Sentiment Analyzer for Arabic Comments System

Alaa El-Dine Ali Hamouda
Faculty of Engineering
Al-Azhar University

Fatma El-zahraa El-taher
Faculty of Engineering,
Al-Azhar University

***Abstract*—Today, the number of users of social network is increasing. Millions of users share opinions on different aspects of life every day. Therefore social network are rich sources of data for opinion mining and sentiment analysis. Also users have become more interested in following news pages on Facebook. Several posts; political for example, have thousands of users' comments that agree/disagree with the post content. Such comments can be a good indicator for the community opinion about the post content. For politicians, marketers, decision makers …, it is required to make sentiment analysis to know the percentage of users agree, disagree and neutral respect to a post. This raised the need to analyze theusers' comments in Facebook. We focused on Arabic Facebook news pages for the task of sentiment analysis. We developed a corpus for sentiment analysis and opinion mining purposes. Then, we used different machine learning algorithms – decision tree, support vector machines, and naive bayes - to develop sentiment analyzer. The performance of the system using each technique was evaluated and compared with others.**

Keywords*—Analysis for Arabic comments; machine learning algorithms; sentiment analysis; opinion mining

I. INTRODUCTION

Recently the rate of users comments and reviews increased dramatically as a medium of expressing ideas across the WWW specially in Facebook (Active users of Facebook increased from just a million in 2004 to over 750 million in 2011[1]). The fast growth of such content has not been fully harnessed yet. Information left by the users is not analysis yet.

Users are interested in knowing the percentage of users agree, disagree and neutral respect to a post in news pages on Facebook. For example, a lot of posts in the Arabic news pages like رصد، سلفيو كوستا، 6أبريل كلنا خالد سعيد، الصفحه الرسميه لرئاسة مجلس الوزراء، (Rassd, Silvio Costa, 6 April, We are all Khaled Said, official page for the presidency of the Council of Ministers) get thousands of comments on each post. The posts can express politician declarations, government decision, products announcement… The politician analysts, marketers, and decision makers, newspapers and news channels need to measure the community opinions about a certain topic expressed by a post. This is the motive for us to design and develop an analyzer system for Arabic comments.

An important part of our information-gathering behavior has always been to find out what other people think. With the growing availability and popularity of opinion-rich resources such as online review sites and personal blogs, new opportunities and challenges arise as people now can, and do, actively use information technologies to seek out and understand the opinions of others. The sudden eruption of activity in the area of opinion mining and sentiment analysis, which deals with the computational treatment of opinion, sentiment, and subjectivity in text, has thus occurred at least in part as a direct response to the surge of interest in new systems that deal directly with opinions as a first-class object [2].

Existing supervised learning methods can be readily applied to sentiment classification, e.g., naïve Bayesian, and support vector machines (SVM), etc. Pang et al. [3] took this approach to classify movie reviews into two classes, positive and negative. It was shown that using unigrams (a bag of individual words) as features in classification performed well with either naïve Bayesian or SVM.

Some researchers have been performed for creating automatic analysis of Twitter posts during recent years. Some of these researchers investigate the utility of linguistic features for detecting the sentiment of Twitter messages [4].Other researchers use text messages from Twitter, a popular microblogging platform, for building a dataset of emotional texts. Using the built dataset, the system classifies the meaning of adjectives into positive or negative sentiment polarity according to the given context. The approach is fully automatic. It does not require any additional hand-built language resources and it is language independent [5].

In [6], the authors use web-blogs to construct a corpus for sentiment analysis and use emotion icons assigned to blog posts as indicators of users' mood. The authors applied SVM and CRF learners to classify sentiments at the sentence level and then investigated several strategies to determine the overall sentiment of the document. Emoticon; a textual representation of an author's emotion was used in Internet blogs and textual chats. The winning strategy was defined by considering the sentiment of the last sentence of the document as the sentiment at the document level [6].

Although there is some work for twitter analysis, there is no real work –to the best of our knowledge- to investigate and develop such analyzer for Facebook comments.

Our proposed system uses classification methods to analyze users' comments and detect the comments that agree, disagree or is neutral with respect to a post .The system structure is presented in section 2 including features, classifiers and corpus details. Then, implementation and system evaluation are discussed in sections 3 and 4 respectively. Finally, conclusion comes in section 5.

II. PROPOSED SENTIMENT ANALYZER SYSTEM

Proposed system uses classification techniques to get better results using specified set of features. To train the classifiers, labeled (annotated) training and testing corpus are prepared .The system components include preprocessing,

features selection, and classification methods to make the sentiment analysis for comments. The system components are described in details in the following sections.

A. Preprocessing

Preprocessing phase includes stop words removal and very long comments removal. Stop words are common words that carry less important meaning than keywords. Removing stop words from the comments return the most important words. By sampling 6000 comments in Egyptians pages in most important pages about 80% of very long comments in most cases are advertise for pages on Facebook.

B. The Proposed Features

The input comments are segmented into words, spaces, commas, parenthesis and new line for identifying words .Our approach is to use different machine learning classifiers and feature extractors. We use two groups of features and evaluate them. These features are explained in the following sections.

1) Common Words between Post and Comments Features

The first group of features includes the number of words in post only, comment only, both post and comment. They are normalized by them by the length of comment and post. The idea behind these features is that the intersection between important words (not stop words) in post and comment may express if the comment agrees or disagrees with the post or not. The used equations for each feature are as follows.

Feature 1: Number of Words in Post Only Number of words in post only feature is (after stop words removal) computed using equation 1.

$$\frac{\text{Num Of words in Post Only}}{(\text{length of comment} + \text{length of post})} \quad (1)$$

Feature 2: Number of Words in Comment Only Number of words in comment only feature is (After stop word removal) computed using equation 2.

$$\frac{2 * \text{Num Of words inComment only}}{(\text{length of comment} + \text{length of post})} \quad (2)$$

The numerator is multiply by 2 for normalization .If the comment and post are similar will give one

Feature 3: Number of Words Common between Post and Comment:

Number of words common between post and comment feature is (After stop word removal) computed using equation 3.

$$\frac{\text{Number of Words common between post and comment}}{(\text{length of comment} + \text{length of post})} \quad (3)$$

2) *All Words in Posts and Comments Features*

The second group of features is the union of all words in the posts and comments. Each word (feature) takes one of the four values:

- "C "if the word is not in the post or the comment
- "M" if the word is in the post only
- "N" if the word is in the comment only
- "H" if the word is in both of the post and the comment

By that, we have number of features equal to the union of the words in both posts and comments.

Example:

Pots: الشرطة العسكرية تمارس «الضبطية القضائية»

"Military Police applied 'judicial officers'"

Comment1: واضح اننا هنشوف ايام سوداء من الشرطة العسكرية

"It is clear that we will see bad days"

Comment 2: بجد لا تعليق

"No comment"

	الشرطه	العسكريه	تمارس	الضبطيه	القضائيه	واضح	هنشوف	ايام	سوداء	بجد	تعليق
Comment 1	H	H	M	M	M	N	N	N	N	C	C
Comment 2	H	H	H	H	H	C	C	C	C	N	N

3) *Negation and Relevance Features*

The negation and similarity features are added to the features in group one and group two to improve the results. The negation words like (لن ، لا ، لم ، ما ، ليس) (no, not) .these features are:

Feature 1: Number of Negation Words in Post: Number of Negation words in post feature are (after stop words removal) computed using equation 4.

$$\frac{\text{Number of negative words in post}}{\text{length of post}} \quad (4)$$

Feature 2: Number of Negation Words in Comment: Number of negative words in comment feature are (after stop words removal) computed using equation 5.

$$\frac{\text{Number of negative words in comment}}{\text{length of comment}} \quad (5)$$

Feature 3: Relevance with Post

Relevance with post feature measures the relation between comments and post. The term frequency; Tf for each word in the post and comments (after stop words removal) are computed using equation 6.

$$Tf = F \quad (6)$$

F is the term frequency for a word in the comment. The vectors for the post and comments are formed.

Then, the relevance is computed using Cosine Similarity [7] method.

C. Classifiers

Using machine learning, several classifiers are developed to predict the type of each comment; agree, disagree or neutral to the post. Different machine learning techniques are used as follows.

1) Naive Bayes

Naive Bayes is a simple model which works well on text categorization [8]. We use a multinomial Naive Bayes model. Class c* is assigned to comment d, where

$$C^* = argmac_c P_{NB}(c|d)$$

$$P_{NB}(c|d) = \frac{P(c)\sum_{i=1}^{m} P(f|c)^{n_{i(d)}}}{P(d)} \qquad (7)$$

In this formula, f represents a feature and ni (d) represents the count of feature fi found in comment d. There are a total of m features. Parameters P(c) and P (f|c) are obtained through maximum likelihood estimates, and add-1 smoothing is utilized for unseen features

2) Decision Tree

The task of inducing a decision tree is typically handled by a recursive partitioning algorithm which, at each non-terminal node in the tree, branches on that attribute which discriminates best between the cases filtered down to that node [9].

3) Support Vector Machines

Standard support vector machines (SVMs), which are powerful tools for data classification, classify 2-category points by assigning them to one of two disjoint half spaces in either the original input space of the problem for linear classifiers, or in a higher dimensional feature space for nonlinear classifiers [10].

III. Implementation

To apply the proposed system, we developed the following components as follows.

A. Preprocessing

In this phase, the data is prepared before feeding to the classifiers .We used stop words listed in [11] and added additional stop words for the Colloquial Arabic. Stop words for the Colloquial Arabic are like (..... ,اللى ,ده دى) (Da, de, Elly, ...)

The similarity feature is used to detect the redundant comments. If two comments have similarity value equal to or more than a threshold (0.4), the shortest one will be removed. For long comments, they are not included in the summary. We ignore comments with number of words -after stop words removal- more than 150 words. The preprocessing also includes removing special characters like #, @, !, % and others. In addition, the redundant letters like منقووووول (Menkowoowol) are removed to get single written way for the same word.

B. The Proposed Features

The value of each feature is normalized to be between zero and one. The features explained in section 2 are used.

C. Data Corpus

To train the classifiers, a corpus is collected from the news pages in the Facebook. Recent "Egypt" and "Arabic Region" news were selected. We used the news pages of

رصد، سلفيو كوستا، 6أبريل ، كلنا خالد سعيد، الصفحه الرسميه لرئاسة مجلس الوزراء، شبكه اخبار مصر، اخر اخبار ميدان التحرير، المصرى اليوم.

(Rassd, Silvio Costa, 6 April, We are all Khaled Said, official page for the presidency of the Council of Ministers, Egypt News Network, Tahrir Square News, Egyptian today).

The total corpus size is 2400 comments collected from 220 posts; 800 neutral comments, 800 supportive comments, and 800 attacking comments. Each comment is represented into a single record, and then grouped manually into 3 groups; group one with the value "y" corresponding to supportive comments, group two with the value "n" corresponding to attacking comments and group three with the value "u" corresponding to neutral comments.

D. The Classifiers

The training data is used to learn all classifiers including Naive Bayes, decision tree and support vector machines.

IV. System Evaluations and Results

Classification approach is to classify comments to neutral comments, supportive comments, and attacking comments. After training the classifier, it will able to classify new comment to one of the three classes. Comparing these labeled of comments with those labeled that given manually by a human expert, we calculate the precision and recall.

For system evaluation, we tried different groups of features with different classifiers; support vector machines, naive bayes, and decision trees to find the features that give the best performance. The classifiers classify the comments to three categories; supportive comments 'y', attacking comments 'n', and neutral comments 'u'. Adding negation words and similarity features for all words in posts and comments features give the best performance. Naive Bayes gives 59.9%. With the decision tree, the precision and recall improved with 10%. Finally, SVM gives the best results 73.4% for precision and recall.

TABLE I. Using Common Words between Post and Comments Features

	SVM		Naive Bayes		Decision Trees	
	Precision	Recall	Precision	Recall	Precision	Recall
Attacking	34.1%	10.8 %	40.8%	27.3 %	25%	1%
Neutral	28.6%	20.7 %	34.1%	11.7 %	51.1%	18.4 %

Supporting	33.9%	66.8%	34.1%	66.8%	35.4%	92%
Average	32.3%	32.7%	36.5%	36.5%	36.6%	37%

TABLE II. Adding Negation Words and Similarity Features for Common Words between Post and Comments Features

	SVM		Naive Bayes		Decision Trees	
	Precision	Recall	Precision	Recall	Precision	Recall
Attacking	40.3%	30.4%	42.8%	22.7%	34.6%	6.3%
Neutral	46.5%	23%	49.3%	28.9%	69.9%	281%
Supporting	35.5%	61.3%	35.8%	67.2%	37.4%	90.1%
Average	40.3%	30.4%	42.5%	39.6%	46.6%	41.3%

TABLE III. Using All Words in Posts and Comments Features

	SVM		Naive Bayes		Decision Trees	
	Precision	Recall	Precision	Recall	Precision	Recall
Attacking	66.7%	67.6%	59.2%	51.6%	63.8%	66.5%
Neutral	75%	62.2%	61.2%	53.7%	75.8%	58.1%
Supporting	77.3%	82.9%	61.1%	76.4%	71.7%	85.2%
Average	73%	72.7%	60.5%	60.5%	70.4%	70%

TABLE IV. Adding Negation Words and Similarity Features for All Words in Posts and Comments Features

	SVM		Naive Bayes		Decision Trees	
	Precision	Recall	Precision	Recall	Precision	Recall
Attacking	67%	68.7%	58.9%	49.5%	64.1%	68.7%
Neutral	75.8%	62.6%	61.3%	53.3%	71.1%	59.3%
Supporting	77.5%	88.9	59.7%	77.1	73.3%	80.1
Average	73.4%	73.4%	59.9%	59.9%	69.5%	69.4%

TABLE V. Show Precision/ Recall for two human experts that show the difference between people in detect the class of comment

	Precision	Recall
Attacking	87.4%	31%
Neutral	79.8%	25.8%
Supporting	72.4%	33.8%
Average	79.8%	30.2%

V. Conclusions

In this paper, we used Facebook to collect training data to perform a sentiment analysis. We constructed corpora for supportive comments, attacking comments, and neutral comment with regard to different posts. We tried different groups of features. We improved them by adding similarity and sentiment words features. We use different classifiers; support vector machines, naive bayes, and decision tree. The best result was obtained by the support vector machine classifier. We could reach up to 73.4% of accuracy on their test set.

References

[1] Shu-Chuan Chu," VIRAL ADVERTISING IN SOCIAL MEDIA: PARTICIPATION IN FACEBOOK GROUPS AND RESPONSES AMONG COLLEGE-AGED USERS", Journal of Interactive Advertising, 2011

[2] Bo Pang , Lillian Lee, "Opinion Mining and Sentiment Analysis, Foundations and Trends in Information Retrieval", v.2 n.1-2, p.1-135, January 2008

[3] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 79–86, 2002

[4] EfthymiosKouloumpis,TheresaWilson,JohannaMoore,"Twitter Sentiment Analysis:The Good the Bad and the OMG!", Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media,2011.

[5] Alexander Pak, Patrick Paroubek "Twitter Based System: Using Twitter for Disambiguating Sentiment Ambiguous Adjectives," Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pages 436–439.

[6] Changhua Yang, Kevin Hsin-Yih Lin, and Hsin-Hsi Chen. 2007. Emotion classification usingweb blog corpora. In WI '07: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, pages 275–278, Washington, DC, USA. IEEE Computer Society.

[7] Anna Huang, Similarity Measures for Text Document Clustering, New Zealand, Computer Science Research Student Conference 2008, April 2008.

[8] C. D. Manning and H. Schutze. Foundations of statistical natural language processing.MIT Press, 1999.

[9] [White and Liu, 1994] A.P. White and W.Z. Liu. Bias in Information-based measures in decision tree induction. Machine Learning, 15:321-329, 1994.

[10] GLENN M. FUNG, O. L. MANGASARIAN, Multicategory Proximal Support Vector Machine Classifiers, 2005 Springer Science + Business Media, Inc. Manufactured in The Netherlands.

[11] Ibrahim Abu El-Khair, Effects of stop words elimination for Arabic information retrieval a comparative study, study, International Journal of Computing and Information Sciences, Decembers 2006.

[12] Cyril Goutte and Eric Gaussier, A Probabilistic Interpretation of Precision, Recall and F-score, with Implication for Evaluation, The final version of this paper will appear in:D.E. Losada and J.M. Fernaandez-Luna (eds) Proceedings of the European Colloquium on IR Resarch (ECIR'05), LLNCS 3408 (Springer), pp. 345-359