Biomedical Informatics Insights

ORIGINAL RESEARCH

Combining Lexico-semantic Features for Emotion Classification in Suicide Notes

Bart Desmet[1,2] and Véronique Hoste[1,2]

[1]University College Ghent, Ghent, Belgium. [2]Ghent University, Ghent, Belgium.
Corresponding author email: bart.desmet@hogent.be; veronique.hoste@hogent.be

Abstract: This paper describes a system for automatic emotion classification, developed for the 2011 i2b2 Natural Language Processing Challenge, Track 2. The objective of the shared task was to label suicide notes with 15 relevant emotions on the sentence level. Our system uses 15 SVM models (one for each emotion) using the combination of features that was found to perform best on a given emotion. Features included lemmas and trigram bag of words, and information from semantic resources such as WordNet, SentiWordNet and subjectivity clues. The best-performing system labeled 7 of the 15 emotions and achieved an F-score of 53.31% on the test data.

Keywords: emotion classification, topic classification, suicide, suicide notes, machine learning

Biomedical Informatics Insights 2012:5 (Suppl. 1) 125–128

doi: 10.4137/BII.S8960

This article is available from http://www.la-press.com.

Introduction

The second track of the 2011 i2b2 Natural Language Processing Challenge[1] is a shared task on emotion classification. Its aim is to automatically annotate suicide notes with a set of emotions. The data used for the challenge consisted of a training set of 600 suicide notes, and a test set of 300 notes.

The notes were annotated by three human annotators, who were asked to assign 15 relevant labels to each sentence in a note. As a result, each sentence could be annotated with none, one or more of the following labels: abuse, anger, blame, fear, forgiveness, guilt, happiness, hopefulness, hopelessness, information, instructions, love, pride, sorrow, thankfulness. If at least two annotators agreed on an annotation, it was retained. Inter-annotator agreement was measured with Krippendorff's α coefficient with Dice's coincidence index, and was 0.546 at the sentence level.[1]

On average, notes were 7.7 sentences long and contained 132.5 tokens (17.2 tokens per sentence) in the training set, and 7.0 sentences long with 121.5 tokens (17.5 tokens per sentence) in the test set. The distribution of the labels in both sets is presented in Table 1.

Methodology

The operational unit of the task is the sentence. A successful system would accurately predict for each sentence which, if any, emotions are present. Because the 15 emotion labels are not mutually exclusive, there are $15^2 = 225$ possible label combinations.

Table 1. Distribution of emotions in training and test set: average number of annotations per 1000 sentences.

Label	Training set	Test set
Instructions	177.0	183.1
Hopelessness	98.2	109.8
Love	63.9	96.4
Information	63.7	49.9
Guilt	44.9	56.1
Blame	23.1	21.6
Thankfulness	20.3	21.6
Anger	14.9	12.5
Sorrow	11.0	16.3
Hopefulness	10.1	18.2
Happiness	5.4	7.7
Fear	5.4	6.2
Pride	3.2	4.3
Abuse	1.9	2.4
Forgiveness	1.3	3.8

Note: Sorted by frequency in the training set.

We therefore decided to use 15 binary classifiers that each determined whether or not to assign a specific emotion, and combined their outputs.

Features

Shallow inspection of the data showed that most emotions were strongly lexicalized. We hypothesized that classifiers would perform adequately with a feature set that generalized lexical information and included subjectivity information from external resources.

The data was first preprocessed with the Memory-Based Shallow Parser (MBSP) for Python v1.4,[2] which provided lemmas and part-of-speech tags. The following features were extracted from the training data:

Lemmas—The set of lowercased lemmas present in the training corpus used as binary bag-of-words (BOW) features. The feature value for a given lemma is 1 if the lemma occurs once or more in a sentence, 0 otherwise. The preprocessed training data contained 4932 unique lowercased lemmas.

Lemmas + POS—The set of unique combinations of a lemma and a POS tag present in the training corpus, as binary BOW features (7447 features).

Pruned lemmas + POS—The lemma—POS pair BOW features, reduced to only include pairs where the POS tag is either verb, noun, adjective or adverb. By using only content words, the importance of function words for emotion classification could be gauged (6936 binary BOW features).

Trigrams—1742 binary BOW features representing trigrams from the training corpus. The trigrams were selected on the basis of how indicative they were of the presence of an emotion: only those trigrams were used that occurred proportionately at least ten times as often in the set of positive sentences for one or more emotions.

WordNet synsets—13146 binary BOW features, representing WordNet synsets in which the training data lemmas occurred. These features allow to generalize lexical clues present in the training data to all their synonyms.

SentiWordNet information—SentiWordNet is a lexical resource for opinion mining that assigns three sentiment scores to each synset of WordNet: positivity, negativity and objectivity.[3] Scores range from 0.0 to 1.0 in steps of 0.125. The features based on SentiWordNet information were: average positivity and negativity score (sum of scores divided by the number of synsets found in the sentence), the proportion of synsets with a score above

a given threshold (thresholds of 0.125 to 1.0 in steps of 0.125 for positivity and negativity), and the proportion of words in the weak positive or negative range (lower half from 0.125 to 0.5).

Subjectivity clues—We used a publicly available collection of subjectivity clues,[4] comprising 8221 word forms that are likely to occur in a subjective context. Each clue is categorized as subjective in most contexts (strongly subjective) or as only having some subjective usages (weakly subjective), and its prior polarity is marked (out of context, does the clue seem to evoke something positive or negative?). Features derived from the clues were: the proportion of lemma—POS pairs in a sentence present in the subjectivity clue collection, and the proportion of clues with strong positive, weak positive, weak negative and strong negative prior polarity, relative to the total amount of clues found in a sentence.

Classifier

All experiments were done with Support Vector Machine (SVM) classifiers. A standard SVM is a supervised learning classifier for binary classification. It learns from the training instances by mapping them to a high-dimensional feature space using a kernel function, and constructing a hyperplane along which they can be separated into the two classes, the decision boundary. Unseen instances are mapped to the feature space, and labeled depending on their position with respect to the decision boundary. The distance from the instance perpendicular to the hyperplane can be used as a measure of classification certainty.

SVM-Light[5] was used in our experiments, through the pysvmlight[a] Python binding. SVM-Light outputs a floating point number for unseen instances: its sign designates the position, its absolute value the distance relative to the decision boundary. Bootstrap resampling[6] was used to determine for each classifier which decision threshold maximized F-score.

Results and Discussion

We experimentally determined the best-performing combination of features for each of the 15 emotions. For the majority of the emotions, lemma and trigram bag of words proved to be indispensable features. For 6 emotions, these features alone yield the best results, while for another 7 emotions, classifiers achieve the best scores with the addition of subjectivity clues. Only 2 emotions benefit from WordNet and SentiWordNet information.

Data sparsity is a problem for some emotions. This has a direct influence on classifier performance, given that they use supervised learning and rely on positive examples of a class to learn from. All the best classifiers for emotions with an incidence of less than 20% (average number of annotations per 1000 sentences in the training data) have an F-score below 21.0, and all the best classifiers for emotions with an incidence above 20% score above 28.0. Emotions with an incidence of over 40% all score above 40.0, along with thankfulness, which proves easily learnable despite a low incidence of 20.3%. It is likely that classifier performance for the low-incidence emotions would rise considerably if more training data were obtained, without the need for new features.

In order to produce the final system output, the output each emotion's classifier is combined into one output file. Global system performance is calculated in terms of micro-averaged F-score, which is computed globally over all annotations, whereas macro-averaged F-scores would be computed over each emotion first, and then averaged over the 15 emotions.

Because micro-averaged F-score gives equal weight to each annotation, good performance on majority classes is important, because they have a larger number of annotations and therefore influence the global F-score more. Similarly, rare emotions, if predicted correctly, only bring a small positive contribution to overall F-score. However, if there is a lot of noise in the predictions due to high recall with low precision, minority classes can have a substantial negative influence on global F-score.

For this reason, we tried leaving out annotations of rare emotions, on which our classifiers performed poorly, and determined experimentally which pruned set of emotions yielded the best overall result on the training data set. This was achieved by leaving out emotions with an frequency of less than 2% (in the training set), resulting in output containing only 7 emotions: blame, guilt, hopelessness, information, instructions, love and thankfulness. The test data was then processed with classifiers trained on all the training data, using the appropriate feature set and threshold per emotion. Two versions of its output

[a]https://bitbucket.org/wcauchois/pysvmlight.

Table 2. Micro-averaged F-scores on the training and test set for all emotions, and the 7 best-performing emotions (pruned).

	Training	Test
All emotions	49.11	51.19
Pruned emotions	51.04	53.31

were submitted: one containing all emotions, the other containing the same pruned set of emotions.

Table 2 presents the overall F-scores for all emotions and for the best-performing pruned set of emotions, both on the training data and on the test data. Pruning the output resulted in an increase in micro-averaged F-score of 1.93 percentage points on the training data, and 2.12 percentage points on the test data.

The scores on the test data are 2.08 and 2.27 percentage points higher than the scores on the training data, for all emotions and pruned emotions, respectively. These increases indicate that there was no overfitting problem with the classifiers.

Conclusions and Future Work

This paper described experiments with lexico-semantic features for emotion classification in suicide notes. The results suggested that such features perform well, but suffer from data sparseness. This could be remedied by collecting more training examples for rare emotions.

An alley for future work would be to investigate the effect of applying spelling correction as a pre-processing step. Given the amount of spelling errors in the data, and the dependence of our classifiers on lexical features such as lemmas and trigrams, data sparsity could be significantly reduced by correcting spelling mistakes.

Deeper semantic analysis of suicide notes could also yield informative features for emotion classification. Furthermore, classifiers might benefit from features that model negation and modality. Simple bag of word features alone do not take into account such modification that may flip the meaning of significant word sequences.

Disclosures

Author(s) have provided signed confirmations to the publisher of their compliance with all applicable legal and ethical obligations in respect to declaration of conflicts of interest, funding, authorship and contributorship, and compliance with ethical requirements in respect to treatment of human and animal test subjects. If this article contains identifiable human subject(s) author(s) were required to supply signed patient consent prior to publication. Author(s) have confirmed that the published article is unique and not under consideration nor published by any other publication and that they have consent to reproduce any copyrighted material. The peer reviewers declared no conflicts of interest.

References

1. Pestian J, Matykiewicz P, Linn-Gust M, et al. Sentiment analysis of suicide notes: A shared task. *Biomedical Informatics Insights*. 2012; 5 (Suppl. 1):3–16.
2. Daelemans W, van den Bosch A. *Memory-based Language Processing*. Cambridge University Press; 2005.
3. Baccianella S, Esuli A, Sebastiani F. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: *Proceedings of LREC*. 2010.
4. Wilson T, Wiebe J, Hoffmann P. Recognizing contextual polarity in phrase-level sentiment analysis. In: *Proceedings of HLT/EMNLP*. 2005.
5. Joachims T. *Advances in Kernel Methods—Support Vector Learning*, chapter Making large-Scale SVM Learning Practical. MIT-Press; 1999.
6. Noreen EW. *Computer Intensive Methods for Testing Hypotheses: An Introduction*. John Wiley & Sons, New York; 1989.