Expert Systems with Applications 40 (2013) 6351–6358

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Emotion detection in suicide notes

Bart Desmet [a,b,*], Véronique Hoste [a,c]

[a] LT3 Language and Translation Technology Team, University College Ghent, Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
[b] Department of Applied Mathematics and Computer Science, Ghent University, Krijgslaan 281 (S9), 9000 Ghent, Belgium
[c] Department of Linguistics, Ghent University, Blandijnberg 2, 9000 Ghent, Belgium

ARTICLE INFO

Keywords:
Natural language processing
Suicide
Emotion

ABSTRACT

The success of suicide prevention, a major public health concern worldwide, hinges on adequate suicide risk assessment. Online platforms are increasingly used for expressing suicidal thoughts, but manual monitoring is unfeasible given the information overload experts are confronted with. We investigate whether the recent advances in natural language processing, and more specifically in sentiment mining, can be used to accurately pinpoint 15 different emotions, which might be indicative of suicidal behavior.

A system for automatic emotion detection was built using binary support vector machine classifiers. We hypothesized that lexical and semantic features could be an adequate way to represent the data, as emotions seemed to be lexicalized consistently. The optimal feature combination for each of the different emotions was determined using bootstrap resampling. Spelling correction was applied to the input data, in order to reduce lexical variation.

Classification performance varied between emotions, with scores up to 68.86% F-score. F-scores above 40% were achieved for six of the seven most frequent emotions: thankfulness, guilt, love, information, hopelessness and instructions. The most salient features are trigram and lemma bags-of-words and subjectivity clues. Spelling correction had a slightly positive effect on classification performance.

We showed that fine-grained automatic emotion detection benefits from classifier optimization and a combined lexico-semantic feature representation. The modest performance improvements obtained through spelling correction might indicate the robustness of the system to noisy input text. We conclude that natural language processing techniques have future application potential for suicide prevention.

1. Introduction

Suicide, which can be defined as death caused by self-directed injurious behavior with any intent to die, is a major public health concern and a leading cause of death worldwide, with an estimated 1 million victims per year Värnik, 2012. In the United States, suicide is the third leading cause of death among 1- to 24-year-olds Miniño and Murphy, 2012, an age bracket that actively uses new technologies. Young people create online content on blogs, social networking and video-sharing sites. However, these communication channels may be used for cyberbullying (sometimes leading to suicide risk), forming suicide pacts, and accessing or producing suicide-related content (e.g. methods).

Suicide prevention efforts focus on early risk recognition and referral to appropriate support. However, the information overload generated in social media applications has rendered it unfeasible and undesirable for interest groups (such as website administrators, health professionals and suicide prevention centers) to manually and continuously monitor them. There is a need for automatic procedures that can spot suicidal messages and allow stakeholders to detect and quickly react to online suicidal behavior or incitement. The effectiveness of such short interventions has been proven (Christensen, Griffiths, & Jorm, 2004; Christensen, Griffiths, & Korten, 2002). Furthermore, technological solutions are an important means to assuring privacy and protection of personally identifiable information (Weiss, 2008).

Consequently, there is growing interest in the application of natural language processing (NLP) and machine learning techniques for detecting emotions indicative of suicidal behavior. Because text corpora with evidence of suicidal intent are scarce, research has been focused on suicide notes. They can be considered an "occluded genre" (Shapero, 2011), since people producing the notes rarely have access to example texts to guide them. Modeling the emotions present in such notes may help health professionals in assessing suicide risk, by comparing the model to texts written by at-risk subjects, such as psychiatric patients or online content producers.

In this paper, we investigate whether the recent advances in machine learning and natural language processing (NLP), and more

* Corresponding author at: LT3 Language and Translation Technology Team, University College Ghent, Groot-Brittanniëlaan 45, 9000 Ghent, Belgium. Tel.: +32 9 224 97 53.
E-mail addresses: bart.desmet@hogent.be (B. Desmet), veronique.hoste@hogent.be (V. Hoste).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.05.050

specifically sentiment mining, can be used to automatically detect emotions in suicide notes. To this end, we seek to combine shallow lexical text characteristics with semantic information and other features which have proven their effectiveness in other domains.

2. Theory

The linguistic analysis of suicidal text has a long history, starting as early as 1957 (Clues to Suicide., 1957; Edelman & Renshaw, 1982; Gleser, Gottschalk, & Springer, 1961; Osgood & Walker, 1959). Most of this research was based on a corpus of 66 suicide notes collected by Shneidman, half genuine and half simulated, and the task was to identify features to differentiate between genuine and fake notes. Early work was mostly focused on manual analysis and detection of such features, e.g. by relying on techniques from discourse analysis (Clues to Suicide, 1957) or by focusing on shallow text characteristics like the usage of modals and auxiliaries (Osgood & Walker, 1959), the choice of verbs and adverbs (Gleser et al., 1961), etc. We observe a recent tendency to also rely on automatic corpus analysis techniques for the detection of suicidal messages. In Shapero (2011), two corpora of suicide notes are investigated to define the typical suicide note, by quantifying word usage and semantic concepts. As far as we know, (Pestian, Nasrallah, Matykiewicz, Bennett, & Leenaars, 2010) were the first to experiment with machine learning techniques for automatic suicide note classification. They showed that automatic systems could outperform mental health professionals in separating genuine from fake notes.

The application of flagging fake notes aside, it is important to determine what exactly makes a note a real suicide note, independent of the features of elicited notes or the distinguishing characteristics between both types of notes. A note corpus of positive-only data, annotated with fine-grained emotions, was released in the framework of the 2011 i2b2 NLP Challenge on emotion classification in suicide notes (Pestian et al., 2012), allowing research on which emotions might be indicative of suicidal behavior, and how they can be found automatically.

For automatic emotion detection and classification, we can rely on the recent advances in NLP (Jurafsky & Martin, 2009) and machine learning (Mitchell, 1997). Although the NLP research community has long focused on the "factual" aspects of content analysis, the mass of information currently produced on social media has also fueled interest in the automatic analysis of emotions and opinions expressed in this content. An NLP system which can extract both factual information and opinions, sentiments and beliefs from text creates a wealth of opportunities for organizations and individuals: in the benchmarking of products (Pang & Lee, 2008), in advising individuals in their purchases (Dabrowski et al. 2010), for prediction or measurement in the stock markets (Bollen et al., 2010) or in politics (Tumasjan et al., 2010), but also for the detection of emotions in suicide notes.

Given the large amount of research done within the field in the past few years, the terminology employed to define the different tasks and concepts dealing with subjectivity is not yet uniform across the research community. Until now, most work has concentrated on discovering whether a specific *object* (person, product, organization, event, etc.) is regarded in a positive or negative manner by a specific *source* (i.e. a person, an organization, a community, people in general, etc.). This task has been given many names, from opinion mining, to sentiment analysis, review mining, attitude analysis, appraisal extraction and many others.

A variety of techniques have been suggested to solve the task, ranging from simple lexicon-based systems, which decide on the polarity of a given text unit by counting the number of positive and negative words, to supervised machine learning techniques that rely on annotated corpora. Examples of supervised machine learning approaches are naive Bayes, support vector machines, memory-based learning and Hidden Markov Models (e.g. Banea et al., 2008; Pak & Paroubek, 2010; Wilson, Wiebe, & Hoffmann, 2005; Rentoumi et al., 2010).

In order to automatically extract and classify opinions or emotions at the document, sentence and clause level, different types of information sources (features) can be used. It is generally assumed that the semantic properties of individual words are good predictors of the semantic characteristics of the phrase or text that contain them, which is why a lot of effort has been devoted to the manual, semi-automatic, and automatic development of lists of words indicative of sentiment (Grefenstette, Qu, Evans, & Shanahan, 2006; Hatzivassiloglou & McKeown, 1997; Turney & Littman, 2003; Wiebe, 2000; Wiebe, Bruce, & O'Hara, 1999). Lexicon-based sentiment mining systems exploit the lexical information present

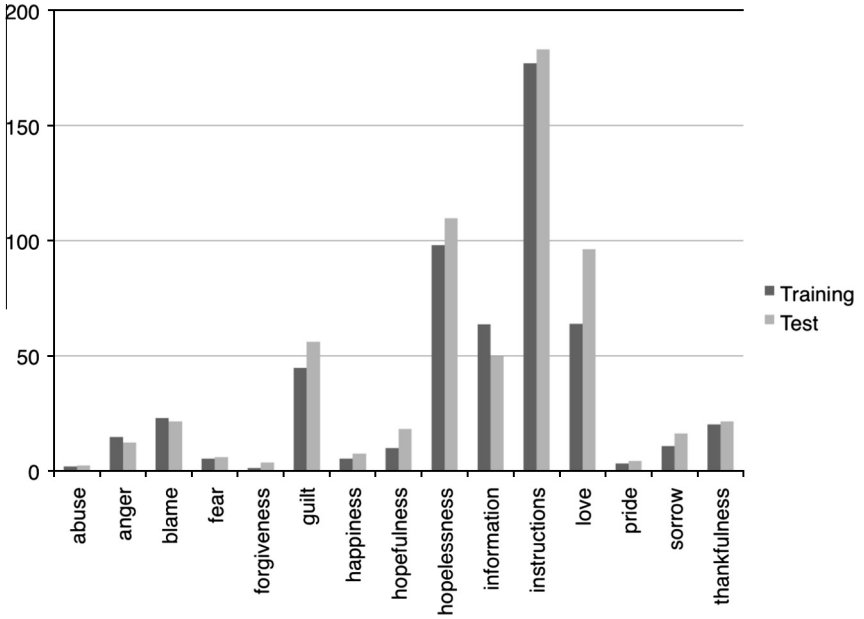

Fig. 1. Distribution of labels in training and test set: average number of annotations per 1000 sentences.

in the data (e.g. unigrams, bigrams, etc.) and external sentiment lexicons to determine for a given text, sentence or phrase whether it is positive, negative or neutral (e.g. Bermingham and Smeaton, 2010; Pak and Paroubek, 2010 and Agarwal et al. 2011). One might expect that a suicide note contains words relating to suicide. For example, dictionaries of suicide-related keywords have been used successfully for the detection of suicidal blogs (Huang, Goh, & Liew, 2007), although many false positives were reported (e.g. *Has anyone close to you committed suicide?* which cannot directly be considered as content from a suicidal person). The expression of sentiment might not only reside in single words, but also in combinations of words (e.g. *I would never kill myself* versus *I will kill myself* or the verb *hang* in *hang oneself* versus *to hang Christmas lights*). In order to capture negations, different word senses, etc., specific NLP components can provide other relevant features, such as named entity recognizers for object identification, synonymy detection systems (e.g. Agarwal & Bhattacharyya, 2005; Zhai et al. 2011), parsers (e.g. Nakagawa, Inui & Kurohashi, 2010; Di Caro & Grella, 2012), word sense disambiguation systems (e.g. Baccianella, Esuli, & Sebastiani, 2010), coreference resolvers for topic identification (e.g. Stoyanov & Cardie, 2008; Jakob & Gurevych, 2010), systems for negation scope detection, etc.

The analysis itself can be performed at different levels. Coarse-grained opinion mining at the document level aims at determining the overall sentiment properties of a text by classifying it as positive, negative or neutral, and has been widely studied (e.g. by Dave, Lawrence, & Pennock, 2003; Pang & Lee, 2004, 2008; Turney, 2002; Whitelaw, Garg, & Argamon, 2005). Fine-grained sentiment analysis is concerned with sentences and clauses (e.g. Tackstrom & McDonald, 2011; Ding et al., 2008; Popescu & Etzioni, 2005). It may identify the targets of opinions at the entity and aspect level (Hu & Liu, 2004; Qiu et al., 2011), focus on automatic classification of the intensity of opinions (Wilson et al., 2005) or on the fine-grained distinction between different types of emotions. This line of research was for example investigated by (Strapparava & Mihalcea 2007) or (Bellegarda, 2010), who distinguished between the six universal emotions as proposed by (Ekman, 1993): anger, disgust, fear, sadness, joy and surprise. Both the low inter annotator agreement scores and the large gap between the manual annotations and the reported system results show that there is a lot of room for improvement in the domain of emotion detection.

Our contribution can be situated along this line of fine-grained sentiment detection research. We aim at the fine-grained detection of a larger set of emotions (15 classes in total) in suicide notes. The emotions are detected at the sentence level. In order to do so, we used a supervised machine learning approach and trained a support vector machine approach on a manually labeled suicide note corpus, which exploits both lexical, structural and semantic information to differentiate between emotions.

3. Materials and methods

3.1. Dataset

We used a dataset released in the framework of a shared task on emotion classification in suicide notes, organized for the 2011 i2b2 NLP Challenge (Pestian et al., 2012), which aimed to create a permanently available resource facilitating future research. The corpus contains suicide notes written by 1319 people, collected between 1950 and 2011. Spelling and grammar errors were retained, and all names, addresses, telephone numbers and dates were anonymized to canonical forms. The challenge dataset consists of 900 notes, 600 of which are intended for developing and tuning the system (the training set). The remaining 300 notes were used for testing and comparing the performance of the participating systems (test set), and were released after the challenge ended.

Each note was annotated by at least three annotators who were asked to evaluate all sentences. A sentence could be annotated with none, one or more of the labels listed in Fig. 1. A single gold standard was created by retaining annotations on which at least two out of three annotators agreed. Inter-annotator agreement, measured with Krippendorff's α with Dice's coincidence index was 0.546 at the sentence level (Pestian et al., 2012).

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

It is apparent that some annotations are much more frequent than others. The most frequent labels are, in order, *instructions, hopelessness, love, information* and *guilt*. It is remarkable that *instructions* and *information* were annotated for this task, and with high frequency. Although technically they are not emotions, their presence and distribution in a suicide note may provide clues for further analysis (Shapero, 2011).

3.2. Data representation

Inspection of the training data showed that most emotions were strongly lexicalized, i.e. some words are typically associated with them. Table 1 lists example tokens for each emotion. It is notable that function words can also be indicators of emotions (e.g. *yours* for love and *without* for hopelessness). This is consistent with findings in stylometry research, where function words play a crucial role for e.g. authorship attribution Argamon and Levitan (2005).

We hypothesized that an emotion classification system would perform adequately with a feature set that generalized lexical information and included subjectivity information from external resources.

We adopted a supervised machine learning methodology, in which we trained *classifiers* to predict the presence of one or more emotions in unseen sentences.

The data was first preprocessed with the MBSP Memory-Based Shallow Parser for Python v1.4 (Daelemans, 2005), which provided lemmas and part-of-speech (POS) tags. We defined features as described below.

(1) Lemmas – To limit variation in word forms, words are reduced to a lowercased, uninflected head form, the lemma (e.g. *produced/producer/Producing → produce*). The set of lemmas in the training corpus was used for binary bag-of-words (BOW) features: a feature value is 1 if the corresponding

Table 1
Tokens typically associated with specific emotions.

Emotion	Frequent tokens
Abuse	*Awful, unbearable, curse*
Anger	*Sick, worst, nothing, dirty*
Blame	*Blame, fault, lie, cheat, bad*
Fear	*Afraid, fear, scared*
Forgiveness	*Forgive*
Guilt	*Sorry, forgive, cause*
Happiness	*Beautiful, happy, peaceful, stop*
Hopefulness	*Peace, worth, life*
Hopelessness	*Tired, hopeless, suffer, burden, illness, hell*
Love	*Love, yours, forever, darling*
Information	*Insurance, owe, savings, find*
Hopelessness	*Stand, bear, without, alone, hate*
Instructions	*Notify, please, tell, funeral*
Pride	*Proud, pride*
Sorrow	*Sorry, nobody, lonely*
Thankfulness	*Thank, good, kind*

lemma occurs once or more in a sentence. If the lemma does not occur, its value will be 0. The preprocessed training data contains 4932 unique lemmas.

(2) Lemmas + POS tags – The meaning of a lemma may depend on its part-of-speech tag (e.g. *produce* can be the lemma of either a noun or a verb). This ambiguity can be eliminated using a combination of lemma and POS tag. There are 7447 unique lemma-POS pairs in the corpus.

(3) Pruned lemmas + POS tags – A reduced set of lemma-POS pairs, containing the 6936 pairs with content words (the POS tag is either verb, noun, adjective or adverb). This allows to gauge the importance of function words for emotion classification: if removing function words hurts performance, they can be considered helpful.

(4) Trigrams – Because lemmas do not capture text sequences, combinations of three consecutive lemmas (trigrams) can be used. Trigrams were selected based on how indicative they were of the presence of an emotion: they should occur at least ten times as often in the set of positive sentences than the set of negative sentences. This yielded 1742 trigrams (e.g. *give them to*, *there be nothing*).

(5) WordNet synsets – WordNet (WordNet: An electronic lexical database., 1998) is a lexical database that groups English words into sets of synonyms (synsets), which allow to generalize lemmas. The lemmas in the training data occurred in 13146 synsets, used as BOW features.

(6) SentiWordNet information – SentiWordNet (Baccianella et al., 2010), a resource for opinion mining, assigns three sentiment scores between 0 and 1 to each WordNet synset: positivity, negativity and objectivity. We weigh how positive or negative the words in a sentence are, using the following features: average positivity and negativity score (sum of scores divided by the number of synsets in the sentence), proportion of synsets with a score above a threshold (thresholds of 0.125–1.0 in steps of 0.125), and proportion of words in the weak (0.125 to 0.5) positive or negative range.

(7) Subjectivity clues – We used a publicly available collection of subjectivity clues[24], comprising 8221 word forms likely to occur in a subjective context. Each clue is categorized as subjective in some or most contexts (weakly or strongly subjective), and out-of-context polarity. Features are the proportion of lemma-POS pairs in a sentence present in the collection, and the proportion of clues with weak/strong positive/negative prior polarity.

An ideal feature vector has highly informative features that lead the classifier to optimal performance. To determine the optimal feature combination, the 7 feature groups were combined into 17 feature sets, as described in Table 2.

3.3. Spelling correction

Some suicide notes contain many spelling errors. Most NLP modules, such as the MBSP shallow parser we used, are trained on large corpora (e.g. the Wall Street Journal corpus or the British National Corpus). However, when applying these NLP modules on data with many spelling mistakes, they fail to cope with the intense surface variation and consequently fall short in this early stage of analysis. Errors in the preprocessing step negatively affect the performance of downstream systems (error percolation). Similar observations have been made for the processing of social media data (Liu, 2010).

Given the amount of spelling errors in the data, and the dependence of our classifiers on lexical features, we hypothesized that by correcting spelling mistakes, we could reduce data sparsity and improve lexical recall.

We therefore ran spelling correction on all texts, using TICCL (Reynaert, 2010), a corpus-based spelling correction system. All experiments described below were performed on both the original and the spellchecked datasets.

An example of the spelling correction is given below, with errors underlined.

Original fragment: *This do n't blame eny baddy for whate mite happe to me for I know that I am heare in the way and am surfing all the time and I have ben to sovle doctors and thay say that thay have don all that thay can do for me so I am surffing all the time very bad and I dont know eny auther way out if it.*

Fragment spellchecked with TICCL: *This do n't blame any bad for what mite happen to me for I know that I am heard in the way and am surfing all the time and I have ben to move doctors and they say that they have don all that they can do for me so I am surfing all the time very bad and I do not know any further way out if it.*

The original fragment contains 18 errors. Five of those are not corrected (*mite*, *surfing*, *ben*, *don* and *if*), eight are corrected (*eny*, *whate*, *happe*, *thay* and *dont*), and five are changed incorrectly (*baddy*, *heare*, *sovle*, *surffing* and *auther*). No correct words were unnecessarily changed (hypercorrections). The fragment is annotated with the *hopelessness* emotion. It is possible that this emotion will be found easier, because the lexical variety is reduced in the spellchecked version. One obvious problem with the TICLL spellchecking module, however, is that it only considers words in isolation. Taking into account context, and merging or splitting consecutive tokens, would be beneficial. The following note fragment may serve as an example: *You have been good wife an suffer a nof. I wish all luck in the word because your deserved. I suffer tu much. God will for give me.* Here, potentially relevant lemmas such as *enough* and *forgive* would be missed: after spellchecking, they are *a now* and *for give*, respectively.

4. Fine-grained sentiment classification

4.1. Classifier

A successful system would accurately predict for each sentence which emotions it contains (if any). The 15 emotion labels are not mutually exclusive, so there are $15^2 = 225$ possible combinations. We therefore decided to use 15 binary classifiers that assign a specific emotion or not, and combine their outputs. We used Support

Table 2
The 17 experimental feature sets, named A–Q on the horizontal axis. The presence of a feature in a feature set is indicated with an x (e.g. feature set F contains lemma and trigram bag-of-words features).

Feature set	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
Lemmas	x					x	x	x	x		x	x	x	x	x	x	x
Lemmas + POS		x															
Pruned lemmas + POS			x														
Trigrams				x		x				x	x				x	x	x
WordNet synsets					x		x					x	x		x		x
SentiWordNet								x				x		x		x	x
Subjectivity clues									x	x	x		x	x	x	x	x

Vector Machines (SVMs), which have been shown to work well in other NLP tasks, such as word sense disambiguation and sentiment analysis.

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 the instances can be separated into two classes, the decision boundary. Unseen instances are mapped to the feature space, and labeled depending on their position with respect to this decision boundary. The distance from the instance perpendicular to the hyperplane can be used as a measure of classification certainty. SVM-Light (Joachims, 1999) was used, through the pysvmlight Python binding. SVM-Light outputs a floating-point number for unseen instances: its sign designates the position, its value the distance relative to the decision boundary.

The results of a classifier are reported as F_1-score, which is the weighted average of precision and recall. Precision measures the amount of false positives (of all the sentences that the classifier labeled with a specific emotion, how many did effectively have this emotion in the manual annotations?), recall measures the amount of false negatives (of all the sentences with a specific emotion in the annotations, how many were found by the classifier?). F_1-score, also known as balanced F-score, is a measure that places equal emphasis on both optimization objectives.

For some applications, optimizing more towards precision or recall might be appropriate. In the case of finding suicide-related content on the web, for example, some precision might be traded up in favour of recall, because a false negative (missing a suicidal message) is worse than a false positive. In such cases, a non-balanced F-score is used.

4.2. Bootstrap resampling

Although machine learning algorithms tend to come with heuristic default parameter settings, these may not be the optimal settings for our task. Since parameter optimization has been shown to lead to dramatic performance improvements (Daelemans, Hoste, De Meulder, & Naudts, 2003), we experimented with various decision boundaries, other than the ones proposed by the SVM classifiers.

Classification can be influenced by moving the decision boundary further away from the positive instances (improving recall, sacrificing precision), or by moving it closer. This can be done by defining a classification threshold other than 0. With a classification threshold of −2.0, for example, instances with an output in the range [0, −2.0] that were previously classified as negative, would now be classified as positive, thus improving recall at the expense of precision.

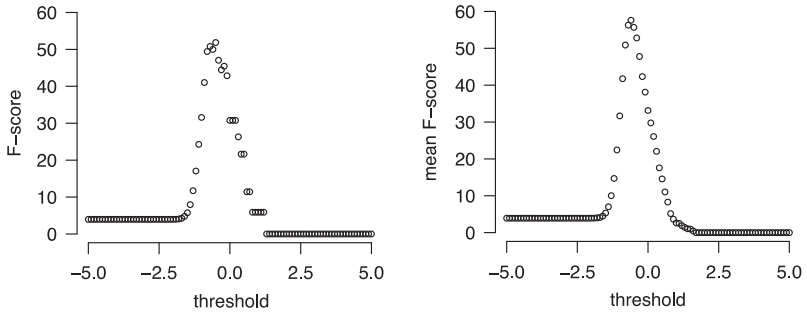

Fig. 2. F-score per threshold value for 1 sample, and average F-score over 50 samples (thankfulness, feature set 1, optimal threshold: −0.6).

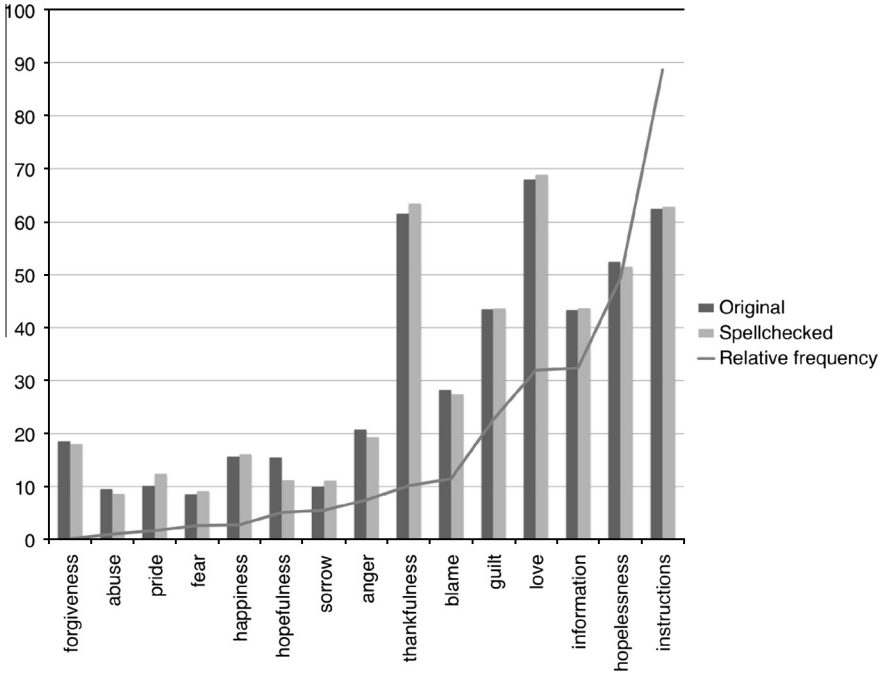

Fig. 3. F-scores and relative frequency for each emotion. F-scores are given in percent, frequency is given as average number of annotations per 500 sentences.

Table 3
The three best-performing feature sets per emotion (1 = best, 2 = second best, 3 = third best), on the spellchecked data set. The emotions are ordered according to their performance. The best-performing feature set per emotion is bold-faced.

Emotion	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
Love						**1**					3					2	
Thankfulness					3					2					**1**		
Instructions						**1**					2					3	
Hopelessness						2					**1**					3	
Information						**1**					2					3	
Guilt						2					**1**					3	
Blame				2		3				**1**							
Anger				2						**1**	3						
Forgiveness			3	2						**1**							
Happiness				3						2	**1**						
Pride				3						**1**				2			
Hopefulness		**1**		2		3											
Sorrow						2				**1**	3						
Fear					**1**					2			3				
Abuse						3				**1**						2	

Table 4
Micro-averaged F-scores on the training and test set for all emotions, and the 7 best-performing emotions (pruned). Results in the left half are on the original data set, in the right half on the spellchecked data set.

	Original		Spellchecked	
	Training	Test	Training	Test
All emotions	49.11	51.19	49.12	53.31
Pruned emotions	51.04	53.31	51.17	53.87

Bootstrap resampling (Noreen, 1989) was used to determine which threshold maximized F-score for each classifier. In every sampling round, 3000 sentences were randomly selected for training, and testing was done on the remaining 1633 sentences. F-score was calculated for thresholds ranging from −5.0 to 5.0 in steps of 0.1. After 50 bootstrapping rounds, we calculated average F-scores per threshold value, and selected the threshold with the highest average score.

By using bootstrap resampling, the choice for a particular threshold is more robust, because the range of average F-scores over thresholds is smoother, as is apparent in Fig. 2. As a result, the optimal threshold value can be determined more precisely.

5. Results and discussion

5.1. Results on individual emotions

We conducted bootstrap resampling experiments using SVM classifiers with 17 feature sets on the 15 emotions. Fig. 3 displays performance on the original and spellchecked dataset, with emotions ordered by frequency in the training data. Reported F-score is the average over 50 bootstrapping samples for the best-performing feature set.

Emotion frequency determines classifier performance considerably. For all emotions with a frequency of more than 40 annotations per 1000 sentences, F-scores are above 40%. Unsurprisingly, classifiers for rare emotions perform worst, because fewer training examples are available to learn from. Classifier performance would likely improve for the low-frequency emotions if more training data were obtained, without the need for new features.

Some emotions (*forgiveness, thankfulness* and *love*) perform better than would be expected from frequency alone, indicating that they are easier to learn than others – possibly because they are lexicalized more often, or more consistently.

The differences in F-score between the original and spellchecked datasets are small. Spelling correction is usually marginally beneficial, especially for the frequent emotions, although it hurts performance for *hopefulness*.

Table 3 presents the best-performing feature sets per emotion on the spellchecked dataset, ordered by F-score. The 15 best classifiers use only 6 feature sets: trigrams and subjectivity clues (set J, 6 classifiers), lemmas and trigrams (set F, 3 classifiers), lemmas, trigrams and subjectivity clues (set K, 3 classifiers), lemma-POS pairs (set B, 1 classifier), WordNet (set E, 1 classifier) and lemmas, trigrams, WordNet and subjectivity clues (set O, 1 classifier). Trigram bag-of-words and subjectivity clues are indispensable features for most emotions: they are used in 13 and 10 classifiers, respectively. For six emotions, these features alone yield the best results (set J). For six more emotions, combinations of trigrams with lemmas and subjectivity clues perform best (sets F and K), the classifier for *thankfulness* additionally uses WordNet (set O). Only two emotions do not use any of these features: the *hopefulness* classifier only uses lemma-POS pairs (set B), *fear* only uses WordNet (set E). None of the feature sets score well for these emotions, however. SentiWordNet information and pruned lemma-POS pairs are never used in a winning classifier.

It is notable that set A (lemma BOW features, 26.25% average F-score over all emotions) performed better for most emotions than set B (lemma-POS pair BOWs, 25.12% average F-score), which in turn performed better than the pruned variant in set C (24.52% average F-score). This may be explained by the fact that the lemma-POS pair bag-of-words introduces more sparseness. The worsening of the scores due to pruning (set C compared to set B) suggests that function words are useful for emotion classification.

5.2. Combining and pruning

The final system output is produced by aggregating the outputs of the best-performing classifier for each emotion. Global system performance is measured with micro-averaged F-score, which is computed globally over all annotations (whereas macro-average F-score would be computed over each emotion first, then averaged over the 15 emotions). Because micro-averaged F-score gives equal weight to each annotation, good performance on majority classes is important, as their larger number of annotations 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 low precision, minority classes can have a substantial negative influence on global F-score.

We therefore 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 dataset. This was achieved by retaining only seven emotions in the output: *blame, guilt, hopelessness, information, instructions, love* and *thankfulness*, which all have an incidence of more than 20 pro mille.

The test dataset was processed with classifiers trained on all the training data, using the appropriate feature set and threshold per emotion. Table 4 presents the overall F-scores on all emotions and the pruned set of emotions, both on the training and the test dataset. Results are reported with and without spellchecking.

Pruning the output resulted in an increase in micro-averaged F-score of 1.93 percentage points (spellchecked: 2.05) on the training data, and 2.12 percentage points (spellchecked: 0.56) on the test data. Pruning therefore proves to be worthwhile, especially on the original dataset.

It is noteworthy that all systems score better on the test set than on the training set, differing from 2.08 to 4.19 percentage points. This indicates that there was no problem of overfitting, which happens when a system is tailored too specifically to a training set, and does not generalize well on unseen test data.

Applying spelling correction consistently improves the overall results. Differences are small on the training set (0.01 for all emotions, 0.13 for pruned emotions), but the benefits are clearer on the test data, where improvements of 2.12 and 0.56 percentage points are obtained. The conservative impact of spelling correction could indicate that more advanced spelling correction techniques are required, or that the original system is robust to noise.

6. Conclusions and future work

This paper described a machine learning methodology for fine-grained emotion detection in suicide notes, using support vector machines. To differentiate between the 15 different emotions present in the suicide notes, we experimented with lexical and semantic features, viz. bags-of-words of lemmas, part-of-speech tags and trigrams, and information from external resources that encode semantic relatedness and subjectivity. The results suggest that such features perform well for frequent emotions, but suffer from data sparseness in rare emotions. The most salient features are trigram and lemma bags-of-words and subjectivity clues. The task of accurately predicting the 15 emotions is not a solved one, but given the clear correlation between performance and data availability, a promising alley would be to collect more training examples for the rare emotions.

Spelling correction was applied to improve lexical recall. This had some positive effect, although its importance may have been limited because the original system was robust to noise. It may be worthwhile to look into more advanced forms of spelling correction that use language models to correct unlikely sequences of correctly spelled words (e.g. *take ever thing*), tokenization errors (e.g. *for give*) and confusables (e.g. *heart and sole*).

Deeper semantic analysis could also yield informative features for emotion classification. For example, classifiers might benefit from features that model negation and modality (e.g. *I forgive you* vs. *I would never forgive you if...*). Simple bag-of-word features do not capture such modification, which may flip the meaning of significant sequences.

Emotion detection in suicide notes has direct applications in forensic linguistics and suicide prevention. Although the applicability of this research in medical practice is preliminary, a logical next step for future research would be to use the presence of emotions as features for a model that predicts suicidality in text, with interesting opportunities for further research on its use and effectiveness. In future work, we also intend to investigate the problem of detecting suicidal content for online suicide prevention, and to which extent emotion detection and its features can be used to its advantage.

Acknowledgements

The authors would like to thank Dr. John Pestian and his team for organizing the 2011 i2b2 shared task and making available the data. We are also grateful to Dr. Martin Reynaert for his advice and help on spelling correction. This work was carried out in the SubTLe project, funded by the University College Ghent Research Fund.

References

Agarwal, A., & Bhattacharyya, P. (2005). Sentiment Analysis : a new approach for effective use of linguistic knowledge and exploiting similarities in a set of documents to be classified. In *Proceedings of the International Conference on Natural Language Processing* (pp. 238-247).

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment Analysis of Twitter Data. In *Proceedings of the Workshop on Language in Social Media* (pp. 30-38).

Argamon, S., & Levitan, S. (2005). Measuring the usefulness of function words for authorship attribution. In *Proceedings of ACH/ALLC* (Vol. 5).

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the seventh conference on language resources and evaluation (LREC'10)* (Vol. 25, pp. 2200–2204). Valletta, MT.

Banea, C., & Mihalcea, R. (2008). Multilingual subjectivity analysis using machine translation. In *Proceedings of the 2008 conference on empirical methods in natural language processing* (pp. 127-135).

Bellegarda, J. R. (2010). Emotion analysis using latent affective folding and embedding. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text* (pp. 1-9).

Bermingham, A., & Smeaton, A. (2010). Classifying sentiment in microblogs: is brevity an advantage?. In *Proceedings of the 19th ACM international conference on information and knowledge management: ACM press*.

Bollen, J., Mao, H., & Zeng, X.-J. (2010). Twitter mood predicts the stock market. *Journal of Computational Science, 2*, 1–8.

Christensen, H., Griffiths, K. M., & Jorm, A. F. (2004). Delivering interventions for depression by using the internet: Randomised controlled trial. *BMJ, 328*, 265.

Christensen, H., Griffiths, K. M., & Korten, A. (2002). Web-based cognitive behavior therapy: Analysis of site usage and changes in depression and anxiety scores. *Journal of Medical Internet Research, 4*.

Clues to Suicide. (1957). McGraw-Hill Companies.

Dabrowski, M., Acton, T., Jarzebowski, P., & O'Riain, S. (2010). Improving customer decisions using product reviews. In *Proceedings of the 6th International Conference on Web Information Systems and Technologies* (Vol. 190).

Daelemans, W., Hoste, V. e., De Meulder, F., & Naudts, B. (2003). Combined optimization of feature selection and algorithm parameters in machine learning of language. *Machine Learning*, 84–95.

Daelemans, W., & van den Bosch, A. (2005). *Memory-based Language Processing*. Cambridge University Press.

Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on world wide web* (Vol. 17). ACM.

Di Caro, L., & Grella, M. (2012). Sentiment analysis via dependency parsing. Computer Standards & Interfaces.

Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the International Conference on Web Search and Web Data Mining* (pp. 231-240).

Edelman, A. M., & Renshaw, S. L. (1982). Genuine versus simulated suicide notes: An issue revisited through discourse analysis. *Suicide Life Threat Behavior, 12*, 103–113.

Ekman, P. (1993). Facial expression and emotion. *American Psychologist, 48*, 384–392.

Gleser, G. C., Gottschalk, L. A., & Springer, K. J. (1961). An anxiety scale applicable to verbal samples. *Archives of General Psychiatry, 5*, 593–605.

Grefenstette, G., Qu, Y., Evans, D. A., & Shanahan, J. G. (2006). Validating the coverage of lexical resources for affect analysis and automatically classifying new words along semantic axes. In *Computing attitude and affect in text: Theory and applications*.(pp. 93-107). Springer Netherlands.

Hatzivassiloglou, V., & McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the association for computational linguistics*.

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 168–177). ACM New York: NY, USA.

Huang, Y.-P., Goh, T., & Liew, C. L. (2007). Hunting suicide notes in web 2.0 – preliminary findings. *In Ninth IEEE international symposium on multimedia workshops (ISMW 2007)* (pp. 517–521). IEEE.

Jakob, N., & Gurevych, I. (2010). Using anaphora resolution to improve opinion target identification in movie reviews. In *Proceedings of the ACL 2010 conference short papers* (pp. 263-268). Association for Computational Linguistics.

Joachims, T. (1999). Making large-scale SVM learning practical. In *Advances in kernel methods – support vector learning*. MIT Press.

Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall.

Liu, B. (2010). Sentiment analysis and subjectivity. In *Handbook of natural language processing*. .

Miniño, A. M., & Murphy, S. L. (2012). Death in the United States, 2010. In NCHS data brief (pp. 1–8).

Nakagawa, T., Inui, K., & Kurohashi, S. (2010). Dependency tree-based sentiment classification using CRFs with hidden variables. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL* (pp. 786–794): Association for Computational Linguistics.

Mitchell, T. M. (1997). *Machine Learning*. Burr Ridge: McGraw-Hill.

Noreen, E. W. (1989). *Computer intensive methods for testing hypothesis: An introduction*. New York: John Wiley & Sons.

Osgood, C. E., & Walker, E. G. (1959). Motivation and language behavior: A content analysis of suicide notes. *Journal of Abnormal and Social Psychology, 59*, 58–67.

Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of LREC* (pp. 1320–1326). European Language Resources Association (ELRA).

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd annual meeting on association for computational linguistics*.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval, 2*, 1–135.

Pestian, J., Matykiewicz, P., Linn-Gust, M., South, B., Uzuner, O., Wiebe, J., et al. (2012). Sentiment analysis of suicide notes: A shared task. *Journal of Biomedical Informatics Insights, 5*, 3–16.

Pestian, J., Nasrallah, H., Matykiewicz, P., Bennett, A., & Leenaars, A. (2010). Suicide note classification using natural language processing: A content analysis. *Journal of Biomedical Informatics Insights, 3*, 19–28.

Popescu, A.-M., & Etzioni, O. (2007). Extracting product features and opinions from reviews. In *Natural language processing and text mining* (pp. 9–28). London: Springer.

Qiu, G., Liu, B., Bu, J., & Chen, C. (2011). Opinion word expansion and target extraction through double propagation. *Computational Linguistics, 37*, 9–27.

Rentoumi, V., Petrakis, S., Klenner, M., Vouros, G. A., & Karkaletsis, V. (2010). United we stand: improving sentiment analysis by joining machine learning and rule based methods. In Proceedings of the 7th conference on language resources and evaluation (LREC'10) (pp. 1089–1094).

Reynaert, M. W. C. (2010). Character confusion versus focus word-based correction of spelling and OCR variants in corpora. *International Journal on Document Analysis and Recognition, 14*, 173–187.

Shapero, J. J. (2011). *The language of suicide notes*. Diss. University of Birmingham.

Stoyanov, V., & Cardie, C. (2008). Topic identification for fine-grained opinion analysis. In *Proceedings of the 22nd international conference on computational linguistics* (pp. 817–824). Coling 2008 Organizing Committee.

Strapparava, C., & Mihalcea, R. (2007). SemEval-2007 Task 14: Affective Text. In *Proceedings of the 4th international workshop on the semantic evaluations (SemEval 2007)* (pp. 70–74). Prague: Czech Republic.

Täckström, O., & McDonald, R. (2011). Discovering fine-grained sentiment with latent variable structured prediction models. In *Advances in Information Retrieval* (pp. 368–374).

Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with Twitter : what 140 characters reveal about political sentiment. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media* (pp. 178–185).

Turney, P. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting of the association for computational linguistics* (pp. 417–424).

Turney, P., & Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems, 21*, 315–346.

Värnik, P. (2012). Suicide in the World. *International Journal of Environmental Research and Public Health, 9*, 760–771.

Weiss, S. (2008). The need for a paradigm shift in addressing privacy risks in social networking applications. In *The future of identity in the information society* (Vol. 262, pp. 161–171). Springer: Boston.

Whitelaw, C., Garg, N., & Argamon, S. (2005). Using appraisal groups for sentiment analysis. In *Proceedings of the 14th ACM international conference on information and knowledge management – CIKM '05* (pp. 625–631). ACM.

Wiebe, J. (2000). Learning Subjective Adjectives from Corpora. In *Proceedings of the national conference on artificial intelligence*. AAAI Press.

Wiebe, J. M., Bruce, R. F., & O'Hara, T. P. (1999). Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th annual meeting of the association for computational linguistics on computational linguistics* (pp. 246–253).

Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. In *Proceedings of the human language technology conference and the conference on empirical methods in natural language processing (HLT/EMNLP)* (pp. 347–354).

WordNet: An electronic lexical database. (1998). Cambridge, MA: MIT Press.

Zhai, Z., Liu, B., Xu, H., & Jia, P. (2011). Constrained LDA for grouping product features in opinion mining. In *Advances in knowledge discovery and data mining* (pp. 448-459). Berlin Heidelberg: Springer.