Application-Oriented Pattern Extraction for Emotion Event Detection

|  |  |  |
| --- | --- | --- |
| [name]  [address]  [address]  [address]  [e-mail] | [name]  [address]  [address]  [address]  [e-mail] | [name]  [address]  [address]  [address]  [e-mail] |

|  |  |
| --- | --- |
|  |  |

Abstract

The system-provided tags for posters to record their current emotions on social websites have served as useful materials for emotion analysis on social media. LiveJournal[[1]](#footnote-1), which collects a large amount of user posts and their respective emotions from 132 options, is one of the system providing these tags and thus very often is utilized for developing emotion analysis techniques. Detecting emotions from web posts often has been treated as a text classification problem and various features extracted from posts have been tested for their usefulness in machine learning methods. This paper approaches this research problem from a different perspective by recognizing emotion events using learned patterns. With the sentence structure from the parsing tree, our method keeps as many verbs as possible and removes trivial information to put a whole emotion event in a nutshell. Starting with pattern extraction, our system not only can determine the emotion class of an input post but also can bring out major events of a specific emotion by important patterns. From the analysis of these major events, we also find that same events happening to different persons arouse different emotions. Experiments on 16,000 posts labeled with one of 40 major emotion tags from LiveJournal show that our method outperform the previous results. In addition to evaluate the performance by the traditional metrics accuracy and AUC (Area Under the Curve) for the classification problem, we propose a new evaluation metric MED (Mean Emotion Distance) which considers the strength and likelihood of emotions according to the emotion wheel. MED will give larger penalty to the comparably wrong answer among incorrect ones, and it has the potential to display the emotion of a post in a 3D chromatic visualization model for further applications.  Overall, the proposed system and evaluation metric MED in this paper facilitate the process of improving the emotion classification techniques.

1. Introduction

Emotion analysis has attracted much attention in research communities due to the close relativeness to people and its large amount of applications. The emotions researchers tried to detect were usually defined by psychologists Ku et al. (2012) or social media websites where people can express their feelings by writing articles and labeled themselves a specific emotion tag. The number of major emotions defined by psychologists usually range from 5 to 10 including those like happy, angry, sad, surprise, disgust, etc. However, detecting emotions in the articles from the websites is usually more challenging. There are two reasons for this: there are usually more emotion categories which include minor emotions defined by psychologists, and there are also informal emotions like “empty”, “blah”, “sick” included. *LiveJournal* is one of such social media websites. It allows users to freely note their emotions or select the most appropriate one from 132 predefined tags when posting articles. Posts in LiveJournal are often selected as experimental materials for emotion classification in the research community because of availability of correct answers.

Detecting emotions from texts is often treated as a text classification problem. Therefore, traditional approaches and machine learning approaches for classification are widely used for solving this problem. However, Internet articles contain much noise due to the casual writing style. Therefore, the performance is difficult to improve. Mishne (2005) has done some thorough experiments to show that for articles in *LiveJournal*, even people can only classify them to the degree of 60% accuracy. Moreover, the traditional evaluation methods for classification, i.e., accuracy and AUC, view each class as of equal importance. This is not true for emotion classification as some emotions do relate to each other more closely. This could be the reason that the performances of emotion classification on articles of LiveJournal reported in literature had little significance. Therefore, though using materials from LiveJournal for research are very handy and it can work as a benchmark, it is not easy to decide which system is more superior based on the experimental results.

In this paper, we also hope to utilize the LiveJournal materials to develop techniques for emotion classification. We try to solve the mentioned problem from two aspects. First, we propose a pattern-based approach to capture emotion events and avoid noise in web articles. Second, we propose a new evaluation metric Mean Emotion Distance (MED) which considers the relations between emotions and gives the less appropriate answer more penalty. In the process of designing methods and performing experiments, we will highlight the advantages of the proposed pattern-based approaches by showing example instances and evaluating using traditional metrics and MED.

1. Related Work

Emotion recognition has been an active research field. Numerous approaches have been proposed for detecting different emotions from textual data.

Mihalcea & Liu employed a corpus-based approach to extract salient keywords for seeking out *happy* and *sad* emotions in the blog posts. Tao adopted the keyword spotting method to build a lexicon from context. Each keyword in the lexicon contains six emotion tags and corresponding weights. Similarly, Yang et al. created emotion lexicons using Yahoo! Kimo Blog as corpora. In their studies, emoticons were introduced to identify emotions associated with textual keywords. Chaumartin developed a rule-based system, which utilized the linguistic information from knowledge bases to detect six specific emotions in news headlines. The results show that their approach is capable to achieve high accuracy on emotion and valence annotations with the help of several lexicon resources, including *WordNet*, *WordNet-Affect* and *SentiWordNet*. In addition to surface keywords or emoticons, Liu et al. presented a set of commonsense-based linguistic affect models, which employed a real-world knowledge base of commonsense to recognize six basic emotions at the sentence level.

Apart from using handcrafted models or rule-based approaches, a number of researches considered emotion recognition task as a classification problem. Mishne conducted preliminary experiments in classifying emotions of blog text at the document level. He used a variety of features for the classification process and utilized the *SVM* classifier to identify the intensity of the community mood. Tokuhisa et al. proposed a two-step method for the sentence-level emotion classification. Firstly, the sentences are grouped into two categories, emotion-involved and neutral using a *SVM* classifier; then the sentences tagged with emotion-involved are further classified into ten emotion classes by a k-nearest-neighbor (*kNN*) classifier.

Each method has its advantage and shortage, for example, approaches utilizing human-craft lexicon and pre-defined rules often perform better than statistical machines learning model in small corpus. However, building lexicons could be painstaking, time-consuming and might not achieve high coverage of emotion keywords and events. Therefore, Yang et al. built a hybrid model that incorporates lexicon-based keyword spotting and several machine learning-based emotion classification models. The results generated by each modules are integrated using vote-based strategies and it demonstrates that the system is capable of recognizing 15 different emotion in the suicide notes.

Although several natural language processing techniques have been used in the hybrid model and they could well complement each other to achieve high accuracy, it is still challenging to recognize emotions in sentences containing implicit affective meaning. For example, in this sentence, “Recently, I really wish I have someone staying with me”, we can easily recognize the emotion *lonely* even there is no explicit emotion expression. To discover the implicit events, we introduced a ***Scenario Identification*** method utilizing grammatical relations as well as a set of pre-specified semantic structures. After that, we adopted a hybrid model to combine the ***Scenario identification*** with machine learning classifier aiming to effectively recognize emotions in blog posts.

1. Constructing the Corpus

The training and testing corpora used in this study comprise blog posts from *Livejournal*, a free weblog service with over 42 million registered users and 2.1 million active users. The most distinguished feature of *Livejournal* is that its web interface allowing users to update their blog with a optional mood tag, which can be chosen from a predefined list with 132 tags (e.g., *angry* and *happy*), or be typed in free-text.

Similar to the experiments conducted in the previous works (Mishne 2005, Yang and Liu 2012), we used the data set, LJ40k, which is a subset of the 21 million posts Leshed and Kaye collected in 2005. LJ40K contains 40,000 blog posts tagged in top 40 occurring moods in the entire corpus, i.e., 1,000 blog posts in each mood. The number of words in the corpus is 14,284,230 (average of 357 words per post), while the unique number of words is 266,506. Table 1 shows the distribution of the most popular moods in LJ40K.

An additional point to note regarding this corpus is that it does not constitute a representative sample of writer while containing a large amount of users. According to Livejournal, many of the blog writers are not adults (the average of bloggers is about 18).

1. Pattern

We focus on the issue of recognizing emotions in blog posts. It is still challenging for past approaches using shallow linguistic features such as n-grams, POS tags and word frequency to detect emotions when there is barely affective keyword existed in a sentence. Moreover, people often convey their mental states by describing a certain event rather than using direct or explicit emotion words in blog posts. For example, consider this post, *I got hit by a drunk driver.*, it’s never difficult for human to capture the negative (e.g., *angry* or *depressed*) emotion event without any affective keywords. Thus, it is important to incorporate deep linguistic information to help capture such events, which can precisely reflect the authors’ intended emotions.

* 1. Pattern extraction

Due to insufficient affect keyword in emotional expressions, previous research use fragments of sentences as features aiming at better capturing emotion events. The most widely used fragments are word sequences, i.e., *n-grams*, with varied window sizes. (2005 Mishne, 2008 Tokuhisa)

However, low-order n-grams such as unigrams, bigrams are often not very comprehensible by human. On the other hands, a higher order n-gram will have the generated text repeat tremendous portions of the source text. It leads to the result that researches often limit the length of n-grams for computational efficiency.

Therefore, this study aims to extract more informative fragments or multiword expressions which are representative of certain event. In the mean time, researches in linguistic indicate that verbs play a central role in the syntactic and semantic interpretation of a sentence (Joanis 2003).

According to this results, we extract verbs and corresponding arguments by grammatical relations and semantic roles to construct verb argument frames. Consider this sentence, for example, *I have a pretty good chance of getting accepted!*, which is a post in LiveJournal labeled as the *excited* mood tag. We identify the verb *have* along with its subject *I* and direct object *chance* to form a verb argument frame: <*I, have, chance*>.

Compare to previous approaches using n-grams (e.g., *I have a*, *have a pretty),* verb argument frames (e.g., <*I, have, chance*>) may be able to carry much closer meaning to the origin sentence. In addition, large amounts of n-grams generated as the window size increased to cover all arguments of a verb, in this case, the window size is 6, i.e., *I have a pretty good chance*.

In this paper, we preliminarily consider four major grammatical relations of verb arguments: *subject*, *object*, *negation* and *prepositional phrase*, to construct patterns.

* 1. Pattern Scoring

After extracting patterns which may represent events from posts, we need a scoring function to tell us their strength of relations to designated emotions. As mentioned, we will evaluate whether the extracted patterns can detect emotion events by seeing whether they can help in label the emotion of web posts using machine learning tools. We find two issues here. One is that using patterns as features in machine learning models is not feasible as the feature dimension is too high (in our case, ?????). Therefore, the designed scoring function should be able to reduce dimensions to a certain degree. The other issue is that the scores of patterns should enable us to select important causal emotion events for further application. As a result, we hope this scoring function can give scores to answer the following question: does a certain pattern (and the implicit emotion event) lead to the current emotion? Assume there are *n* emotions in the corpus. To answer this question, we design a score *s(p,e)* of a pattern *p* to be of the current emotion *e*, a score *s(p,-e)* of the same pattern to be *not* of the current emotion *e*, determining by the other *(n-1)* emotions in the corpus except *e*. Moreover, we hope the final score *S(p, e)* could be normalized to the range from 0 to 1. From observations, at least four requirements according to the number of observed occurrences of the pattern in each emotion category *f(p,e)* should be fulfilled for this purpose:

**Requirement 1**: if the number of occurrence for *p* in each emotion categories has only two values 0 and *k* and *f(p,e)* equals *k*, *S(p,e)* should be 1 when all *f(p, -e)* are 0 and strictly monotonously decrease to 0.5 when all, a total of n-1, *f(p, -e)* are 1. In other words, if *f* are equal in all emotion categories, the score *p* is of *e* should be equal to *p* is *not* of *e*, and thus equals 0.5. Except *f(p,-e)* are all 1 or all 0, if there are the same number of *k* in *f(p,-e)*, when *k* is larger, *S(p,e)* should become larger too.

**Requirement 2**: if *f(p,e)* is closeto the sum of all *f(p,-e)* and all value of *f(p,-e)* are almost the same (evenly distributed), if only *f(p,e)* is not 0, *S(p,e)* equals 1; if *f(p,e)* is close to each *f(p,-e)*, *S(p,e)* decreases to a value between 0.5 and 1; if *f(p,e)* gets larger, *S(p,e)* gets larger again and when *f(p,e)* is comparably much larger than all *f(p,-e)*, *S(p,e)* will be very close to 1.

**Requirement 3**: if *f(p,e)* is comparably larger than the sum of all *f(p,-e)* and all value of *f(p,-e)* are almost the same (evenly distributed), if only *f(p,e)* is not 0, *S(p,e)* equals 1; when the sum of *f(p,-e)* gets larger (but still evenly distributed in all categories), *S(p,e)* will strictly monotonously decrease.

**Requirement 4**: if *f(p,e)* is fixed and only one *f(p,-e)* is increasing, *f(p,e)* decreases. To a certain point of time, *f(p,e)* will be less than 0.5.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *Requirement 1*   |  |  | | --- | --- | | *f(p,e)* | *f(p,-e)* | | k | 0, 0, 0, …, 0 | | k | k, 0, 0, …, 0 | | k | k, k, 0, …, 0 | | … | … | | k | k, k, k, …, k | | *Requirement 2*   |  |  |  | | --- | --- | --- | | *f(p,e)* | *f(p,-e)* | | | 1 | 0, 0, 0, …, 0 | | | 2 | 1, 0, 0, …, 0 | | | 3 | 1, 1, 0, …, 0 | | | … | … | | n | 1, 1, 1, …, 1 | | |
| *Requirement 3*   |  |  | | --- | --- | | *f(p,e)* | *f(p,-e)* | | 100 | 0, 0, 0, …, 0 | | 100 | 1, 0, 0, …, 0 | | 100 | 1, 1, 0, …, 0 | | … | … | | 100 | 1, 1, 1, …, 1 | | *Requirement 4*   |  |  |  | | --- | --- | --- | | *f(p,e)* | *f(p,-e)* | | | 5 | 0, 0, 0, …, 0 | | | 5 | 1, 0, 0, …, 0 | | | 5 | 2, 0, 0, …, 0 | | | … | … | | 5 | 39, 0, 0, …, 0 | | |

Table X. Examples of Four Requirements

A simple scoring function generates *S(p,e)* which fulfills these four requirements is designed as follows:

|  |  |
| --- | --- |
|  | (1) |
| , | (2) |
|  | (3) |

The score *s(p, e)* of pattern *p* for emotion *e* is actually its observed occurrences in *e*, *f(p,e)*, as shown in equation (1). To calculate s(p,-e), the distribution of f(p,-e) should be considered. The main idea of equation (2) is the 2-norm of f(p,-e) normalized by the 1-norm of f(p,-e). 2-norm here gives us a value considering all f(p,-e); when these values are similar, 2-norm of them is small, but when they varies, 2-norm is large. The magnitude of 2-norm is usually normalized by 1-norm. However, we still need the growth of the magnitude when f(p,-e) gets larger as the larger the observed occurrence, the more reliable the distribution information. Therefore, in equation (2), both 1-norm and 2-norm are not averaged by dividing by the total number and having the square root, respectively. This keeps the value of s(p,-e) increase linearly to the scale of f(p,-e). β is a smoothing factor designed to fulfill requirement 1. Equation (3) gives S(p,e), which is the final score assigned to p for emotion e. If s(p,e) is larger than s(p,-e), S(p,e) is above 0.5, and vice versa.

The proposed scoring function will give each pattern a total of *n* *S(p,e)* scores, one for each emotion category. It enables us to reduce the feature dimension of patterns to a factor of *n*. Moreover, we can give patterns multiple emotion labels according to these *n* scores. These scores not only can be calculated in a simple way and have many helpful functions; in our experiments, we will also show that they are effective for emotion classification.

(↓這個圖如果要放的話要重畫)

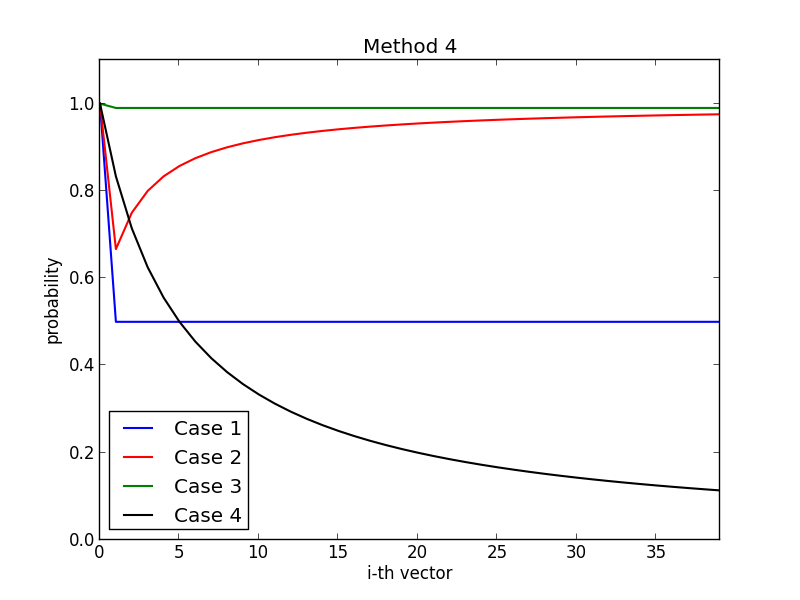


Figure X.

1. Feature Extraction

Keyword Frequency Counts

In emotion recognition, it is very common to regard a text as a set of words and use the frequency counts of those words as features. However, not all words are informative for recognizing emotions, i.e., neutral words and stop words. In our experiments, a word list from WordNet-Affect which contains 3,785 affective words is used. The generated bag-of-keywords features provide explicit emotional information. Besides, we adopt word lemma to remove tense and plurality.

Pattern Frequency Counts

Similar to the concept of representing a text by the words it holds, a text can also be regarded as a bag of patterns. Different from keywords, all patterns extracted from the corpus are kept as features except those that appear less than four times in the corpus.

While the affective keywords represent explicit emotional expressions, patterns represent implicit emotional expressions or events; thus these two feature sets can serve as complementary pairs.

Text Emotional Orientations

In blog articles, people often mention compound events with different emotional orientations but yield yet another overall emotion label. Besides, people tend to hold several relevant emotions at the same time, which means the emotional orientations are not independent. For example, a text labeled as *excited* may also has high emotional orientations in *happy* and *hopeful* but low emotional orientation in *depressed* and *sad*.

As a result, better performance on classification can be achieved if the orientations of different emotion labels are considered jointly. [reference?]

To acquire the emotional orientations of a blog article, the emotional orientations of each keyword or pattern included in the article should first be computed. We assume higher occurrence count of a keyword or pattern in an emotion category indicates that the keyword or pattern holds higher emotional orientation in that emotion category. From this point of view, we further use three types of values to represent emotional orientations — [frequency], [binary] ,and [scoring] — based on occurrence counts.

­­­[Table II?] shows an example of the computation of the emotional orientations features. The distribution of a certain keyword or pattern in 5 distinct emotion categories is listed, with the total count equals to 7+2+11+16+4 = 40. Given a cutoff percentage k%, the occurrence counts are accumulated from the top emotion until the accumulated sum exceeds 40\*k%. The emotion categories which have not been accumulated are then filtered out, or given zero value for their emotion orientations. The [frequency] feature vector is thus generated, as shown in [Table III].

[Table III] also shows the other two types of features. [binary] feature vector is to

Text emotional orientation = sum of keyword or pattern emotional orientation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | e1 | e2 | e3 | e4 | e5 |
| Count | 7 | 2 | 11 | 16 | 4 |

Table

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | e1 | e2 | e3 | e4 | e5 |
| Frequency | 0 | 0 | 11 | 16 | 0 |
| Binary | 0 | 0 | 1 | 1 | 0 |
| Scoring | 0 | 0 | 0.49 | 0.67 | 0 |

Table

Positions

Most previous works using linguistic features such as n-grams, POS tags and word frequency in text classification tasks did not consider where these linguistic features locate in an article. That is, the features extracted would be the same even if the order of sentences in the article are rearranged.

However, the order of the events a blogger narrates also provides some useful information for emotion recognition. This phenomenon is particularly obvious for some emotions. In the *sleepy* category from *LiveJournal*, for example, the blogger may write about events with various emotional orientations but concludes

1. Experiment and results

In this section, we start with an overview of the classification environment, and follow with a description of the experiments conducted and corresponding results.

* 1. Classification

Setup

We formulate the emotion recognition problem as a classification task and trained binary classifiers for each user mood class. The output of a classifier indicates whether the mood is related to a post or not.

For the experiments, we held out 32,000 posts of LJ40K as the training set, and the remaining 8,000 for testing. As for training, we adopted the support vector machine (SVM), which is widely used in text classification tasks since they scale to the large amount of features, with radial-kernel implemented by the LIBSVM.

For each mood, to reduce the risk of sampling bias, we created balanced training set consisting of all the positive data (800 examples), and an equal amount of negative ones (31,200 examples), randomly drawn form all other moods.

Experiment

In this paper, we

use the entire list of features given above, rather than select subsets of it and experiment with them separately. This was done due to space constraints; our ongoing work includes evaluating the performance gain contributed by each feature subset.

* 1. Results used

|  |  |
| --- | --- |
| Settings | Average accuracy |
| **kw** | 56.32% |
| **pt** | 53.41% |
| **kw-e-f** | 51.57% |
| **kw-e-b** | 50.73% |
| **kw-e-s** | 56.80% |
| **pt-e-f** | 51.58% |
| **pt-e-b** | 56.39% |
| **pt-e-s** | 57.68% |
| **kw + pt** | 58.09% |
| **kw + pt-e-s** | 59.58% |
| **kw-e-s + pt-e-s** | 59.63% |
| **kw + kw-e-s + pt-e-s** | 59.80% |

Table2:

1. Conclusion

According to the experiments conducted by Mishne (2005), even the human classification on *LiveJournal* can only achieve 63% accuracy. This indicates that the emotion tags of blog posts in *LiveJournal* reflect the authors’ subjective emotions, which makes this task much more challenging to machines. In this paper, we presented experiment results showing that our approach combing keyword spotting and learned patterns can achieve comparable accuracy with previous works (Yang and Liu [2012], Mishne [2005])

The task of emotion detection form web posts was usually treated as a classification problem, where the strength and likelihood of wrong answers were not considered. Using the proposed MED evaluation metric, the results show that our approach can rank the emotions based on their similarity to the users’ emotions more precisely by capturing implicit events. However, in our approach, the learned patterns are extracted based on sets of predefined syntactic structures. However, the set of structures that is most representative of implicit emotional events needs to be further explored through experiments or linguistic analysis.

Reference

Joanis, Eric, and Suzanne Stevenson. "A general feature space for automatic verb classification." *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2003.

Chaumartin, F. R. 2007. UPAR7: A knowledge-based system for headline sentiment tagging. In Proceedings of the 4th International Workshop on Semantic Evaluations, 422-425. Association for Computational Linguistics.

Klein, D., & Manning, C. D. 2003. Accurate unlexicalized parsing. In*Proceedings of the 41st Annual Meeting on Association for Computational Linguistics,* volume1, 423-430. Association for Computational Linguistics.

Ku, L. W., Sun, C. W., & Hsueh, Y. H. 2012. Demonstration of IlluMe: creating ambient according to instant message logs. In *Proceedings of the ACL 2012 System Demonstrations*, 97-102. Association for Computational Linguistics.

Leshed, G., & Kaye, J. J. 2006. Understanding how bloggers feel: recognizing affect in blog posts. In CHI'06 extended abstracts on Human factors in computing systems, 1019-1024. ACM.

Liu, H., Lieberman, H., & 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.

Mihalcea, R., & Liu, H. 2006. A Corpus-based Approach to Finding Happiness. In AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs, 139-144.

Miller, G. A. 1995. WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39-41.

Mishne, G. 2005. Experiments with mood classification in blog posts. In Proceedings of ACM SIGIR 2005 Workshop on Stylistic Analysis of Text for Information Access, volume 19.

Strapparava, C., & Valitutti, A. 2004. WordNet Affect: an Affective Extension of WordNet. In *LREC*, volume 4, 1083-1086.

Tao, J. 2004. Context based emotion detection from text input. InINTERSPEECH.

Tokuhisa, R., Inui, K., & Matsumoto, Y. 2008. Emotion classification using massive examples extracted from the web. In Proceedings of the 22nd International Conference on Computational Linguistics-volume 1, 881-888. Association for Computational Linguistics.

Yang, C., Lin, K. H. Y., & Chen, H. H. 2007. Building emotion lexicon from weblog corpora. In Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, 133-136. Association for Computational Linguistics.

Yang, H., Willis, A., De Roeck, A., & Nuseibeh, B. 2012. A hybrid model for automatic emotion recognition in suicide notes. Biomedical informatics insights, 5(Suppl. 1), 17.

Yang, Y. H., & Liu, J. Y. 2013. Quantitative study of music listening behavior in a social and affective context. IEEE Trans. Multimedia, 15(6).

1. http://www.livejournal.com/ [↑](#footnote-ref-1)