www.cs.cornell.edu/People/pabo/movie-review-data/, accessed Dec. 2009.

[16] http://www.lemurproject.org/, accessed Dec. 2009.

[17] Ramakrishnan, G., A. Jadhav, A. Joshi, S. Chakrabarti, and P. Bhattacharyya, "Question answering via Bayesian inference on lexical relations", *Proc. 41st Annual Meeting of the ACL*, 2003.

[18] http://www.borgelt.net/bayes.html, accessed Dec. 2009.

[19] http://www.cs.jhu.edu/~mdredze/, accessed Dec. 2010.