Prior and contextual emotion of a word in sentential context

Diman Ghazi, Diana Inkpen & Stan Szpakowicz

EECS, University of Ottawa
Ottawa, Ontario, Canada

uOttawa

Computer Speech & Language **28**(1), 2014

www.sciencedirect.com/science/article/pii/S0885230813000363

Outline

Outline

Motivation

- Expression of emotion or affect in text can be traced to words, but not just single words.
- Words are interrelated; they mutually influence their affect-related interpretation.
- Prior emotion – emotion ascribed to individual words – cannot capture the overall emotion of the sentence.

Prior versus contextual emotion of a word

Prior emotion

Contextual emotion

Prior versus contextual emotion of a word

Prior emotion

. . . the emotion listed for a word in an emotion lexicon.

Contextual emotion

Prior versus contextual emotion of a word

Prior emotion

. . . the emotion listed for a word in an emotion lexicon.

Contextual emotion

. . . the emotion of that word's context – here, a sentence.

Prior versus contextual emotion of a word

Prior emotion

. . . the emotion listed for a word in an emotion lexicon.

. . . *afraid* is associated with the emotion of **fear** in the WordNet-Affect lexicon.

Contextual emotion

. . . the emotion of that word's context – here, a sentence.

Prior versus contextual emotion of a word

Prior emotion

. . . the emotion listed for a word in an emotion lexicon.

. . . *afraid* is associated with the emotion of **fear** in the WordNet-Affect lexicon.

Contextual emotion

. . . the emotion of that word's context – here, a sentence.

. . . *afraid* in "I'm afraid it's going to rain." is only a vacuous polite term.

Outline

Related work

Emotion is a form of *sentiment*: a more general phenomenon whose simplest kind is polarity, the positive or negative orientation of a text.

- There is a lot of work on determining the presence of sentiment and in particular of polarity in texts, notably in short texts.
- Machine learning methods have been applied to corpus-based features, mainly unigrams, combined with lexical features (Pang *et al.* 2002, Alm *et al.* 2005, Aman & Szpakowicz 2007, Katz *et al.* 2007).
- Rule-based methods have also been attempted.

Related work

- Determining the presence of polarity in a text.
  - Pros: useful for some commercial application such as the analysis of consumer product reviews
  - Cons: insufficient in most realistic (or realistically scaled) applications

    (suicide notes are a dramatic example of a sub-genre whose analysis requires much more than polarity)
- Machine learning methods applied to corpus-based and lexical features.
- Rule-based methods.

Related work

- Determining the presence of polarity in a text.
- Machine learning methods applied to corpus-based and lexical features.
  - Pros: perform well and are simple to apply
  - Cons:
    - disregard negation, syntactic relations and semantic dependencies
    - require large annotated corpora for meaningful statistics and good performance
    - can take much processing time (and annotation effort is inevitably high)
- Rule-based methods.

Related work

- Determining the presence of polarity in a text.
- Machine learning methods applied to corpus-based and lexical features.
- Rule-based methods.
    - Pros: tend to ensure high precision
    - Cons:
        - require a substantial (manual) rule-creation effort, an expensive process with weak guarantee of consistency and coverage
        - are likely to be quite task-dependent.

Outline

Aman's dataset

- A collection of 4090 blog sentences, each annotated with Ekman's six emotions (happiness, sadness, anger, disgust, surprise, fear) or marked as not carrying an emotion.
- Highly imbalanced, with 68% of no-emotion sentences.
- Our three set of experiments work with different parts of the dataset:
  - all emotion sentences;
  - sentences with one emotion word according to WordNet-Affect (more in a while);
  - sentences with more than one emotion word according to WordNet-Affect.

Aman's dataset

- A collection of 4090 blog sentences, each annotated with Ekman's six emotions or marked as not carrying an emotion.
- Highly imbalanced, with 68% of no-emotion sentences.

hp	sd	ag	dg	sr	fr	ne	total
536	173	179	172	115	115	2800	4090

The distribution of labels in Aman's modified dataset.

The labels: happiness, sadness, anger, disgust, surprise, fear, no emotion.

Aman's dataset

	hp	sd	ag	dg	sr	fr	ne	total
1	536	173	179	172	115	115	0	1290
2	196	64	64	63	36	52	150	625
3	51	18	22	18	9	14	26	158
2 + 3	247	82	86	81	45	66	176	783

The distribution of labels in the portions of Aman's dataset in our experiments:

1. 1290 emotion sentences;
2. 625 sentences with one emotion word;
3. 158 sentences with more than one emotion word.

The labels: happiness, sadness, anger, disgust, surprise, fear, no emotion.

Lexicons

- **WordNet-Affect lexicon:** marked with Ekman's emotions.
- **Prior-Polarity lexicon:** over 8000 positive and negative words.
- **NRC Emotion lexicon:** a list of words for eight emotions (Ekman's six plus anticipation and trust).
- **Intensifier lexicon:** a list of 112 modifiers.

Lexicons

- **WordNet-Affect lexicon:** marked with Ekman's emotions.

happiness	sadness	anger	disgust	surprise	fear	total
398	201	252	53	71	141	1116

Distribution of labels in the WordNet-Affect lexicon.

- **Prior-Polarity lexicon:** over 8000 positive and negative words.
- **NRC Emotion lexicon:** a list of words for eight emotions (Ekman's six plus anticipation and trust).
- **Intensifier lexicon:** a list of 112 modifiers.

Lexicons

- **WordNet-Affect lexicon:** marked with Ekman's emotions.
- **Prior-Polarity lexicon:** over 8000 positive and negative words.

Neutral	Negative	Positive	Both
6.9%	59.7%	31.1%	0.3%

Distribution of labels in the Prior-Polarity lexicon.

- **NRC Emotion lexicon:** a list of words for eight emotions (Ekman's six plus anticipation and trust).
- **Intensifier lexicon:** a list of 112 modifiers.

Lexicons

- **WordNet-Affect lexicon:** marked with Ekman's emotions.
- **Prior-Polarity lexicon:** over 8000 positive and negative words.
- **NRC Emotion lexicon:** a list of words for eight emotions (Ekman's six plus anticipation and trust).

joy	sad-ness	anger	dis-gust	sur-prise	fear	antici-pation	trust
689	1191	1247	1058	534	1476	839	1231

Distribution of labels in the NRC Emotion lexicon.

- **Intensifier lexicon:** a list of 112 modifiers.

Lexicons

- **WordNet-Affect lexicon:** marked with Ekman's emotions.
- **Prior-Polarity lexicon:** over 8000 positive and negative words.
- **NRC Emotion lexicon:** a list of words for eight emotions (Ekman's six plus anticipation and trust).
- **Intensifier lexicon:** a list of 112 modifiers.
  - Two annotators assigned intensity coefficients to modifiers.
  - The coefficient values signal strengthening (0.0 .. 1.0) or weakening (1.0 .. 2.0).
  - The result was averaged.
  - Examples:
    - no, not, never 0.0
    - a bit 0.4
    - absolutely 1.3
    - deeply, significantly 2.0

Outline

Features

All feature counts are based on emotion words – from any of the lexicons – which occur in a given sentence.

We work with four sets of features:

- emotion word features;
- part-of-speech features;
- dependency-tree features;
- sentence features.

Emotion word features

- The emotion of a word according to WordNet-Affect.
- The polarity of a word according to the Prior-Polarity lexicon.
- The presence of a word on a short list of modifiers.

Emotion word features

Example

It was the best summer I have ever experienced.

- The emotion of a word according to WordNet-Affect. Happy
- The polarity of a word according to the Prior-Polarity lexicon. Positive
- The presence of a word on a short list of modifiers. False

Part-of-speech features

- The POS of the emotion word according to the emotion lexicon.
- The POS of the emotion word in the sentence according to the Stanford tagger.
- The POS of the words in a [-2, 2] window in the same sentence.

Part-of-speech features

Tagging example

It/PRP was/VBD the/DT best/JJS summer/NN I/PRP have/VBP ever/RB experienced/VBN ./.

- The POS of the emotion word according to the emotion lexicon. adj
- The POS of the emotion word in the sentence according to the Stanford tagger. JJS
- The POS of the words in a [-2, 2] window in the same sentence. [VBD, DT, __, NN, PRP]

Dependency-tree features (1)

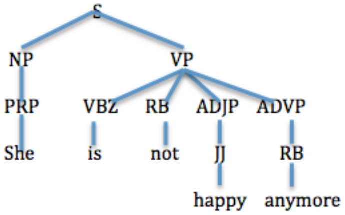
- Negation modifier: whether the word is in a “neg” dependency.
- Adjectival modifier: whether the word is in an “amod” dependency.
- Adverbial modifier: whether the word is in an “advmod” dependency.

Dependency-tree features (1)

- Negation modifier: whether the word is in a “neg” dependency. True
- Adjectival modifier: whether the word is in an “amod” dependency. False
- Adverbial modifier: whether the word is in an “advmod” dependency. True

“She is not happy anymore.” – typed dependencies:

- nsubj(happy-4, She-1)
- cop(happy-4, is-2)
- neg(happy-4, not-3)
- root(ROOT-0, happy-4)
- advmod(happy-4, anymore-5)



Dependency-tree features (2)

- Modifies a positive word
- Modifies a negative word
- Modified by a positive word
- Modified by a negative word

Dependency-tree features (2)

- Modified by a positive word. True
- Modified by a negative word. True

"The price was too good to be true." – typed dependencies:

- det(price-2, The-1)
- nsubj(good-5, price-2)
- cop(good-5, was-3)
- advmod(good-5, too-4)
- root(ROOT-0, good-5)
- aux(true-8, to-6)
- cop(true-8, be-7)
- xcomp(good-5, true-8)

Dependency-tree features (3)

- Modifies an intensifier-strengthen word
- Modifies an intensifier-weaken word
- Modified by an intensifier-strengthen word
- Modified by an intensifier-weaken word

Dependency-tree features (3)

- Modifies an intensifier-strengthen word.
- Modifies an intensifier-weaken word.
- Modified by an intensifier-strengthen word.
- Modified by an intensifier-weaken word. True

"The movie was barely horrifying." – typed dependencies:

- det(movie-2, the-1)
- nsubj(horrifying-5, movie-2)
- cop(horrifying-5, was-3)
- advmod(horrifying-5, barely-4)
- root(ROOT-0, horrifying-5)

Sentence features

Sentence features

- The number of words in a sentence.

Sentence features

Sentence features

- The number of words in a sentence.

Several other sentence-based features are only used in the third set of experiments.

- The number of emotion words in each emotion class according to the NRC Emotion lexicon.
- The number of polar words for each polarity class based on the Prior-Polarity lexicon.
- The number of intensifiers in the sentence, found in the intensifier lexicon.
- Whether the sentence – as proposed by Wiebe *et al.* 2005 –
  1. is in passive voice,
  2. contains an auxiliary verb,
  3. contains a copula.

Outline

Experiments

The first set of experiments

. . . on sentences with one emotion word.

The second set of experiments

. . . on sentences with more than one emotion word.

The third set of experiments

. . . on all emotion sentences.

Experiments

The first set of experiments

. . . on sentences with one emotion word.

. . . example: “I pride myself on listening to a wide variety of music ”, indicating *Happiness*

The second set of experiments

. . . on sentences with more than one emotion word.

. . . example: “I am losing enjoyment out of things I love to do and that’s never a good sign.”, indicating *Sadness*

The third set of experiments

. . . on all emotion sentences.

The first set of experiments (1)

- **The first baseline**: the emotion of the most frequent class, 31% accuracy.
- **The second baseline**: the emotion of the sentence is the prior emotion of the only emotion word in the sentence, 51% accuracy.

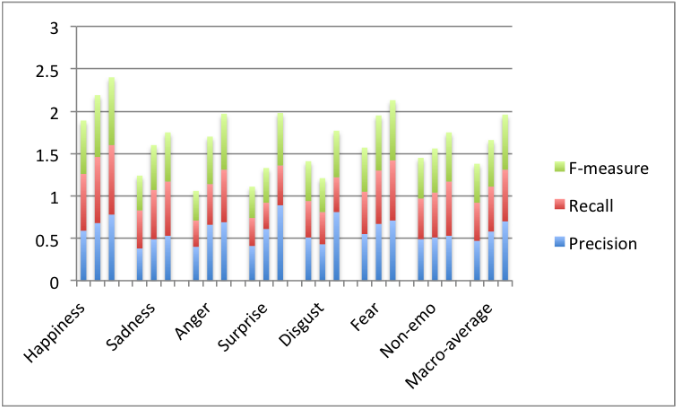
The first set of experiments (2)

Classification results

- SVM + Bag-of-Words: 50.72%
- SVM + our features: 58.88%
- Logistic regression (LR) + our features: 66.88%

We do not run LR with Bag-of-Words, because the process is very slow. LR models with large numbers of features and limited amount of training data are also highly prone to over-fitting.

The first set of experiments (3)



1. SVM + Bag-of-Words: 50.72%
2. SVM + our features: 58.88%
3. Logistic regression (LR) + our features: 66.88%

The first set of experiments (4)

Highlights

1. Logistic regression is a good choice of classifier for our representation method.
2. We note consistent improvement.
3. The results of both experiments using our set of features significantly outperform both the baselines and SVM applied to Bag-of-Words features.

The first set of experiments (5)

Feature set evaluation experiments

1. Prior polarity: is the emotion word positive or negative?
2. Lexical: all the features from our lexicons.
3. Part-of-speech.
4. Dependency: all the features defined as dependency-tree features, except the intensity and negation features.
5. Negation includes two negation features: is the prior polarity of the emotion word negative? and is the emotion word modified by a negative word?
6. Intensity groups all the features based on the intensifier lexicon.
7. Length is the length of the sentence in words.

The first set of experiments (6)

Feature set evaluation experiments

Features	Significant ($p < 0.001$)	Significant ($p < 0.01$)	Not significant	No change
Prior Polarity			54.42%	
Lexical	57.25%			
POS	60.25%			
Dependency		55.20%		
Negation		55.99%		
Intensity			53.86%	
Length				53.44%

T-test results of feature comparison, based on the accuracy of the classifier on each feature set.

The second set of experiments (1)

Classification steps

1. Learn the classification model from the sentences with one emotion word.
2. Using that model, get the probability distribution of emotion classes for each emotion word.
3. Combine the probability distribution of each emotion word by the unsupervised method.

The second set of experiments (2)

- **The first baseline**: the emotion of the most frequent class, 32% accuracy.
- **The second baseline**: the emotion of the sentence is the most frequent prior emotion of the emotion words in the sentence, 45% accuracy.

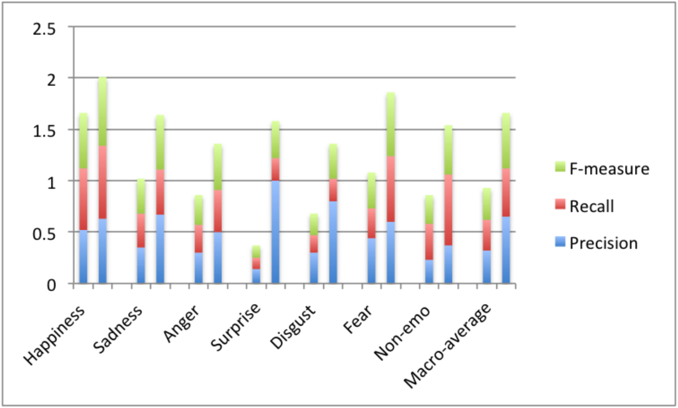
The second set of experiments (3)

Classification results

- SVM + Bag-of-Words: 36.71%
- LR + unsupervised + our features: 54.43%

The SVM system we used does not return a probability distribution, so we did not apply it to our features in this set of experiments. (Other SVM systems, with properly selected kernel functions, might work better.) Besides, the data set is probably too small for SVM to be truly effective.

The second set of experiments (4)



1. SVM + Bag-of-Words: 36.71%
2. LR + unsupervised + our features: 54.43%

The second set of experiments (5)

Highlights

1. Our method and our features significantly outperform the baselines and the SVM result applied to Bag-of-Words.
2. Examples of the contextual emotion differing from the prior emotion of the emotion word.
   - “You look like her I guess.”
   - “Joe said it does not happen that often so it does not bother him.”
3. We also found some common errors due to
   - limited coverage of the emotion lexicon,
   - complex sentences or unstructured sentences which cause the parser to fail or return incorrect data, resulting in incorrect dependency-tree information.

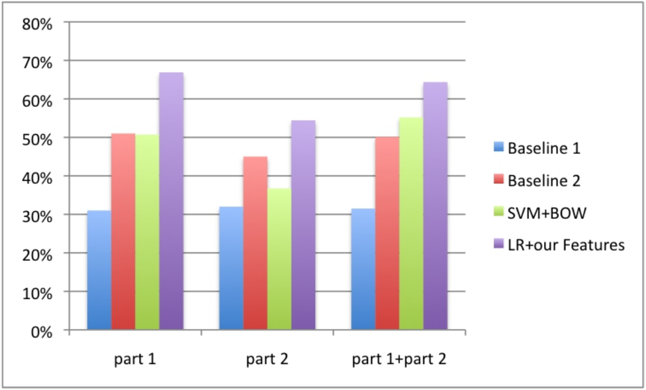
Combining the first two sets of experiments

The results on sentences with at least one emotion word

- The first baseline: 31.5%
- The second baseline: 50.13%
- SVM + Bag-of-Words: 55.17%
- LR + our features: 64.36%

  (we calculate the accuracy by counting the percentage of correctly classified instances in both parts of the dataset from the two preceding experiments)

Result on Aman's dataset



- Part 1: sentences with one emotion word.
- Part 2: sentences with more than one emotion word.

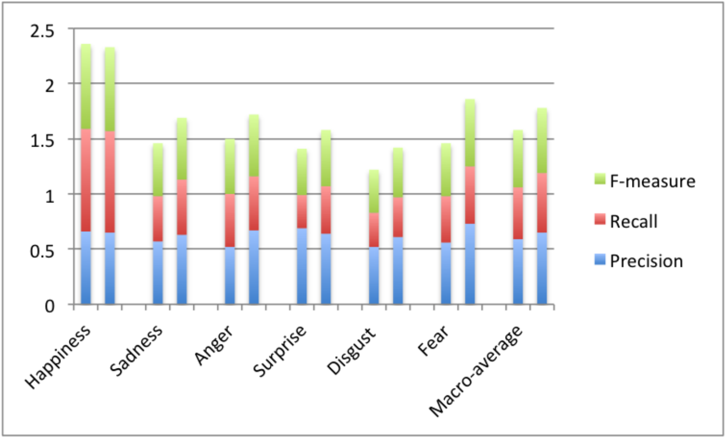
The third set of experiments (1)

- **Sentences with one emotion word**: we consider the only emotion word as a cue word (a word whose context drives the experiment).
- **Sentences with more than one emotion word**: we select as a cue word the emotion word which belongs to the fewest emotion classes. If there is more than one such word, we pick one at random.
- **Sentences with no emotion word**: we take as a cue word the root of the sentence in its dependency tree.

The third set of experiments (2)

- **The first baseline**: the emotion of the most frequent class, 41.5% accuracy.
- **The second baseline**: the emotion of the sentence is the most frequent prior emotion of the emotion words in the sentence, 51.39% accuracy.
- **The third baseline**: SVM applied to Bag-of-Words, 55.89% accuracy.
- **Classification results** on Aman's 1290 emotion sentences
  - SVM + Bag-of-Words: 55.89%
  - Logistic regression + the NRC Emotion lexicon : 61.63%
  - Logistic regression + the WordNet-Affect lexicon: 65.04%

The third set of experiments (3)



1. Logistic regression + the NRC Emotion lexicon : 61.63%
2. Logistic regression + the WordNet-Affect lexicon: 65.04%

The third set of experiments (4)

Highlights

1. Expected an improvement in the first experiment due to a larger lexicon, but the experiment using WordNet-Affect improves the result particularly in the negative emotion classes. (This could be due to the fact that each word can belong to multiple emotion classes in the NRC Emotion lexicon.)
2. A high percentage of sentences with no emotion words are non-emotion sentences – sentences such as “Good night!”, “Okay”, and “Yeah”.
3. Using the whole dataset including all non-emotion sentences, we saw a dramatic drop compared to only using the emotion sentences.

Outline

Conclusion

- We need to take the contextual emotion of a word into account.
- We need to take the syntactic structure of the sentence into account.
- Our method and our features significantly outperform both the baselines and the SVM result applied to Bag-of-Words.
- We grouped the features and found that most of them – lexical, POS, dependency and negation features – improve the results significantly.
- We used a second lexicon and we found that our features improve the results regardless of the lexicon we choose.

Future work

- Show the robustness of these features by applying them to different datasets.
- Use second-order (indirect) dependencies with the emotion word, for example, in the sentence "There is *no* mother who does *not* enjoy the success of her child."
- Evaluate the effect of our feature sets for each emotion class separately.
- Consider the specification of each emotion class and expand the dependency features based on each emotion class.

Outline

Bibliography I

Cecilia Ovesdotter Alm, Dan Roth, and Richard Sproat.
Emotions from Text: Machine Learning for Text-based Emotion Prediction.
In *HLT/EMNLP*, pages 347–354, 2005.

Saima Aman and Stan Szpakowicz.
Identifying Expressions of Emotion in Text.
In *Proc. 10th International Conf. Text, Speech and Dialogue*, pages 196–205. Springer-Verlag, 2007.

Paul Ekman.
An Argument for Basic Emotions.
Cognition & Emotion, 6(3):169–200, 1992.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten.
The WEKA data mining software: an update.
SIGKDD Explor. Newsl., 11:10–18, November 2009.

Phil Katz, Matthew Singleton, and Richard Wicentowski.
SWAT-MP: the SemEval-2007 systems for task 5 and task 14.
In *Proc. 4th International Workshop on Semantic Evaluations*, SemEval '07, pages 308–313, 2007.

Marie-Catherine De Marneffe, Bill Maccartney, and Christopher D. Manning.
Generating Typed Dependency Parses from Phrase Structure Parses.
In *Proc. LREC 2006*, pages 449–454, 2006.

Bibliography II

Saif M. Mohammad and Peter D. Turney.
Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon.
In *Proc. NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 26–34, 2010.

Alena Neviarouskaya, Helmut Prendinger, and Mitsuru Ishizuka.
AM: Textual Attitude Analysis Model.
In *Proc. NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 80–88, 2010.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan.
Thumbs up?: Sentiment Classification Using Machine Learning Techniques.
In *Proc. 2002 Conf. on Empirical Methods in Natural Language Processing*, EMNLP '02, pages 79–86, 2002.

Carlo Strapparava and Alessandro Valitutti.
WordNet-Affect: an Affective Extension of WordNet.
In *Proc. 4th International Conference on Language Resources and Evaluation*, pages 1083–1086, 2004.

Janyce Wiebe, Theresa Wilson, and Claire Cardie.
Annotating Expressions of Opinions and Emotions in Language.
Language Resources and Evaluation, 39(2-3):165–210, 2005.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann.
Recognizing Contextual Polarity: An Exploration of Features for Phrase-Level Sentiment Analysis.
Computational Linguistics, 35(3):399–433, 2009.