JCSE Journal of Computing Science and Engineering

Regular Paper

Journal of Computing Science and Engineering,
Vol. 6, No. 2, June 2012, pp. 143-150

Topic Classification for Suicidology

Jonathon Read*, Erik Velldal, and Lilja Øvrelid
Language Technology Group, Department of Informatics, University of Oslo, Norway
jread@ifi.uio.no, erikve@ifi.uio.no, liljao@ifi.uio.no

Abstract
Computational techniques for topic classification can support qualitative research by automatically applying labels in preparation for qualitative analyses. This paper presents an evaluation of supervised learning techniques applied to one such use case, namely, that of labeling emotions, instructions and information in suicide notes. We train a collection of one-versus-all binary support vector machine classifiers, using cost-sensitive learning to deal with class imbalance. The features investigated range from a simple bag-of-words and *n*-grams over stems, to information drawn from syntactic dependency analysis and WordNet synonym sets. The experimental results are complemented by an analysis of systematic errors in both the output of our system and the gold-standard annotations.

Category: Smart and intelligent computing

Keywords: Affect recognition; Sentiment analysis; Skewed class distribution; Text classification

I. INTRODUCTION

Suicide is a major cause of death worldwide, with an annual global mortality rate of 16 per 100,000, and the problem is growing at a rate that has been increasing by 60% in the last 45 years [1]. Researchers have recently called for more qualitative research in the fields of suicidology and suicide prevention [2]. Computational methods can expedite such analyses by labeling related texts with relevant topics.

The work described in this paper was conducted in the context of track 2 of the 2011 Medical Natural Language Processing (NLP) Challenge on sentiment analysis in suicide notes [3]. This was a multi-label non-exclusive sentence classification task, where labels were applied to the notes left by people who died from suicide. This paper presents an evaluation of the utility of various types of features for supervised training of support vector machine (SVM) classifiers to assign labels representing topics including several types of emotion and indications of information and instructions. The information sources explored range from bag-of-words features and *n*-grams over stems, to features based on syntactic dependency analysis and WordNet synonym sets. We also describe how cost-sensitive learning can be used to mitigate the effect of class imbalance.

We begin the remainder of the paper by providing some background on relevant work in Section II. We describe the data provided by the 2011 Medical NLP Challenge task organizers in Section III. Section IV details our approach, which involves training a collection of binary one-versus-all SVM sentence classifiers. Section V presents the performance of our approach, both under cross-validation of the development data and in final evaluation on held-out data. Section VI analyzes common types of errors, both in the gold-standard and the output produced by our system, while our conclusions and thoughts for future work are outlined in Section VII.

Open Access http://dx.doi.org/10.5626/JCSE.2012.6.2.143 http://jcse.kiise.org

Received February 18 2012, **Revised** May 07 2012, **Accepted** May 18 2012
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

II. RELATED WORK

We are not aware of any previous work on the automatic labeling of suicide notes. However, given the emphasis on emotion labels, the most similar previous work is perhaps the emotion labeling subtask of the SemEval-2007 affective text shared task [4], which involved scoring newswire headlines according to the strength of six so-called basic emotions stipulated by Ekman [5] - ANGER, DISGUST, FEAR, JOY, SADNESS and SURPRISE. There were three participating systems in the SemEval-2007 emotion labeling task. SWAT [6] employed an affective lexicon where the relevance of words to emotions was scored in an average emotion score for every headline in which they appear. UA [7] also used a lexicon, which was instead compiled by calculating the point-wise mutual information with headline words and an emotion using counts obtained through information retrieval queries. UPAR7 [8] employed heuristics over dependency graphs in conjunction with lexical resources such as WordNet-Affect [9]. In subsequent work, the task organizers investigated the application of latent semantic analysis (LSA) and a Naive Bayes (NB) classifier that was trained using author-labelled blog posts [10]. As can perhaps be expected given the different approaches of the various systems, each performed best for different emotions. This highlights the need for emotion-labeling systems to draw from a variety of analyses and resources.

III. DATA

The task organizers provided developmental data consisting of 600 suicide notes, comprising 4,241 (pre-segmented) sentences. Note that a "sentence" here is defined by the data, and can range from a single word or phrase to multiple sentences (in the case of segmentation errors). Each sentence is annotated with 0 to 15 labels (listed with their distribution in Table 1). For held-out evaluation, the organizers provided an additional set of 300 unlabelled notes, comprising 1,883 sentences. The task organizers report an inter-annotator agreement rate of 54.6% over all sentences. Fig. 1 provides excerpts from a note in the training data, with assigned labels.

Table 1. The distribution of labels in the training data

Label	Frequency	%
unlabelled sentences	2,460	58.01
INSTRUCTIONS	820	19.34
HOPELESSNESS	455	10.73
LOVE	296	6.98
INFORMATION	295	6.96
GUILT	208	4.91
BLAME	107	2.52
THANKFULNESS	94	2.22
ANGER	69	1.63
SORROW	51	1.20
HOPEFULNESS	47	1.11
HAPPINESS/PEACEFULNESS	25	0.59
FEAR	25	0.59
PRIDE	15	0.35
ABUSE	9	0.21
FORGIVENESS	6	0.14

IV. METHOD

Our approach to the task of labeling suicide notes involves learning a collection of binary *one-versus-all* classifiers. One-versus-all classifiers are a common solution for multi-class problems [11], where the problem is reduced to multiple independent binary classifiers. In a typical one-versus-all setup, an item is assigned the label with the highest score among the classifiers. However, as items in this task can have multiple labels, we simply assign labels according to the decision of each binary classifier.

The classifiers are based on the framework of SVM [12]. SVMs have been found to be very effective for text classification and tend to outperform other approaches such as NB [13]. For each label, we train a linear sentence classifier using the SVM^{light} toolkit [14]. The set of all sentences annotated with the label in question form positive examples for that classifier, with all remaining sentences used as negative examples. Section IV-B describes how the problem of imbalanced numbers of positive and negative examples in the data is alleviated by using unsymmetric cost factors during learning. First, however, Section IV-A below describes the feature functions that define the vector representation given to each sentence.

My Dearest Mother: I love you more than you can ever know.	LOVE
But I'm tired and I'm through with it all.	HOPELESSNESS
Jane Please take care of little John (?) as I love him very much.	INSTRUCTIONS, LOVE
xxxxxxxxxx January 01 2001 10:10 PM .	

Fig. 1. Example sentences from a suicide note in the shared task training data.

A. Features

We explored a range of different feature types for our emotion classifiers. The most basic features we employ are obtained by reducing inflected and derived words to their stem or base form, e.g., *happy*, *happiness*, *happily*, etc., all activate the stem feature *happi*. Together, the stem features provide a bag-of-words type representation for a given sentence. The word stems themselves are determined using the implementation of the Porter Stemmer [15] in the Natural Language Toolkit [16].

Another feature type records *bigrams* of stems (e.g., *happy days* activates the bigram feature *happi day*). We also investigated the use of longer *n*-grams in preliminary experiments, but found that they were counter-productive.

Lexicalized Part-of-Speech features are formed of word stems concatenated with their part-of-speech (PoS). PoS tags are assigned using TreeTagger [17], which is based on the Penn Treebank tagset.

Features based on syntactic dependency analysis provide us with a method for abstracting over syntactic patterns in the data set. The data is parsed with Maltparser, a language-independent system for data-driven dependency parsing [18]. We train the parser on a PoS-tagged version of the Wall Street Journal sections 2-21 of the Penn Treebank, using the parser and learner settings optimized for the Maltparser in the CoNLL-2007 Shared Task. The data was converted to dependencies using the Pennconverter software [19].

The parser was chosen partly due to its robustness to noise in the input data; it will not break down when confronted with incomplete sentences or misspelled words, but will always provide some output. While the amount of noise in the data will clearly affect the quality of the parses, we found that, in the context of this task, having at least some output is preferable to no output at all.

Consider the dependency representation provided for the example sentence in Fig. 2. The features we extract from the parsed data aim to generalize over the main predication of the sentence, and hence center on the root of the dependency graph (usually the finite verb) and its dependents. In the given example, the root is an auxiliary, and we traverse the chain of verbal dependents to locate the lexical main verb, *leave*," which we assume is more indicative of the meaning of the sentence than the auxiliary, *will*." The extracted feature types are as follows, with example instantiations based on the representation in Fig. 2:

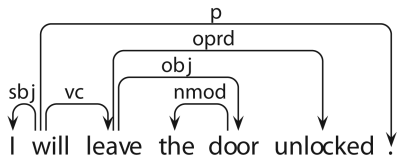

Fig. 2. Example dependency representation.

- Sentence dependency patterns: lexical features (word-form, lemma, PoS) of the root of the dependency graph, e.g., (`leave`, `leave`, `VV`), and patterns of dependents from the (derived) root, expressed by their dependency label, e.g., (`VC-OBJ-OPRD`), part-of-speech (`VV-NN-VVD`) or lemma (`leave-door-unlock`)
- Dependency triples: labelled relations between each head and dependent: `will-SBJ-I`, `will-VC-leave`, `leave-OPRD-unlocked`, etc.

We also include a class-based feature type recording the semantic relationships defined by WordNet synonym sets (synsets) [20]. These features are generated by mapping words and their PoS to the first synset identifier (WordNet synsets are sorted by frequency). For example, the adjectives *distraught* and *overwrought* both map to the synset id 00086555.

WordNet-Affect [9] is an extension of WordNet with affective knowledge pertaining to information such as emotions, cognitive states, etc. We utilize this information by activating features representing emotion classes when member words are observed in sentences. For example, instances of the words *wrath* or *irritation* both activate the WordNet-Affect feature *anger*.

In preliminary experiments, we investigated the difference in performance when representing feature frequency versus presence, as previous experiments in sentiment classification [21] indicated that unigram presence (i.e., a boolean value of 0 or 1) is more informative than their frequencies. For the suicide note analysis, however, we found that features encoding frequency rather than presence always performed better in our end-to-end experiments.

The final type of feature that we will describe represents the degree to which each stem in a sentence is associated with each label, as estimated from the training data. While there is a range of standardly used lexical association measures that could potentially be used for this purpose (such as point-wise mutual information, the Dice coefficient, etc.), the particular measure we will be using here is the *log odds ratio* ($\log \theta$). After first computing the relevant co-occurrence probabilities for a given word w and a label l in the training data, the odds ratio is calculated as:

$$\theta(w,l) = \frac{p(w,l)/p(w,\neg l)}{p(\neg w,l)/p(\neg w,\neg l)}$$

If the probability of having the label l increases when w is present, then $\theta(w, l) > 1$. If $\theta(w, l) = 1$ then w makes no difference in the probability of l, which means that the label and the word are distributionally independent. By taking the natural logarithm of the odds ratio, $\log \theta$, the score is made symmetric, with 0 being the neutral value that indicates independence. In order to incorporate this information in the classifier, we add features of all words in a given sentence towards each label, in addition to

Table 2. Developmental results of various feature types. The baseline corresponds to labeling all items as instructions, the majority class

Feature Set	Prec	Rec	F1
Baseline	18.27	**32.59**	23.27
Stems	70.69	27.53	**39.43**
Bigrams	74.84	21.49	33.21
Parts-of-Speech	74.76	21.51	33.20
Dependency Patterns	67.30	11.95	20.21
Dependency Triples	**75.58**	19.23	30.51
Synonym Sets	68.01	25.61	37.04
WordNetAffect	57.24	10.10	16.97
Association Score	64.91	25.63	36.58
Maximum Association	43.81	24.75	31.50

boolean features indicating which label had the maximum association score.

B. Cost-Sensitive Learning

From the frequencies listed in Table 1, it is clear that the label distributions are rather different. Moreover, for each individual classifier, it is also clear that the class balance will be very skewed, with the negative examples (often vastly) outnumbering the positives. At the same time, it is the retrieval of the positive minority class that is our primary interest. A well-known approach for improving classifier performance in the face of such skewed class distributions is to incorporate the notion of *cost-sensitive learning*. While this is sometimes done by the use of so-called *down-sampling* or *up-sampling* techniques [22], the SVM^{light} toolkit comes with built-in support for estimating cost-sensitive models directly. Working within the context of intensive care patient monitoring, but facing a similar setting of very unbalanced numbers of positive and negative examples, Morik et al. [23] introduced a notion of unsymmetric cost factors in SVM learning. This means associating different cost penalties with false positives and false negatives. Using the SVM^{light} toolkit, it is possible to train such cost models by supplying a parameter (j) that specifies the degree to which training errors on positive examples outweigh errors on negative examples (the default being $j = 1$, i.e., equal cost). In practice, the unsymmetric cost factor essentially governs the balance between precision and recall. The next section includes results of tuning the SVM cost-balance parameter separately for each emotion label in the suicide data and relative to different feature configurations.

V. EXPERIMENTAL RESULTS

As specified by the shared task organizers, overall system performance is evaluated using micro-averaged F_1. In addition, we also compute precision, recall and F_1 for each label individually. We report two rounds of evaluation. The first was conducted solely on the development data using ten-fold cross-validation (partitioning on the note-level). The second corresponds to the system submission for the shared task, i.e., training classifiers on the full development data and predicting labels for the notes in the held-out set.

A. Developmental Results

Table 2 lists the performance of each feature type in isolation (using the same feature configuration for each binary classifier and the default symmetric cost-balance). We also include the score for a simple baseline method that naively assigns the majority label (INSTRUCTIONS) to all sentences. We note that stems are the most informative feature type in isolation and perform best overall ($F_1 = 39.43$). Dependency Triples are most effective in terms of precision, and all feature types have less recall than the majority baseline.

In further experiments that examined the effect of using several feature types in combination, we found that combining stems, bigrams, parts-of-speech and dependency analyses achieved the best performance overall ($F_1 = 41.82$). However, these experiments also made it clear that different combinations of features were effective for different labels. Moreover, as our one-versus-all set-up means training distinct classifiers for each label, we are not limited to using one set of features for all labels. We therefore experimented with a grid search across different permutations of feature configurations, as further described below.

We also tuned the cost-balance parameter described in Section IV above. The reason for introducing the cost-balance parameter in our setup is to alleviate the imbalance between positive and negative examples. For some labels, this imbalance is so extreme that our initial system was unable to identify any positive predictions at all, neither true nor false. An example of such a label is forgiveness, which has only six annotated examples among the 4,241 sentences in the training data. Naturally, any supervised learning strategy will have problems making reliable generalizations on the basis of so little evidence. However, even for the more frequently occurring labels, the ratio of positive to negative examples is still quite skewed.

As we found that the optimal feature configuration was dependent on the value of the cost-balance parameter (and vice-versa), these parameters were optimized in parallel for each classifier. The results of this search are listed in Table 3, with the best feature combinations and cost-balance for each label. We note that the optimal configuration of features varies from label to label, but that stems and synonym sets are often in the optimal setup, while dependency triples and features from WordNetAffect do not occur in any configuration.

As discussed above, the unsymmetric cost factor essen-

Table 3. Labels in the suicide notes task with feature sets and cost-balance (j) optimized with respect to the local label F_1

Label	Features	Cost (*j*)	Prec	Rec	F_1
ABUSE	mas	50	0.17	10.00	0.33
ANGER	bos + sas	90	6.64	10.97	7.83
BLAME	bos + wns	15	17.02	27.05	19.16
FEAR	sas	5	10.00	10.00	10.00
FORGIVENESS	mas + wns	9	5.00	10.00	6.67
GUILT†	pos + wns	5	44.36	51.65	46.90
HAPPINESS/PEACEFULNESS	bos + sas	150	19.17	21.43	18.32
HOPEFULNESS	big + bos + wns	25	15.62	29.02	18.82
HOPELESSNESS†	big + bos + wns	6	54.56	55.37	54.07
INFORMATION†	dep + pos + wns	8	46.34	49.50	46.41
INSTRUCTIONS†	big + bos + dep + pos	3	69.27	66.40	67.32
LOVE†	big + bos + dep + pos	2	76.19	67.80	71.23
PRIDE	mas + wns	15	5.00	5.00	5.00
SORROW	mas + wns	5	12.33	11.36	10.37
THANKFULNESS†	bos + wns	4	69.47	69.44	67.77
micro-average (total)			46.00	54.00	49.41
micro-average†			61.09	51.71	55.81

Only the classifiers for labels marked with † are included in our final setup and in *micro-average*†, whereas *micro-average (total)* includes all labels. The feature types are big, bigrams over stems; bos, bag-of-stems; dep, sentence dependency patterns; mas, maximum association score; pos, parts-of-speech; sas, sum of association scores; wns, WordNet synsets.

tially governs the balance between precision and recall. For many classes, increasing the cost of errors on positive examples during training allowed us to achieve a pronounced increase in recall, though often at a corresponding loss in precision. Although this could often lead to greatly increased F_1 at the level of individual labels, the overall micro F_1 was compromised due to the low precision of the classifiers for infrequent labels in particular. Therefore, our final system only attempts to classify the six labels that can be predicted with the most reliability - GUILT, HOPELESSNESS, INFORMATION, INSTRUCTIONS, LOVE and THANKFULNESS - and makes no attempt on the remaining labels.

Testing by ten-fold cross-validation of the development data, this has the effect of an increased overall system performance in terms of the micro-average scores (compare *micro-average (total)* and *micro-average*† in Table 3). A further point of comparison is the optimal result of using identical setups for all classifiers, where $F_1 = 54.68$, precision = 60.57, and recall = 50.17 (using bag-of-stems, bigrams over stems, parts-of-speech, and sentence dependency patterns as features, and a cost-balance of 6). It should be noted that this rather radical design choice to only attempt classification of six labels is at least partially informed by the fact that micro-averaging (rather than macro-averaging) is used for the shared task evaluation. While micro-averaging is prone to emphasize larger classes, macro-averaging emphasizes smaller classes.

B. Held-out Results

Table 4 describes the performance on the held-out evaluation data set when training classifiers on the entire development data set, with details on each label attempted by our setup. As described above, we only apply classifiers for six of the labels in the data set (due to the low precision observed in the development results for the remaining nine labels). We find that the held-out results are quite consistent with those predicted by cross-validation on the development data. The final micro-averaged F_1 is 54.36, a drop of only 1.45 compared to the development result (see Section VII for a comparison of our system and its

Table 4. Performance of our optimized classifiers trained using the development data and tested on the held-out evaluation data

Label	Prec	Rec	F_1
GUILT	48.72	48.72	48.72
HOPELESSNESS	55.13	56.33	55.72
INFORMATION	37.41	50.00	42.80
INSTRUCTIONS	72.14	60.99	66.10
LOVE	77.99	61.69	68.89
THANKFULNESS	50.79	71.11	59.26
micro-average	60.58	49.29	54.36

The labels that are not attempted are not listed in the table (Prec = Rec = 0).

results to those of other participants in the shared task).

VI. ERROR ANALYSIS

This section offers some analysis and reflections with respect to the prediction errors made by our classifiers. Given the multi-class nature of the task, much of the discussion will center on cases where the system confuses two or more labels. Note that all example sentences given in this section are taken from the shared task evaluation data and are reproduced verbatim.

In order to uncover instances of systematic errors, we compiled contingency tables showing discrepancies between the decisions of the classifiers and the labels in the gold standard. Firstly, we note that BLAME and FORGIVENESS are often confused by our approach, and are closely related semantically. We consider these classes to be polar in nature; while both imply misconduct by some party, they elicit opposite reactions from the offended entity. Their similarity means that their instances often share features and are thus confused by our system.

We also note that the classes of GUILT and SORROW are hard to discern, not only for our system but also for the human annotators. For instance, Example 1 is annotated as SORROW, while Example 2 is annotated as GUILT. This makes features such as the stem of *sorry* prominent for both classes, hence our system often labels instances of either GUILT or SORROW with both labels. We also note some instances that are unlabelled but where the context is typically indicative of GUILT/SORROW, such as in Example 3. Furthermore, *sorry* appears to be a particularly ambiguous word; conceivably, it might also be associated with BLAME, (e.g., *you will be sorry*).

Example 1. Am sorry but I can't stand it ...

Example 2. I am truly sorry to leave ...

Example 3. ... sorry for all the trouble .

It is worth noting here that some of the apparent inconsistencies observed in the gold annotations are likely due to the way the annotation process was conducted. While three annotators separately assigned sentence-level labels, the final gold standard was created on the basis of majority vote between the annotators. This means that, unless two or more annotators agree on a label for a given sentence, the sentence is left unlabelled (with respect to the label in question).

Some of the labels in the data tend to co-occur. For instance, Example 1 above is actually annotated with both SORROW and HOPELESSNESS. However, these intuitively apply to two different sub-sentential units: *Am sorry* (SORROW) and *I can't stand it* (HOPELESSNESS). A problem that faced in any supervised learning approach here is the fact that the annotations are given at the sentence level, with no distinction between different sentence constituents or subsequences, and so the presence of a token like *sorry* can be deemed a positive feature for both SORROW and HOPELESSNESS by the learner. One possible avenue for improving results would therefore be to apply further annotation describing sub-sentential labels and the constituents to which they apply.

Note that the problem discussed above is also compounded due to errors in the sentence segmentation. For instance, Example 4 is provided as a single sentence in the training data, with the labels THANKFULNESS and HOPELESSNESS. However, as the labels actually apply to different sentences, this will introduce additional noise in the learning process.

Example 4. You have been good to me. I just cannot take it anymore.

Some of the errors made by the learner seem to indicate that having features that are sensitive to a larger context might also be useful, such as taking the preceding sentences and/or previous predictions into account. Consider the following examples from the same note, where both sentences are annotated as INSTRUCTIONS:

Example 5. In case of accident notify Jane.

Example 6. J. Johnson 3333 Burnet Avenue.

While Example 6 is simply an address, it is annotated as INSTRUCTIONS. Of course, predicting the correct label for this sentence in isolation from the preceding context will be near impossible. Other cases would seem to require information that is very different from that captured by our current features, such as pragmatic knowledge, before we could hope to get them right. For example, in several cases, the system will label something as INFORMATION when the correct label is INSTRUCTIONS. This is often because a sentence has communicated information which pragmatically implied an instruction. For example, we presume that Example 7 is annotated as INSTRUCTIONS because it is taken to imply an instruction to collect the clothes.

Example 7. Some of my clothes are at 3333 Burnet Ave. Cincinnati - just off of Olympic .

VII. CONCLUSIONS

This paper has provided experimental results for a variety of feature types for use when learning to identify various fine-grained emotions, as well as information and instructions, in suicide notes. These feature types range from simple bags-of-words to syntactic dependency analyses and information from manually-compiled lexical-semantic resources. We explored these features using an array of binary SVM classifiers.

A challenging property of this task is the fact the classifiers are subject to extreme imbalances between posi-

tive and negative examples in the training data; the infrequency of positive examples can make the learning task intractable for supervised approaches. In this paper, we have shown how a cost-sensitive learning approach that separately optimizes the cost-balance parameter for each of the topic labels, can be successfully applied for addressing problems with such skewed distributions of training examples. For the less-frequent labels, however, the optimal F_1 tended to arise from gains in recall at the great expense of precision. Thus, we found that discarding poorly-performing classifiers resulted in improvements overall. While arguably an ad hoc solution, this is motivated by the shared task evaluation scheme of maximizing micro-averaged F_1.

Of the twenty-five submissions to the shared task, our system was placed fifth (with a micro-averaged F_1 of 54.36); the highest-performer achieved an F_1 of 61.39, while the lowest scored 29.67. The mean result was 48.75 ($\sigma = 7.42$), and the median was 50.27. The primary differences between the system we describe and the other top-ranked approaches include: combining machine learning with heuristics [24, 25] and keyword-spotting for infrequent labels [25]; manually labeling training data from additional sources [24]; manually re-annotating training data to remove inconsistencies [26] and extracting other features (such as character *n*-grams [27] and spanning *n*-grams that skip tokens [24]).

An analysis of the errors made by our system has suggested possible instances of inter-annotator confusion, and has provided some indications for directions for future work. These include re-annotating data at the subsentential level, and drawing in the context and predictions of the rest of the note when labeling sentences. We also note that text in this domain tends to contain many typographical errors, and thus models might benefit from features generated using automatic spelling correction.

In other future work, we will conduct a search of the parameter space to find optimal parameters for each label with respect to the overall F_1 (rather than the label-local F_1 we used in the current work). Finally, we will look to boost performance for labels with few examples by drawing information from large amounts of unlabelled text. For instance, inferring the semantic similarity of words from their distributional similarity has been effective for other emotion-labeling tasks [28].

ACKNOWLEDGMENTS

We are grateful to the organizers of the 2011 Medical NLP Challenge for their efforts in compiling the data and managing the shared task. We also thank the anonymous reviewers and our colleagues for their helpful feedback. Large-scale experimentation was carried out with the titan high performance computing facilities at the University of Oslo.

REFERENCES

1. World Health Organization, Suicide prevention (SUPRE) [Internet]. http://www.who.int/mental_health/prevention/suicide/suicideprevent/en/.
2. H. Hjelmeland and B. L. Knizek, "Why we need qualitative research in suicidology," *Suicide and Life-Threatening Behavior*, vol. 40, no. 1, pp. 74-80, 2010.
3. J. P. Pestian, P. Matykiewicz, M. Linn-Gust, B. South, O. Uzuner, J. Wiebe, K. B. Cohen, J. Hurdle, and C. Brew, "Sentiment analysis of suicide notes: a shared task," *Biomedical Informatics Insights*, vol. 5, no. Suppl 1, pp. 3-16, 2012.
4. C. Strapparava and R. Mihalcea, "SemEval-2007 task 14: affective text," *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, Czech, 2007, pp. 70-74.
5. P. Ekman, "Biological and cultural contributions to body and facial movement," *The Anthropology of the Body*, New York: Academic Press, 1977. pp. 39-84.
6. P. Katz, M. Singleton, and R. Wicentowski, "SWAT-MP: the SemEval-2007 systems for task 5 and task 14," *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, Czech, 2007, pp. 308-313.
7. Z. Kozareva, B. Navarro, S. Vazquez, and A. Montoyo, "UA-ZBSA: a headline emotion classification through web information," *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, Czech, 2007, pp. 334-337.
8. F. R. Chaumartin, "UPAR7: a knowledge-based system for headline sentiment tagging," *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, Czech, 2007, pp. 422-425.
9. C. Strapparava and A. Valitutti, "WordNet-affect: an affective extension of WordNet," *Proceedings of the 4th International Conference on Language Resources and Evaluation*, Lisbon, Portugal, 2004, pp. 1083-1086.
10. C. Strapparava and R. Mihalcea, "Annotating and identifying emotions in text," *Intelligent Information Access. Studies in Computational Intelligence vol. 301*, Heidelberg: Springer Berlin, 2010, pp. 21-38.
11. K. B. Duan and S. S. Keerthi, "Which is the best multiclass SVM method? An empirical study," *Proceedings of the 6th International Workshop on Multiple Classifier Systems*, Seaside, CA, 2005, pp. 278-285.
12. V. N. Vapnik, *The Nature of Statistical Learning Theory*, New York: Springer, 1995.
13. T. Joachims, "Text categorization with support vector machines: learning with many relevant features," *Proceedings of the 10th European Conference on Machine Learning*, Chemnitz, Germany, 1998, pp. 137-142.
14. T. Joachims, "Making large-scale support vector machine learning practical," *Advances in Kernel Methods: Support Vector Learning*, Cambridge: MIT Press, 1999. pp. 169-184.
15. M. F. Porter, "An algorithm for suffix stripping," *Program: Electronic Library and Information Systems*, vol. 14, no. 3, pp. 130-137, 1980.
16. S. Bird and E. Loper, "NLTK: the natural language toolkit," *Proceedings of the ACL on Interactive Poster and Demonstration Sessions*, Barcelona, Spain, article no. 31, 2004.
17. H. Schmid, "Probabilistic part-of-speech tagging using decision trees," *Proceedings of the International Conference on*

New Methods in Language Processing, Manchester, UK, 1994, pp. 44-49.
18. J. Nivre, J. Hall, J. Nilsson, G. Eryigit, and S. Marinov, "Labeled pseudo-projective dependency parsing with support vector machines," *Proceedings of the 10th Conference on Computational Natural Language Learning*, New York, NY, 2006, pp. 221-225.
19. R. Johansson and P. Nugues, "Extended constituent-to-dependency conversion for English," *Proceedings of the 16th Nordic Conference of Computational Linguistics*, Tartu, Estonia, 2007, pp. 105-112.
20. C. Fellbaum, *WordNet: An Electronic Lexical Database*, Cambridge: MIT Press, 1998.
21. B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA, 2002, pp. 79-86.
22. K. McCarthy, B. Zabar, and G. Weiss, "Does cost-sensitive learning beat sampling for classifying rare classes?" *Proceedings of the 1st International Workshop on Utility-Based Data Mining*, Chicago, IL, 2005, pp. 69-77.
23. K. Morik, P. Brockhausen, and T. Joachims, "Combining statistical learning with a knowledge-based approach: a case study in intensive care monitoring." *Proceedings of the 16th International Conference on Machine Learning*, Bled, Slovenia, 1999, pp. 268-277.
24. Y. Xu, Y. Wang, J. Liu, Z. Tu, J. T. Sun, J. Tsujii, and E. Chang, "Suicide note sentiment classification: a supervised approach augmented by web data," *Biomedical Informatics Insights*, vol. 5, no. Suppl 1, pp. 31-41, 2012.
25. H. Yang, A. willis, A. de Roeck, and B. Nuseibeh, "A hybrid model for automatic emotion recognition in suicide notes," *Biomedical Informatics Insights*, vol. 5, no. Suppl 1, pp. 17-30, 2012.
26. S. Sohn, M. Torii, D. Li, K. Wagholikar, S. Wu, and H. Liu, "A hybrid approach to sentiment sentence classification in suicide notes," *Biomedical Informatics Insights*, vol. 5, no. Suppl 1, pp. 43-50, 2012.
27. C. Cherry, S. M. Mohammad, and B. de Bruijn, "Binary classifiers and latent sequence models for emotion detection in suicide notes," *Biomedical Informatics Insights*, vol. 5, no. Suppl 1, pp. 147-154, 2012.
28. J. Read and J. Carroll, "Weakly supervised techniques for domain-independent sentiment classification," *Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion*, Hong Kong, 2009, pp. 45-52.

Jonathon Read

Jonathon Read is a post-doctoral research fellow in the Language Technology Group at the Department of Informatics, University of Oslo, Norway. He holds a D.Phil. in Computer Science and Artificial Intelligence from the University of Sussex, UK, and his research interests include sentiment analysis, resolution of speculation and negation, and language resources.

Erik Velldal

Erik Velldal is a post-doctoral research fellow in the Language Technology Group at the Department of Informatics, University of Oslo, where he also completed his PhD. In his thesis, he focused on training statistical grammars for natural language generation, but has later worked on a variety of other topics such as emotion labeling, negation and speculation recognition, dimensionality reduction, and machine translation.

Lilja Øvrelid

Lilja Øvrelid is an associate professor in the Language Technology Group at the Department of Informatics, University of Oslo, Norway. She holds a PhD in computational linguistics from the University of Gothenburg. Her research interests include linguistically motivated, data-driven modelling and syntactic parsing.