Kernel models for affective lexicon creation

Nikos Malandrakis[1], *Alexandros Potamianos*[1], *Iosif Elias*[1], *Shrikanth Narayanan*[2]

[1]Dept. of ECE, Technical Univ. of Crete, 73100 Chania, Greece
[2]SAIL Lab, Dept. of EE, Univ. of Southern California, Los Angeles, CA 90089, USA
[nmalandrakis,potam,iosife]@telecom.tuc.gr, shri@sipi.usc.edu

Abstract

Emotion recognition algorithms for spoken dialogue applications typically employ lexical models that are trained on labeled in-domain data. In this paper, we propose a domain-independent approach to affective text modeling that is based on the creation of an affective lexicon. Starting from a small set of manually annotated seed words, continuous valence ratings for new words are estimated using semantic similarity scores and a kernel model. The parameters of the model are trained using least mean squares estimation. Word level scores are combined to produce sentence-level scores via simple linear and non-linear fusion. The proposed method is evaluated on the SemEval news headline polarity task and on the ChIMP politeness and frustration detection dialogue task, achieving state-of-the-art results on both. For politeness detection, best results are obtained when the affective model is adapted using in domain data. For frustration detection, the domain-independent model and non-linear fusion achieve the best performance.
Index Terms: language understanding, emotion, affect, affective lexicon

1. Introduction

An important research problem, relevant for interactive spoken dialogue and natural language system design, is the analysis of the affective content of user input. Recently, significant progress has been made in identifying acoustic linguistic and pragmatic/interaction features for emotion recognition in interactive systems [1, 2, 3]. In this paper, we focus specifically on the use of lexical information for affect modeling and emotion recognition. Lexical models of affect typically employ words or groups of words as features, and rely on in-domain data to train simple statistical models. Dimensionality reduction and feature selection methods have been proposed in the literature that employ latent semantic analysis or mutual information criteria (emotional saliency) [1, 2]. Although such methods have been successful, they are both application and emotion recognition task dependent. In this paper, we investigate a domain-independent approach to affective text analysis, as well as, adaptation of affective models to a new application or domain using very little labeled data.

Affective text characterization, the assignment of affective labels to lexical units, is relevant for many other applications beyond interactive systems, e.g., market analysis, opinion mining, multimedia content analysis. Due to the variety of the different affective representations (categorical vs dimensional, discrete vs continuous), perspectives (speaker emotion, acted emotion listener/reader emotion), task needs (word, sentence, full text characterization), and research communities involved (web, natural language, speech, multimedia) there is a significant fragmentation of the research effort. For spoken dialogue systems, the emphasis is on identifying hot-spots in the interaction, thus, the binary characterization of the emotion space in frustration/annoyance vs neutral [1] is usually adequate. For sentiment analysis binary affective ratings using "positive - negative" labels, also known as polarity, is more appropriate and has received much research attention. Here we attempt to provide a unified domain-independent framework for both types of affective categorization tasks.

Domain-independent approaches to affect modeling have at their core an *affective lexicon*, i.e., a resource mapping words to a set of affective ratings. There exists a number of manually created affective lexicons for English, e.g., the Affective norms for English Words (ANEW) [4], but such lexicons typically contain only a few thousand words, failing to provide good coverage. Therefore computational methods are used to create or expand an already existing lexicon, e.g., [5]. For the vast majority of these methods, the underlying assumption is that *semantic similarity can be translated to affective similarity*. Therefore, given some metric of the similarity between two words, e.g., [6, 5, 7], one can derive the similarity between their affective ratings. The final step is the combination of these word ratings to create ratings for larger lexical units, phrases or sentences [8, 9]. Individual word ratings may be combined using simple numerical average or using rules that incorporate linguistic information, e.g., valence shifters.

Our aim in this paper, is to investigate kernel models of affect for the purpose of lexicon creation. The models can be trained from a small set of labeled words and then extended to unseen words in new application domains. Word levels ratings are combined to compute sentence-level ratings using simple fusion schemes. These domain-independent models are evaluated on both sentiment analysis and spoken dialogue systems datasets. Finally, we investigate the use of small amounts of in-domain data for adapting the affective models. Results show that domain-independent models perform very well for certain tasks, especially, for frustration detection in spoken dialogue systems.

2. Kernel Methods for Affect Modeling

In this paper, we extend the approach pioneered in [5]. Starting from a set of words with known affective ratings, the rating of a new (unseen) word is estimated as a function of the semantic similarities between the unseen word and each one of the known words. These reference words are usually referred to as *seed words*. Here we propose a weighted combination of the similarity and valence scores of the seed words to produce the valence rating of the unseen words. Adding a seed word-dependent weight to the affective model is motivated by the fact that not all features (seed words) are equally informative. For example, seed words that have high affective variance (differ-

ence senses of the word have very different valence ratings) are expected to be worse features than seed words with low variance. Thus, every seed word is assigned a weight that modifies its importance in determining the rating of new words. Because the assignment of weights to the seed words is too complex to model analytically, we propose a supervised method to estimate them from an existing lexicon, using *Least Mean Squares (LMS) estimation*.

The proposed affective model assumes that the continuous valence ratings in $[-1, 1]$ (similarly for other affective dimensions) of any word can be represented as a linear combination of a function of its semantic similarities to a set of seed words and the valence ratings of these words, as follows:

$$\hat{v}(w_j) = a_0 + \sum_{i=1}^{N} a_i \; v(w_i) \; f(d_{ij}), \tag{1}$$

where w_j is the word we mean to characterize, $w_1 ... w_N$ are the seed words, $v(w_i)$ is the valence rating for seed word w_i, a_i is the weight corresponding to word w_i (that is estimated as described next), d_{ij} is a measure of semantic similarity between words w_i and w_j and $f()$ is some function. Assuming we have a training corpus of K words with known ratings and a set of $N < K$ seed words for which we need to estimate weights a_i, we can use (1) to create a system of K linear equations with with $N + 1$ unknown variables as shown in (2); the N weights $a_1 ... a_N$ and the extra weight a_0 which acts as a DC offset (bias). The optimal values of these variables can be estimated using LMS. Once the weights of the seed words are estimated the valence of an unseen word w_j can be computed using (1). The functions $f()$ that were used in our experiments[1] are shown in Table 1.

$$\begin{bmatrix} 1 & f(d_{11})v(w_1) & \cdots & f(d_{1N})v(w_N) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & f(d_{K1})v(w_1) & \cdots & f(d_{KN})v(w_N) \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} 1 \\ v(w_1) \\ \vdots \\ v(w_K) \end{bmatrix} \tag{2}$$

linear	$f(d(\bullet)) = d(\bullet)$
exp	$f(d(\bullet)) = e^{d(\bullet)}$
log	$f(d(\bullet)) = log(d(\bullet))$
sqrt	$f(d(\bullet)) = \sqrt{d(\bullet)}$

Table 1: *The functions of similarity used.*

An essential component of the proposed method is the semantic similarity metric used in (1). In this paper, we use hit-based similarity metrics that estimate the similarity between two words/terms using the frequency of co-existence within larger lexical units (sentences, documents). The underlying assumption is that terms that co-exist often are very likely to be related. A popular method to estimate co-occurrence is to pose conjunctive queries including both terms to a web search engine; the number of returned hits is an estimate of the frequency of co-occurrence [10]. Hit-based metrics do not depend on any language resources, e.g., ontologies, and do not require downloading documents or snippets, as is the case for context-based semantic similarities. Here we use the *google semantic relatedness* metric that is defined in [11]

$$G(w_i, w_j) = e^{-2E(w_i, w_j)} \tag{3}$$

as a function of the Normalized Google Distance [12]

$$E(w_i, w_j) = \frac{\max\{L\} - \log \mid D; w_i, w_j \mid}{\log \mid D \mid - \min\{L\}}, \tag{4}$$

where $w_i, \ldots, w_{i+n}$ are the query words, $\{D; w_i, \ldots, w_{i+n}\}$ is the set of results $\{D\}$ returned for these query words, $\mid D; w_i, \ldots, w_{i+n} \mid$ is the number of documents in each result set and $L = \{\log \mid D; w_i \mid, \; \log \mid D; w_j \mid\}$. To get the required co-occurrence hit-count we use simple "AND" queries.

2.1. Sentence Level Tagging

To produce sentence-level scores the word-level scores have to be combined. Here we perform no feature selection, all words in a sentence contribute to the final affective rating using simple linear and nonlinear fusion schemes. The simplest model computes the valence of a sentence simply by taking the average valence of all words in that sentence. The affective content of a sentence $s = w_1 w_2 ... w_N$ for this linear average model is:

$$v(s) = \frac{1}{N} \sum_{i=1}^{N} v(w_i). \tag{5}$$

Simple linear fusion is a crude approximation given that nonlinear affective interaction between words (especially adjacent words) in the same sentence is common. In general, words with high (absolute) affective scores are expected to be more important in determining the sentence level scores. Thus, we also consider a normalized weighted average fusion scheme, where words with high absolute valence values are weighted more, as follows:

$$v(s) = \frac{1}{\sum_{i=1}^{N} |v(w_i)|} \sum_{i=1}^{N} v(w_i)^2 \cdot \text{sign}(v(w_i)), \tag{6}$$

where sign$(.)$ is the signum function (other non-linear scaling functions could be also used here instead of square). Alternatively, we consider non-linear max fusion, where the word with the highest absolute valence value dominates the meaning of the sentence:

$$\begin{aligned} v(s) &= \max_i(|v(w_i)|) \cdot \text{sign}(v(w_z)) \\ z &= \arg\max_i(|v(w_i)|) \end{aligned} \tag{7}$$

where $\arg\max$ is the argument of the maximum.

3. Corpora and Experimental Procedure

Three corpora were used in this work: (i) the ANEW word corpus for the training of the affective model and the evaluation of the affective lexicon, (ii) the SemEval headline polarity corpus (positive vs negative valence evaluation task) and (iii) the ChIMP spoken dialogue corpus (politeness and frustration detection tasks).

The *Affective Norms for English Words* (ANEW) dataset contains 1034 words, rated in 3 continuous dimensions of arousal, valence and dominance. We performed a 10-fold cross-validation experiment using the ANEW dataset. On each fold

[1] An alternative method to the one described above we also used various Support Vector Machines (SVMs) kernels to perform the same task, using the same word similarities as features. For more details see the experimental section.

90% of the words were used for training and 10% for evaluation. The seed words were selected using a maximum absolute valence criterion. Words were added to the seed set in descending absolute valence order[2]. Then the linear equation system matrix was created and LMS estimation was performed to calculate the weights. Finally, the resulting equation was used to estimate the ratings of words in the evaluation set. An example of the estimated weights (linear similarity function) for a small number of features is shown in Table 2. The final column, $v(w_i) \times a_i$, is a measure of the affective "shift" of the valence of a word (provided that the similarity between this word and the seed word w_i is 1). Note that the weights a_i take positive values but are not bounded in $[0, 1]$.

w_i	$v(w_i)$	a_i	$v(w_i) \times a_i$
mutilate	-0.8	0.75	-0.60
intimate	0.65	3.74	2.43
poison	-0.76	5.15	-3.91
bankrupt	-0.75	5.94	-4.46
passion	0.76	4.77	3.63
misery	-0.77	8.05	-6.20
joyful	0.81	6.4	5.18
optimism	0.49	7.14	3.50
loneliness	-0.85	3.08	-2.62
orgasm	0.83	2.16	1.79
w_0	1	0.28	0.28

Table 2: *Estimated weights for a set of 10 seed words.*

Next, the SemEval corpus was used to validate the sentence-level performance of our method. The *SemEval 2007: Task 14* corpus [13] contains 1000 news headlines manually rated in a fine-grained valence scale $[-100, 100]$ (rescaled to $[-1, 1]$ for our experiments). We perform a binary classification experiment on this corpus, attempting to detect sentences with positive (vs negative) valence. The affective lexicon is expanded with the words in the SemEval corpus using the model in (1) trained on all the words of the ANEW corpus (N of them used as seed words). The word level scores are combined using one of the three fusion methods to obtain sentence-level scores.

Finally, the ChIMP database was used to evaluate the method on spontaneous spoken dialog interaction. The ChIMP corpus contains 15585 manually annotated spoken utterances, with each utterance labeled with one of three emotional state tags: neutral, polite, and frustrated [14]. While the labels reflect emotional states, their valence rating is not obvious. In order to adapt the affective model to the ChIMP task, the discrete sentence level valence scores were mapped as follows: frustrated was assigned a valence value of -1, neutral was 0 and polite was 1. To bootstrap the valence scores for each word in the ChIMP corpus, we used the average sentence-level scores for all sentence where that word appeared. Finally, the ANEW equation system matrix was augmented with all the words in the ChIMP corpus and the valence model in (2) was estimated using LMS. Note that for this training process a 10-fold cross validation experiment was run on the ChIMP corpus sentences. The relative *weight* of the ChIMP corpus adaptation data was varied by adding the respective lines multiple times to the augmented system matrix, e.g., adding each line twice gives a weight of $w = 2$. We tested weights of $w = 1$, $w = 2$, and using only the samples from ChIMP as training samples (denoted as $w = \infty$). The valence boundary between frustrated and other classes was selected based on the a-priori probability distribution for each class, and is simply the Bayesian decision boundary (similarly between polite and other classes).

[2]We have also tested wrapper feature selection using a minimum mean square error criterion, as well as, random feature selection. Random feature selection gave the poorest results but the differences compared to wrapper (that performed the best) were small; up to 0.04 in correlation scores for the ANEW corpus. Maximum absolute valence was selected here because it requires no training, and gives a good trade-off between performance and complexity.

4. Results

In Fig. 1, the performance of the kernel models are evaluated for the task of affective lexicon creation on the ANEW corpus. Two-class classification accuracy (positive vs negative valence[3]) is shown as a function of the number of seed words N in the model (1). Results are shown for the similarity functions $f()$ in Table 2 and for SVM classifiers (linear and polynomial kernel). Overall, the proposed method produces state-of-the-art classification accuracy at around 85%. Best results are achieved with the linear SVM kernel for a small number of seed words[4]. SVMs using more complex kernels were tested, but performed poorly, probably due to the small number of training samples (here results for the polynomial kernel are shown as an example). The kernel models of (1) also achieve very good performance. Among the similarity functions, the "linear" and "exp" functions are the top performers but the differences are small. For a large number of seed words (over 300), over-fitting occurs for all methods and performance deteriorates slightly. A larger starting vocabulary would enable us to use even more features effectively. However, even with a small number of seed words the proposed method achieves very competitive results.

In Table 3, the two-class sentence-level classification accuracy is shown for the SemEval (positive vs negative) and ChIMP corpora (polite vs other: "P vs O", frustrated vs other: "F vs O"). For the SemEval and baseline ChIMP experiments, 200 words from the ANEW corpus were used to train the affective model in (1) using the linear similarity function. For the adaptation experiments on the ChIMP corpus, the parameter w denotes the weighting given to the in-domain ChIMP data, i.e., number of times the adaptation equation were repeated in the system matrix (2). Results are shown for the three fusion methods (average, weighted average, maximum).

For the SemEval dataset classification accuracy is just below 70%, significantly higher than that reported in [15] and on par with that reported in [16] when evaluating performance on *all* the sentences in the dataset. For the ChIMP politeness detection task, performance of the baseline (unsupervised) model is lower than that quoted in [14] for lexical features. Performance improves significantly by adapting the affective model using in-domain ChIMP data reaching up to 84% accuracy for linear fusion (matching the results in [14]). The best results for frustration detection task are achieved with the baseline model and max fusion schemes at 66% (good or better than the ones reported in [14]). It is interesting to note that in-domain adaptation does not improve frustration classification. A possible

[3]Ratings for all words in ANEW were produced in a 10-fold cross-validation experiment, then compared to the ground truth (manual annotations of the ANEW corpus). It should be stressed that, on every step of the cross-validation experiment, words that belong to the evaluation set are not eligible to be selected as seed words.

[4]SVM results for less than 80 seed words are not presented, because the training algorithms failed to converge, indicating perhaps a higher dependency of SVMs on well-selected features.

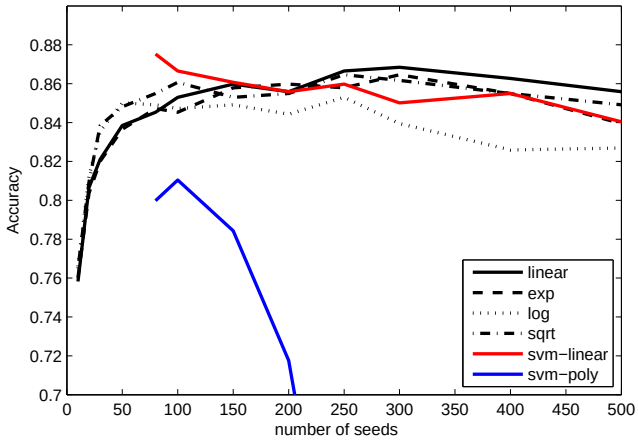

Figure 1: Two-class word classification accuracy (positive vs negative valence) vs the number of seed words for the ANEW corpus.

explanation is that there is a high lexical variability when expressing frustration, thus, the limited adaptation data does not help much. Also frustration may be expressed with a single word that has very negative valence, as a result, max fusion works best here. Overall, very good results are achieved using a domain-independent affective model to classify politeness and frustration. However, the appropriate adaptation and sentence-level fusion schemes seem to be very much task-dependent.

Sentence Classification Accuracy			
	avg	w.avg	max
SemEval baseline	0.67	0.68	**0.69**
ChIMP (P vs O) baseline	**0.70**	0.69	0.54
ChIMP (P vs O) adapt $w = 1$	**0.74**	0.70	0.67
ChIMP (P vs O) adapt $w = 2$	**0.77**	0.74	0.71
ChIMP (P vs O) adapt $w = \infty$	**0.84**	0.82	0.75
ChIMP (F vs O) baseline	0.53	0.62	**0.66**
ChIMP (F vs O) adapt $w = 1$	0.51	**0.58**	0.57
ChIMP (F vs O) adapt $w = 2$	0.49	**0.53**	0.53
ChIMP (F vs O) adapt $w = \infty$	0.52	0.52	0.52

Table 3: *Sentence classification accuracy for the SemEval, ChIMP baseline and ChIMP adapted tasks.*

5. Conclusions

We proposed and evaluated a method for creating an affective lexicon starting for a few hundred annotated seed words. For this purpose, kernel models of affect have been trained using LMS; the assumption behind these models is that similarity of meaning implies similarity of affect. New words can be easily added to the lexicon using the affective model. The process is fully unsupervised and domain-independent; it relies only on a web search engine to estimate semantic similarity between the new words and the seed words. Finally, we presented three fusion metrics that are used to estimate sentence-level scores from word-level scores. The proposed method was evaluated on the ANEW, SemEval and ChIMP datasets. For politeness detection it was shown that adaptation of the affective model and linear fusion achieves the best results. For frustration detection, the domain-independent model and max fusion gave the best performance. Overall, we have shown that an unsupervised domain-independent approach is a viable alternative to training domain-specific language models for the problem of affective text analysis. However, more research is needed to identify the appropriate sentence-level fusion method and the amount of in-domain adaptation data necessary to optimize performance for each task.

6. Acknowledgements

Iosif Elias was funded by the Basic Research Programme, Technical University of Crete, Project Number 99637: "Unsupervised Semantic Relationship Acquisition by Humans and Machines: Application to Automatic Ontology Creation".

7. References

[1] J. Ang, R. Dhillon, A. Krupski, E. Shriberg, and A. Stolcke, "Prosody-based automatic detection of annoyance and frustration in human-computer dialog," in *Proceedings of ICSLP*, Denver, 2002, pp. 2037–2039.

[2] C. M. Lee and S. Narayanan, "Towards detecting emotions in spoken dialogs." *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 2, pp. 293–302, 2005.

[3] B. Schuller, D. Seppi, A. Batliner, A. Maier, and S. Steidl, "Towards more reality in the recognition of emotional speech," in *Proceedings of ICASSP*, vol. 4, 2007, pp. 941–944.

[4] M. Bradley and P. Lang, "Affective norms for english words (ANEW): Stimuli, instruction manual and affective ratings. Technical report C-1." The Center for Research in Psychophysiology, University of Florida, 1999.

[5] P. Turney and M. L. Littman, "Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus. Technical report ERC-1094 (NRC 44929)." National Research Council of Canada, 2002.

[6] A. Andreevskaia and S. Bergler, "Semantic tag extraction from WordNet glosses," in *Proc. LREC*, 2006, pp. 413–416.

[7] M. Taboada, C. Anthony, and K. Voll, "Methods for creating semantic orientation dictionaries," in *Proc. LREC*, 2006, pp. 427–432.

[8] F.-R. Chaumartin, "UPAR7: A knowledge-based system for headline sentiment tagging," in *Proc. SemEval*, 2007, pp. 422–425.

[9] A. Andreevskaia and S. Bergler, "CLaC and CLaC-NB: Knowledge-based and corpus-based approaches to sentiment tagging," in *Proc. SemEval*, 2007, pp. 117–120.

[10] E. Iosif and A. Potamianos, "Unsupervised Semantic Similarity Computation Between Terms Using Web Documents," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 11, pp. 1637–1647, 2009.

[11] J. Gracia, R. Trillo, M. Espinoza, and E. Mena, "Querying the web: A multiontology disambiguation method," in *Proc. of International Conference on Web Engineering*, 2006, pp. 241–248.

[12] P. M. Vitnyi, "Universal similarity," in *Proc. of Information Theory Workshop on Coding and Complexity*, 2005, pp. 238–243.

[13] C. Strapparava and R. Mihalcea, "Semeval-2007 task 14: Affective text," in *Proc. SemEval*, 2007, pp. 70–74.

[14] S. Yildirim, S. Narayanan, and A. Potamianos, "Detecting emotional state of a child in a conversational computer game," *Computer Speech and Language*, vol. 25, pp. 29–44, January 2011.

[15] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon-based methods for sentiment analysis," *Computational Linguistics*, vol. 1, pp. 1–41, 2010.

[16] K. Moilanen, S. Pulman, and Y. Zhang, "Packed feelings and ordered sentiments: Sentiment parsing with quasi-compositional polarity sequencing and compression," in *Proc. WASSA Workshop at ECAI*, 2010, pp. 36–43.