

Topic Classification for Suicidology

Jonathon Read **Erik Vellidal** **Lilja Øvrelid**
Department of Informatics, University of Oslo
{jread,erikve,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 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.

1 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 with the rate increasing by 60% in the last 45 years (WHO, 2011). Researchers have recently called for more qualitative research in the fields of suicidology and suicide prevention (Hjelmeland and Knizek, 2010). Computational methods can expedite such analyses by labeling related texts with relevant topics.

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 informa-

tion 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.

The work described in this paper was conducted in the context of Track 2 of the 2011 Medical NLP Challenge on sentiment analysis in suicide notes (Pestian et al., In press). The task organizers provided developmental data consisting of 600 suicide notes, comprising of 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 2). For held-out evaluation the organizers provided an additional set of 300 unlabeled notes, comprising 1,883 sentences. The task organizers report an inter-annotator agreement rate of 54.6% over all sentences. Figure 1 provides excerpts from a note, with annotations.

We begin the remainder of the paper by providing some background on relevant work in Section 2. Section 3 details our approach, which involves training a collection of binary one-versus-all SVM sentence classifiers. Section 4 presents the performance of our approach, both under cross-validation on the development data and in final evaluation on held-out data. Section 5 analyses 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 6.

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

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

2 Related Work

We are not aware of any previous work on automatic thematic 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 (Strapparava and Mihalcea, 2007), which involved scoring newswire headlines according to the strength of the six so-called *basic emotions* stipulated by Ekman (1977)—ANGER, DISGUST, FEAR, JOY, SADNESS and SURPRISE. There were three participating systems in the SemEval-2007 emotion labeling task. SWAT (Katz et al., 2007) employed an affective lexicon where words’ relevance to emotions was scored as the average emotion score of every headline in which they appear. UA (Kozareva et al., 2007) also used a lexicon, which was instead compiled by calculating the pointwise mutual information with headline words and an emotion using counts obtained through information retrieval queries. UPAR7 (Chaumartin, 2007) employed heuristics over dependency graphs in conjunction with lexical resources such as WordNetAffect (Strapparava and Valitutti, 2004). In subsequent work, the task organizers investigated the application of latent semantic analysis (LSA) and a Naïve Bayes (NB) classifier trained using author-labeled blog posts (Strapparava and Mihalcea, 2010). As can perhaps be expected given the systems’ different approaches, each performed best on different emotions. This highlights the need for emotion-labeling systems to draw from a variety of analyses and resources.

3 Method

Our approach to the suicide notes labeling task involves learning a collection of binary *one-versus-all* classifiers. One-versus-all classifiers are a common solution for multi-class problems (Duan and Keerthi, 2005), 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 Support Vector Machines (SVM) (Vapnik, 1995). SVMs have been found to be very effective for text classification and tend to outperform other approaches such as Naïve Bayes (Joachims, 1998). For each label we train a linear sentence classifier using the SVM^{light} toolkit (Joachims, 1999). 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 3.2 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 3.1 below describes the feature functions that define the vector representation given to each sentence.

3.1 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 (Porter, 1980) in the Natural Language Toolkit (Bird and Loper, 2004).

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 Tree-

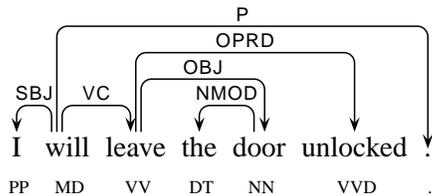


Figure 2: Example dependency representation.

Tagger (Schmid, 1994) 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 (Nivre et al., 2006). 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 (Johansson and Nugues, 2007).

Consider the dependency representation provided for the example sentence in Figure 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 Figure 2:

- Sentence dependency patterns:
Lexical features (wordform, 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:
Labeled 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) (Fellbaum, 1998). 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*.

WordNetAffect (Strapparava and Valitutti, 2004) 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 WordNetAffect feature *anger*.

In preliminary experiments we investigated the difference in performance when representing feature frequency versus presence, as previous experiments in sentiment classification (Pang et al., 2002) 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 pointwise 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 to 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 in-

icates independence. In order to incorporate this information in the classifier, we add features that record the sum of association scores of all words in a given sentence towards each label, in addition to boolean features indicating which label had the maximum association score.

3.2 Cost-Sensitive Learning

From the frequencies listed in Table 2 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 a notion of *cost-sensitive learning*. While this is sometimes done by the use of so-called *down-sampling* or *up-sampling* techniques (McCarthy et al., 2005), the SVM^{light} toolkit comes with built-in support for estimating cost-sensitive models directly, implementing the approach described by Morik et al. (1999). 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. introduced a notion of *unsymmetric cost factors* in SVM learning. This means associating different cost penalties to 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 by 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.

4 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

Feature Set	Prec	Rec	F_1
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

Table 1: Developmental results of various feature types. The baseline corresponds to labelling all items as INSTRUCTIONS, the majority class.

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 heldout set.

4.1 Developmental Results

Table 1 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 naïvely 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 feature were effective for different labels. Moreover, as our one-versus-all setup 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.

In parallel we also tuned the cost-balance pa-

Label	Frequency	%	Features	Cost (j)	Prec	Rec	F ₁
ABUSE	9	0.21	mas	50	0.17	10.00	0.33
ANGER	69	1.60	bos+sas	90	6.64	10.97	7.83
BLAME	107	2.52	bos+wns	15	17.02	27.05	19.16
FEAR	25	0.59	sas	5	10.00	10.00	10.00
FORGIVENESS	6	0.14	mas+wns	9	5.00	10.00	6.67
GUILT [†]	208	4.91	pos+wns	5	44.36	51.65	46.90
HAPPINESS	25	0.59	bos+sas	150	19.17	21.43	18.32
HOPEFULNESS	47	1.11	bos	25	15.62	29.02	18.82
HOPELESSNESS [†]	455	10.73	big+bos+wns	6	54.56	55.37	54.07
INFORMATION [†]	295	6.96	dep+pos+wns	8	46.34	49.50	46.41
INSTRUCTIONS [†]	820	19.34	big+bos+dep+pos	3	69.27	66.40	67.32
LOVE [†]	296	6.98	big+bos+dep+pos	2	76.19	67.80	71.23
PRIDE	15	0.35	mas+wns	15	5.00	5.00	5.00
SORROW	51	1.20	mas+wns	5	12.33	11.36	10.37
THANKFULNESS [†]	94	2.22	bos+wns	4	69.47	69.44	67.77
<i>micro average (total)</i>					46.00	54.00	49.41
<i>micro average (†)</i>					61.09	51.71	55.81

Table 2: Labels in the suicide notes task, with raw and relative frequencies (%) in the development data, optimal feature sets and cost-balance (j) parameters. Only the classifiers for labels marked with [†] are included in our final setup and are included in *micro average (†)*; *micro-average (total)* includes all labels. The feature types are: *dep* = sentence dependency patterns; *big* = bigrams over stems; *bos* = bag-of-stems; *mas* = maximum association score; *pos* = parts-of-speech; *sas* = sum of association scores; *wns* = WordNet synsets.

parameter described in the Methods section above. The reason for introducing the cost-balance parameter in our set-up 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 tuned in parallel. The results of this search are listed in Table 2, 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 Word-

NetAffect do not occur in any configuration.

As discussed above, the unsymmetric cost factor essentially 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 F₁s at the level of individual labels, in particular for the most low-frequency labels, the overall micro F₁ was compromised due to the low precision of the classifiers for infrequent labels. Therefore, in 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 on the development data, this has the effect of an increased overall system performance in terms of the micro-average scores. It should be noted that this rather radical design choice is at least partially informed by the fact that micro- (rather than macro-) averaging is used for the shared task evaluation. While micro-

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
<i>micro-average</i>	60.58	49.29	54.36

Table 3: Performance of our optimised classifiers trained on the development data and tested on the held-out evaluation data. The labels that are not attempted are not listed in the table (Prec = Rec = 0).

averaging is prone to emphasize larger classes, macro-averaging emphasizes smaller classes.

4.2 Held-out Results

Table 3 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 F₁ is 54.36, a drop of only 1.45 compared to the development result. Of the twenty-five submissions to the shared task, our system was placed fifth.¹

5 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 dis-

¹Details of the competing systems will be published in a forthcoming issue of *Biomedical Informatics Insights*; the highest-performer achieved an F₁ of 61.39, while the lowest scored 29.67. The mean result was 48.75 ($\sigma=7.42$), and the median was 50.27.

crepancies 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 semantically related; we consider these classes to be polar in nature as, while both imply misconduct by some party, they indicate opposite reactions by 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, the sentence

- (1) Am sorry but I ca n’t stand it any more .

is annotated as SORROW, while the sentence

- (2) I am truly sorry to leave without notice .

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 the sentence

- (3) Dear Jane Im sorry for all the trouble [...]

Furthermore, *sorry* appears to be a particularly ambiguous word; conceivably, it might also be associated with BLAME, (e.g. *you will be sorry*).

It is also worth noting 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 agreed on a label for a given sentence, the sentence was left unlabeled (with respect to the label in question).

Some of the labels in the data tend to co-occur. For example, Sentence 1 above is actually annotated with both SORROW and HOPELESSNESS. Intuitively, these apply to two different subsentential units, however; *Am sorry* (SORROW) and *I ca n’t stand it anymore* (HOPELESSNESS). A problem that faces 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 example, a sentence such as

- (4) You have been good to me . I just can not take it anymore .

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.

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. For instance, consider the following two-sentence note, where both sentences are annotated as INSTRUCTIONS:

- (5) In case of accident notify Jane .
(6) J. Johnson 3333 Burnet Avenue .

While the second sentence 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 to 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 the sentence

- (7) Some of my clothes are at 3333 Burnet Ave.
Cincinnati - just off of Olymipc .

is annotated as INSTRUCTIONS because it is taken to imply an instruction to collect the clothes.

6 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 with a set of binary SVM classifiers.

A challenging property of this task is the fact the classifiers are subject to extreme imbalances between positive 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, separately optimizing 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 a great expense in 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 .

Our error analysis has suggested possible instances of inter-annotator confusion, and provided some indications for directions in future work. These include re-annotating data at the sub-sentential level, and drawing in the context and predictions of the rest of the note when labeling sentences. We also note that text of this domain tends to contain many typographical errors, and might benefit from 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 unlabeled text. For instance, inferring semantic similarity of words from their distributional similarity has been effective for other emotion-labeling tasks (Read and Carroll, 2009).

Acknowledgements

We are grateful to the organizers of the 2011 Medical NLP Challenge for their efforts in compiling the data. We also thank the anonymous reviewers and our colleagues for their helpful feedback. Large-scale experimentation carried out with the TITAN HPC facilities at the University of Oslo.

References

- Steven Bird and Edward Loper. 2004. NLTK: The Natural Language Toolkit. In *Proceedings of the ACL demonstration session*, Barcelona.
- François-Régis Chaumartin. 2007. UPAR7: A knowledge-based system for headline sentiment tagging. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague.
- Kai-Bo Duan and S. Sathiya Keerthi. 2005. Which is the best multiclass svm method? an empirical study. In *Proceedings of the Sixth International Workshop on Multiple Classifier Systems*.
- Paul Ekman. 1977. Biological and cultural contributions to body and facial movement. In J. Blacking, editor, *Anthropology of the Body*, pages 34–84. Academic Press, London.
- Christine Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.
- Heidi Hjelmeland and Birthe Loa Knizek. 2010. Why we need qualitative research in suicidology. *Suicide and Life-Threatening Behavior*, 40(1).
- Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of the European Conference on Machine Learning*.
- Thorsten Joachims. 1999. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods — Support Vector Learning*. MIT Press.
- Richard Johansson and Pierre Nugues. 2007. Extended constituent-to-dependency conversion for English. In *Proceedings of NODALIDA 2007*.
- Phil Katz, Matthew Singleton, and Richard Wicentowski. 2007. SWAT-MP: The SemEval-2007 systems for task 5 and task 14. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, June.
- Zornitsa Kozareva, Borja Navarro, Sonia Vázquez, and Andrés Montoyo. 2007. UA-ZBSA: A headline emotion classification through web information. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, June.
- Kate McCarthy, Bibi Zabar, and Gary Weiss. 2005. Does cost-sensitive learning beat sampling for classifying rare classes? In *Proceedings of the 1st International Workshop on Utility-Based Data Mining*, Chicago, Illinois.
- Katharina Morik, Peter Brockhausen, and Thorsten Joachims. 1999. Combining statistical learning with a knowledge-based approach — a case study in intensive care monitoring. In *Proceedings of the 16th International Conference on Machine Learning*, Bled, Slovenia.
- Joachim Nivre, Jens Nilsson, Johan Hall, Gülşen Eryiğit, and Svetoslav Marinov. 2006. Labeled pseudo-projective dependency parsing with Support Vector Machines. In *Proceedings of the Conference on Computational Natural Language Learning (CoNLL)*.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*.
- John P. Pestian, Pawel Matykiewicz, Michelle Linn-Gust, Jan Wiebe, Kevin Cohen, Christopher Brew, John Hurdle, Ozlem Uzuner, and Brett South. In press. Sentiment analysis of suicide notes: A shared task. *Biomedical Informatics Insights*.
- Martin F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.
- Jonathon Read and John Carroll. 2009. Weakly supervised techniques for domain-independent sentiment classification. In *Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion Measurement*, Hong Kong.
- Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*.
- Carlo Strapparava and Rada Mihalcea. 2007. Semeval-2007 task 14: Affective text. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, Prague, June.
- Carlo Strapparava and Rada Mihalcea. 2010. Annotating and identifying emotions in text. In *Intelligent Information Access*, Studies in Computational Intelligence. Springer Berlin / Heidelberg.
- Carlo Strapparava and Alessandro Valitutti. 2004. Wordnet-affect: an affective extension of Wordnet. In *Proceedings of the 4th International Conference of Language Resources and Evaluation*, Lisbon.
- Vladimir Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer, New York, USA.
- WHO. 2011. Suicide prevention (SUPRE). Downloaded from http://www.who.int/mental_health/prevention/suicide/suicideprevent on 10 September 2011.