The Workshops of the Thirtieth AAAI Conference on Artificial Intelligence
Knowledge Extraction from Text: Technical Report WS-16-10

EmoGram: An Open-Source Time Sequence-Based Emotion Tracker and Its Innovative Applications

Aditya Joshi,[1,2,3] **Vaibhav Tripathi,**[1] **Ravindra Soni,**[1]
Pushpak Bhattacharyya,[1] **Mark James Carman**[2]
[1]IIT Bombay, India, [2]Monash University, Australia
[3]IITB-Monash Research Academy, India
{adityaj, vtripathi, pb}@cse.iitb.ac.in

Abstract

In this paper, we present an open-source emotion tracker and its innovative applications. Our tracker, *EmoGram*, tracks emotion changes for a sequence of textual units. It is versatile in terms of the textual unit (tweets, sentences in discourse, etc.) and also what constitutes the time sequence (timestamps of tweets, discourse nature of text, etc.). We demonstrate the utility of our system through our applications: a sequence of commentaries in cricket matches, a sequence of dialogues in a play, and a sequence of tweets related to the Maggi controversy in India in 2015. That one system can be used for these applications is the merit of *EmoGram*.

Introduction

Emotion Analysis (EA) is the task of identifying emotions in text. For example, a sentence '*I loved the new film*' will be tagged as happy, while '*I am worried about my examination results*' will be predicted as anxious. Existing emotion analysis approaches use several rule-based and statistical techniques (Liu, Lieberman, and Selker 2003) (Strapparava and Mihalcea 2007) (Yang, Lin, and Chen 2007). Emotion analysis has been widely studied (Liu, Lieberman, and Selker 2003; Yang, Lin, and Chen 2007; Strapparava and Mihalcea 2007; Danisman and Alpkocak 2008). Some emotion analysis approaches in the past have considered sequential/narrative-based applications (Mohammad 2011; Kazemian, Zhao, and Penn 2014; Nalisnick and Baird 2013; Bollen, Pepe, and Mao 2009). Similarly, we consider textual units that form a sequential series (say, tweets published by a person on subsequent days). We present an emotion tracker, a system that **performs emotion analysis for textual data arranged in a time sequence** (*for example*, date, sequence number in a discourse, etc.). We call our emotion tracker, '*EmoGram*'. The name is derived from Electro Cardio Gram (ECG), a diagnostic tool which shows the electrical activity of the heart over a period of time. Similarly, *EmoGram* shows emotional activity over a time sequence. *EmoGram* has been implemented for **twitter data, but works for different kinds of data, as we show in the applications**. We demonstrate three applications of *EmoGram*: (a) How emotions in a cricket match change, based on cricket commentaries (where deliveries in a match form the time sequence), (b) How emotions of characters in a play change, based on their dialogues (where sequence of dialogues form the time sequence), (c) How sentiment towards a product changes, based on tweets (where tweets arranged according to their timestamps result in the time sequence). The code is at: *https://github.com/adityajo/emogram*.

Copyright © 2016, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

EmoGram: Architecture

EmoGram is a lexicon-based emotion tracker that displays emotion in tweets on a timeline axis based on emotion scores for each of the emotion labels. *EmoGram* can be used in two ways:

1. **Keyword is the opinion holder**: If tweets from a twitter handle are downloaded, *EmoGram* gives you how the given twitter user has been feeling over a time period.
2. **Keyword is the opinion target**: If tweets are searched using a keyword, *EmoGram* gives you how users of twitter feel about the given keyword over a time period.

The emotion labels considered are happy, sad, anxious and angry.

Input/Output Details

The input to *EmoGram* is a keyword to download stream of textual units (say, tweets). The output is a visual display of emotion in this series of tweets. We refer to this line graph as emotion time sequence graph.

Emotion Label	Emo-Lex	LIWC
Happy	692	342
Sad	1192	102
Anxious	841	92
Angry	1250	184

Table 1: Lexicon statistics for the four emotion labels

Architecture

The architecture of *EmoGram* is shown in Figure 1. *EmoGram* is a four-stage system that consists of: (a) **Twitter Downloader** that downloads tweets based on the option selected and the keyword, (b) **Tweet Emotion Scorer** that assigns an emotion score to each tweet, (c) **Emotion Scorer**

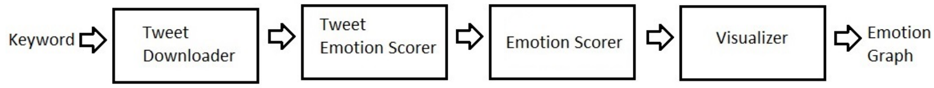

Figure 1: EmoGram: Architecture

that assigns an overall emotion score to a time period, (d) **Visualizer** that illustrates the emotion series.

Twitter Downloader Twitter Downloader downloads tweets using the Twitter4J API (http://www.twitter4j.org) as: user handle name, text and date.

Tweet Emotion Scorer The input to the tweet emotion scorer is a tweet while the output is a four-tuple score corresponding to four emotion labels. Tweet Emotion Scorer is a rule-based implementation that uses a lexicon. We experiment with two lexicons: LIWC (Pennebaker, Francis, and Booth 2001) and Emo-Lex (Mohammad and Turney 2013). Table 1 shows the dictionary sizes for each of the labels.

Sentiment may be expressed by words but is modified by syntactic constructs. Hence, we implement the following linguistic sub-modules:

- **Negation Handler**: We match common negation words such as *not, neither*, etc. and set a negation flag until the end of the sentence or a "*contradicting conjunction*".
- **Conjunction Handler**: Some conjunctions support sentiment while some invert it. One such conjunction is "*but*". These conjunctions are of particular importance in case of handling discourse for sentiment or emotion analysis. We set a contradicting conjunction flag until the end of the tweet in case such a conjunction is observed.
- **Inhibitor Handler**: LIWC gives a list of inhibiting words. An example of such a word is '*stop*'. Consider the sentence '*He stopped corruption*'. Since '*corruption*' is a negative word and the person stopped something negative, the overall sentiment is positive. Hence, we also have an inhibitor handler flag set until the end of the tweet if such a word is observed.

Using the linguistic sub-modules, the tweet emotion scorer works as follows:

- **Angry/Anxious Scorer**: The score associated with the emotion labels "*Angry*" and "*Anxious*" is the number of angry/anxious words present in the tweet. If the negation or inhibitor flag is set, the word is not counted. This can be evident from an example such as '*I am not angry*".
- **Happy/Sad Scorer**: This score is different from the other emotion labels because happy and sad are contrary to one another. In other words, the negation, conjunction or inhibitor flag indicates that a happy word would imply the emotion 'sad' whereas a sad word would imply the emotion 'happy'. Hence, we generate scores for happy and sad using the lexicon and the linguistic sub-modules described above.

The output of Tweet Emotion Scorer is a four-tuple corresponding to each of the four emotion labels.

Emotion	Emo-Lex	LIWC
Anxious	19	**27**
Angry	29	**29**

Table 2: Accuracy (%) of *EmoGram* for two labels: anxious and angry, for two lexicons: LIWC and Emo-Lex

Emotion	HP	HR	SP	SR
Emo-Lex	57.8	16.3	67.3	23.5
LIWC	**60.7**	**29.3**	**67.9**	**45.13**

Table 3: Happy/Sad Precision Recall (HP, HR, SP, SR) of *EmoGram* for two labels: happy and sad, for two lexicons: LIWC and Emo-Lex

Emotion Scorer The Emotion Scorer assigns an emotion score to all labels for a given time period. This module works as follows:

- The module is called with a given time period. All tweets with the given time period (say, minute, day, etc.) are selected.
- For each emotion label,
  - The score for all tweets corresponding to that emotion label are averaged. The resultant score is assigned for a given time period-emotion label pair.
- The output of the Emotion Scorer is then assigned to a given time period.

For example, this module may be called for the past thirty days in order to obtain emotion label scores for each of these days. We do understand that in case we do not have sufficient tweets for a given day, the emotion scorer may not work well.

Visualizer The visualizer was developed using Plot LY [1]. Plot LY is a Python-based package for construction of graphs. It provides a graph generated from the data that has dates on the x-axis and the score on the y-axis. We call this the '**emotion time sequence graph**'. There are four lines on the graph each corresponding to emotion labels.

Evaluation

In this section, we evaluate the performance of *EmoGram* in case of individual tweets. The next section on applications validates that it works well for a sequence of textual units. In order to evaluate EmoGram for individual tweets (i.e. the Tweet Emotion Scorer), we download a set of tweets using

[1] https://plot.ly/

the Twitter API[2] based on hashtags #happy, #sad, #anxious, #angry. This technique has become increasingly popular to obtain distant supervised labels for many past works (Purver and Battersby 2012). This results in a test dataset of 515 angry, 709 sad, 654 hppy and 957 anxious tweets. We then test the accuracy of our system for this labeled dataset of tweets. Table 2 summarizes the performance of *EmoGram* for anxious and angry, whereas Table 3 gives precision/recall values for happy and sad labels. The results are spread over two tables because happy-sad are opposite emotions and hence, class-wise precision-recall can be computed, while this does not hold for anxious and angry. We observe that LIWC performs better than Emo-Lex for *EmoGram*. Hence, we use LIWC for the applications in the forthcoming sections.

Applications

We now describe three innovative applications of *EmoGram*. They differ in their definition of a '*time sequence*'. We reiterate that while *EmoGram* was originally developed to work for tweets, these applications demonstrate its utility in case of different text forms. Table 4 provides a summary.

Event: Cricket Commentaries

The first application is emotion extraction from a live commentary of a sport such as cricket. In this case, the time series is **a sequence of deliveries** in a cricket match. We extracted full match commentaries from http://www.espncricinfo.com for three cricket matches. For better comparison, we selected matches which we expected to be different from each other in terms of the excitement levels: (a) Australia vs New Zealand 2015 World Cup Final (**AUS-NZ**) (this match became predictable towards the end), and (c) India vs Sri Lanka 2011 World Cup Final (**IND-SL**) (this match had a close finish).

Our dataset consisted of 556 and 551 sentences each for the matches, respectively, collected on a ball-by-ball basis. We used *EmoGram* to generate emotion time sequence graphs for the three matches as shown in Figures 2 and 3. For better visualization, we show a period of three overs as a single point on our time sequence.

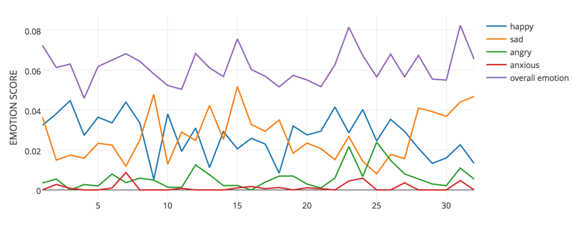

Figure 2: Emotion time sequence graph for Cricket Commentary: India v/s Sri Lanka. x-axis: Timeline of the match. y-axis: Magnitude of emotions

EmoGram captures that the distribution of emotion was different in the matches. For example, the overall emotion

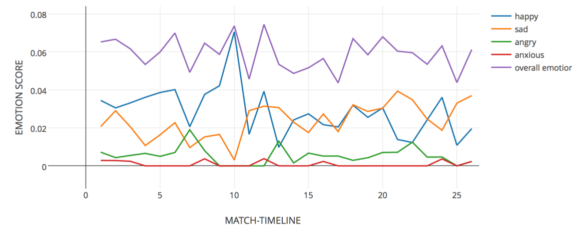

Figure 3: Emotion time sequence graph for Cricket Commentary: Australia v/s New Zealand. x-axis: Timeline of the match. y-axis: Magnitude of emotions

drops down towards the end of the AUS-NZ match because the match did not have a close finish and became less exciting. On the other hand, the excitement level in the IND-SL match shows a steep rise towards the end of the match because of the increased anticipation/excitement. On manual inspection, we observe that errors may arise because of digressions or discussions of past events (for example, a player's performance in the past matches).

Discourse: Characters in a Play

The second application deals with emotion expressed in a discourse. In discourse/narratives, there may be a change in emotions across sentences as described in (Mishra, Joshi, and Bhattacharyya 2014). A time sequence-based analysis of these emotions may help us understand different moods of characters in the narrative. (Nalisnick and Baird 2013) present a similar study for Shakespeare's plays.Towards this, we select **sequence of dialogues in a play** as it appears in the script of a play. We experiment with an English play

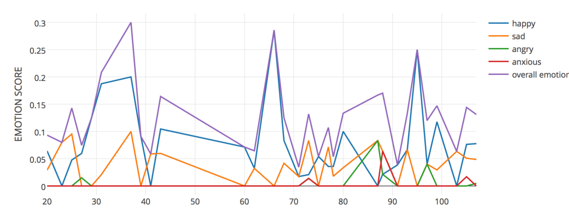

Figure 4: Emotion time sequence graph for dialogues of 'Arjuna' in the play 'Chitra'. x-axis: Dialogue sequence number. y-axis: Magnitude of emotions

called 'Chitra' by Rabindranath Tagore[3]. We consider dialogues of two central characters of the play: Arjuna (the hero) and Chitra (the heroine). We separate dialogues spoken by these two characters, and consider them as a sequence of dialogues. This results in 49 dialogues for Chitra, and 37 dialogues for Arjun. Figures 5 and 4 show the emotions of Chitra and Arjuna, respectively, along the sequence of dialogues. Figure 4 shows that Arjuna is happy in

[2]https://dev.twitter.com/overview/api

[3]https://en.wikipedia.org/wiki/Chitra_\%28play\%29

Application	Author(s)	Text form	Time Sequence	Strengths of *EmoGram*	Open Challenges
1) Event	Sports Commentator	Cricket commentaries	Deliveries in a cricket match	Captures excitement levels	Digressions in commentary such as references to past performance
2) Discourse	Playwright	Dialogues in a play	Dialogue sequence	Captures emotion changes of characters	Emotion interaction between characters
3) Entity-specific Trend	Twitter users	Tweets about an entity	Timestamps	Captures emotion trends towards a product	Subjective extraction may be required for news items that contain emotion words

Table 4: A summary of applications of *EmoGram* with its strengths and open challenges

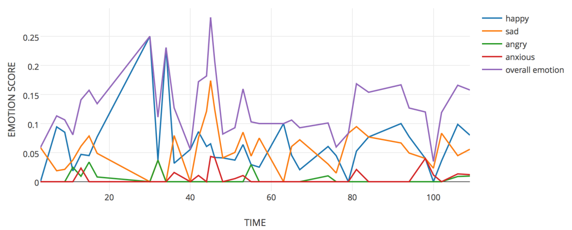

Figure 5: Emotion time sequence graph for dialogues of 'Chitra' in the play 'Chitra'. x-axis: Dialogue sequence number. y-axis: Magnitude of emotions

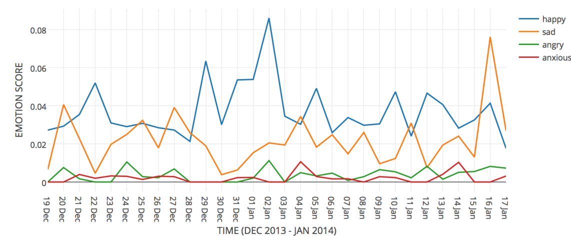

Figure 7: Emotion time sequence graph posted by a celebrity who later committed suicide. x-axis: Sequence of dates, y-axis: Magnitude of emotions

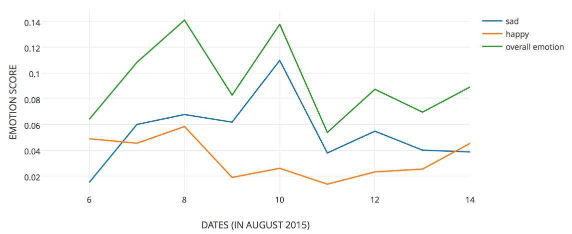

Figure 6: Emotion time sequence graph for the hashtag '#MaggiBan'. x-axis: Date of August 2015, y-axis: Magnitude of emotions

the beginning (score: 20) and the end of the play, although he is not as happy in the end (score: 15). On manual inspection of the dialogues, we validate that there are two happy monologues in the beginning and the end of the play. Similarly, the spike around 70 is due to a long monologue where Arjuna uses dramatic expressions to exhibit his longing for his lover, Chitra. Figure 5 shows that Chitra's character has a happy ending. In the climax, Chitra is elated because she elopes with her lover.

Entity-specific trends: Product reputation

The third application deals with the emotion trends on social media. This is important in case of controversial events that may affect the reputation of a brand. In this case, the time series is **a sequence of dates** related to an event. Towards this, we selected the Maggi noodles controversy in India (http://zeenews.india.com/business/news/companies/timesequence-of-maggi-noodles-ban_133742.html). In August 2015, Maggi was initially banned due to food safety regulations, but the ban was later revoked. We downloaded tweets containing the hashtag #MaggiBan, along with their timestamps. We obtained a set of 9534 tweets, each with the creation date. Figure 6 shows the overall emotion towards the The X-axis represents the day of August, 2015, while the Y-axis represents the magnitude of emotions corresponding to the day. We observe that on 14^{th} August, a day after the Maggi ban was revoked, there is a positive sentiment - and for the first time in several days, the happy emotion is overrides the sad emotion.

Conclusion & Future Work

This paper presented our emotion tracker called *EmoGram*. It consists of: (a) Tweet Downloader (to download tweets), (b) Tweet emotion scorer (to predict emotion in a tweet), (c) Emotion Scorer (to combine emotions in tweets of a given time period), and finally, a (d) Visualizer that places these emotions on a time axis, to generate emotion time sequence graphs. We evaluated the emotion tracker using a tweet dataset. Then, we described three applications, each differing in (a) the text form, and (b) the way a time sequence is defined. The first application considered the time sequence as a set of deliveries in a cricket match. For the second application, a play was considered to be a sequence of dialogues. In the third application, we considered the recent Maggi controversy and validated how change in real-world events correlated with emotions in tweets. That one system can be used for these applications is the biggest merit of *EmoGram*. Our applications can be extended to automatic generation of highlights for games based on commentaries, mental health monitoring, etc.

References

Bollen, J.; Pepe, A.; and Mao, H. 2009. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. *arXiv preprint arXiv:0911.1583*.

Danisman, T., and Alpkocak, A. 2008. Feeler: Emotion classification of text using vector space model. In *AISB 2008 Convention Communication, Interaction and Social Intelligence*, volume 1, 53.

Kazemian, S.; Zhao, S.; and Penn, G. 2014. Evaluating sentiment analysis evaluation: A case study in securities trading. *ACL 2014* 119.

Liu, H.; Lieberman, H.; and Selker, T. 2003. A model of textual affect sensing using real-world knowledge. In *Proceedings of the 8th international conference on Intelligent user interfaces*, 125–132. ACM.

Mishra, A.; Joshi, A.; and Bhattacharyya, P. 2014. A cognitive study of subjectivity extraction in sentiment annotation. *ACL 2014* 142.

Mohammad, S. M., and Turney, P. D. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence* 29(3):436–465.

Mohammad, S. 2011. From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, 105–114. Association for Computational Linguistics.

Nalisnick, E. T., and Baird, H. S. 2013. Character-to-character sentiment analysis in shakespeares plays.

Pennebaker, J. W.; Francis, M. E.; and Booth, R. J. 2001. Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates* 71:2001.

Purver, M., and Battersby, S. 2012. Experimenting with distant supervision for emotion classification. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, 482–491. Association for Computational Linguistics.

Strapparava, C., and Mihalcea, R. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, 70–74. Association for Computational Linguistics.

Yang, C.; Lin, K. H.-Y.; and Chen, H.-H. 2007. Emotion classification using web blog corpora. In *Web Intelligence, IEEE/WIC/ACM International Conference on*, 275–278. IEEE.