Sentiment Analysis of Text–to–Speech Input Using Latent Affective Mapping

Jerome R. Bellegarda

Speech & Language Technologies, Apple Inc.
1 Infinite Loop, Cupertino, CA 95014, USA
jerome@apple.com

***Abstract*—To impart a congruent emotional quality to synthetic speech, it is expedient to leverage the overall polarity of the input text. This is feasible inasmuch as speech generation complies with the outcome of sentiment analysis. We have recently introduced *latent affective mapping* [1]–[3], a new approach to emotion detection which exploits two separate levels of semantic information: one that encapsulates the foundations of the domain considered, and one that specifically accounts for the overall affective fabric of the language. The ensuing framework exposes the emergent relationship between these two levels in order to advantageously inform affective evaluation. This paper applies latent affective mapping to the narrower problem of sentiment analysis, in order to achieve a more robust identification of the polarity of textual data. Empirical evidence gathered on the "Affective Text" portion of the SemEval-2007 corpus [4] shows that this approach is promising for automatic sentiment prediction in text. This bodes well as a first step in ensuring emotional congruence in text-to-speech synthesis.**

I. Introduction

A long-term objective of text-to-speech (TTS) synthesis is to make synthetic speech as expressive as natural speech. In order to attain that goal, the speech produced must be emotionally congruent with the underlying textual input, at least to the extent that such congruence is perceptually salient. Given the current state of emotion detection from text, this is not yet within reach. For all practical purposes, however, it may be sufficient to avoid synthesizing speech with a grossly inappropriate emotional quality. Under that hypothesis, it is better to conform to the general sentiment of the input than risk reflecting a more specific, but incorrect, emotional state.

To make progress in that direction, it is necessary to solve two complementary sub-problems: (i) the overall polarity must be identified from the given input text, and (ii) the corresponding modifications must be effected in speech generation [5]. A number of techniques, often closely tied to the synthesis framework adopted, have been explored to address (ii): cf., e.g., [6]. Comparatively less attention has been given to (i), especially in a TTS context. Typical solutions perform affective analysis based on an underlying emotional knowledge database, i.e., affective information is either built entirely upon manually selected vocabulary as in [7], or derived automatically from data based on expert knowledge of the most relevant features that can be extracted from the input text [8]. In both cases, lexical features emerge as overwhelmingly dominant, and the net effect is to rely on a few thousand annotated "affective terms," the presence of which triggers an emotional label, and thereby the associated polarity value.

We have recently introduced a new framework for emotion detection and classification [1]–[3], based on two separate levels of semantic information: one that encapsulates the foundations of the domain considered, and one that specifically accounts for the overall affective fabric of the language. This approach leverages the latent topicality of two distinct corpora, as uncovered by a global latent semantic mapping (LSM) analysis [9]. The emergent relationship between the two levels is then exploited to expose the desired connection between all terms and emotional categories. Because this connection automatically takes into account the influence of the entire training corpora, it is more encompassing than that based on the relatively few "affective terms" typically considered in conventional processing. As a result, it effectively bypasses the need for any explicit external information.

In [1]–[3], latent affective mapping was observed to perform better than standard emotion classification approaches based on affective weights and similar expert knowledge. In this paper, we apply that framework to the narrower problem of sentiment analysis, in order to achieve a more robust classification of the polarity of a given text. The paper is organized as follows. After briefly reviewing the general approach in the next section, we take a more detailed look than in [1]–[3] at two mapping instantiations: latent folding in Section III, and latent embedding in Section IV. Section V then describes the mechanics of sentiment prediction given the resulting affective description. Finally, in Section VI we report the outcome of experimental evaluations conducted on the "Affective Text" portion of the SemEval-2007 corpus [4].

II. Latent Affective Mapping

Let $\mathcal{T}_D$, $|\mathcal{T}_D| = N_D$, be a collection of training texts (be they sentences, paragraphs, or documents) reflecting the domain of interest, and $\mathcal{V}_D$, $|\mathcal{V}_D| = M_D$, the associated set of all words (possibly augmented with some strategic word pairs, triplets, etc., as appropriate) observed in this collection. Similarly, let $\mathcal{T}_A$, $|\mathcal{T}_A| = N_A$, represent a separate collection of mood-annotated texts (again of arbitrary granularity), representative of broad affective categories (such as JOY and SADNESS). We denote by $\mathcal{V}_A$, $|\mathcal{V}_A| = M_A$, the associated set of words or expressions observed in this collection.

978-1-4673-0367-5/11/$26.00 ©2011 IEEE

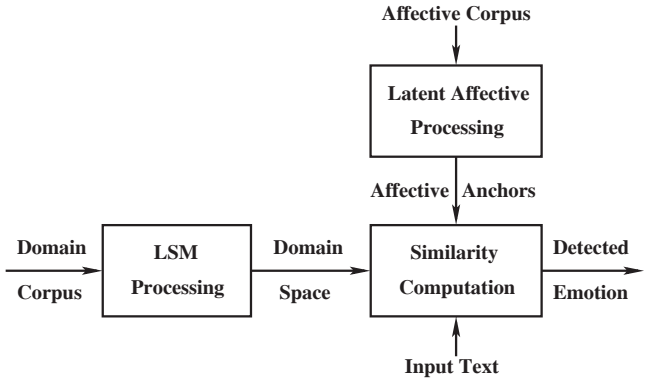

Fig. 1. *Affective Analysis Using Latent Affective Mapping.*

Latent affective mapping proceeds as illustrated in Fig. 1. First, we use LSM to encapsulate the semantic information present in the domain corpus $\mathcal{T}_D$ (which corresponds to the *LSM processing* block on the figure). This is done by constructing a $(M_D \times N_D)$ matrix W_D, whose elements w_{ij} suitably reflect the extent to which each word $w_i \in \mathcal{V}_D$ appeared in each text $t_j \in \mathcal{T}_D$. From [1], w_{ij} is given by:

$$w_{i,j} = (1 - \varepsilon_i)\,\frac{c_{i,j}}{n_j}\,, \tag{1}$$

where $c_{i,j}$ is the number of times w_i occurs in text t_j, n_j is the total number of words present in this text, and ε_i is the normalized entropy of w_i in $\mathcal{V}_D$ (cf. [9]). We then perform a singular value decomposition (SVD) [10]:

$$W_D \simeq U_D\, S_D\, V_D^T\,, \tag{2}$$

where U_D is the $(M_D \times R_D)$ left singular matrix with row vectors $u_{D,i}$ $(1 \leq i \leq M_D)$, S_D is the $(R_D \times R_D)$ diagonal matrix of singular values $s_{D,1} \geq s_{D,2} \geq \ldots \geq s_{D,R_D} > 0$, V_D is the $(N_D \times R_D)$ right singular matrix with row vectors $v_{D,j}$ $(1 \leq j \leq N_D)$, $R_D \ll M_D, N_D$ is the order of the decomposition, and ^{T} denotes matrix transposition [9].

As is well known, both left and right singular matrices U_D and V_D are column-orthonormal, i.e., $U_D^T U_D = V_D^T V_D = I_{R_D}$ (the identity matrix of order R_D). Thus, the column vectors of U_D and V_D each define an orthornormal basis for the space of dimension R_D spanned by the $u_{D,i}$'s and $v_{D,j}$'s. We refer to this space as the *latent domain space* $\mathcal{L}_D$. The (rank-R_D) decomposition (2) encapsulates a mapping between the set of words w_i and texts t_j and (after appropriate scaling by the singular values) the set of R_D-dimensional vectors $y_{D,i} = u_{D,i} S_D$ (for w_i) and $z_{D,j} = v_{D,j} S_D$ (for t_j).

The basic idea behind (2) is that the rank-R_D decomposition captures the major structural associations in W_D and ignores higher order effects. Hence, the relative positions of the input words in the space $\mathcal{L}_D$ reflect a parsimonious encoding of the semantic concepts used in the domain under consideration. This means that any new text mapped onto a vector "close" (in some suitable metric) to a particular set of words can be expected to be closely related to the concept encapsulated by this set. Note that this approach is conceptually richer than

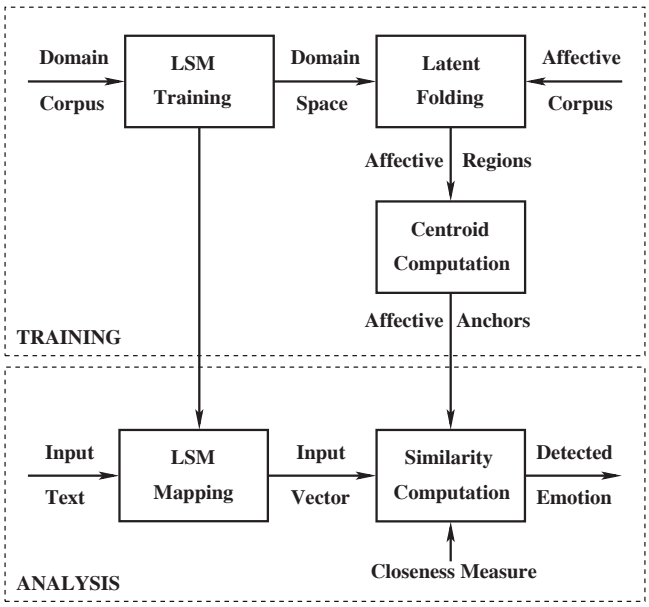

Fig. 2. *Affective Analysis Using Latent Affective Folding.*

methods relying solely on keywords, since it automatically leverages both co-occurrence analysis and dimensionality reduction.

Next, we exploit the affective corpus $\mathcal{T}_A$ to automatically derive, in a data-driven way, so-called *affective anchors* representative of emotional categories in this domain space (which corresponds to the *latent affective processing* block on Fig. 1). This derivation depends on the specific mapping instantiation considered: two alternatives are discussed in Sections III and IV. Finally, once the affective anchors are computed, it remains to compare the representation of a given input text to each of these anchors, which leads to a quantitative assessment for the overall affective affinity of the text.

III. Latent Affective Folding

Taking into account the two separate phases of training and analysis, the first mapping instantiation, latent affective folding, proceeds as illustrated in Fig. 2. First, we generate a suitable representation in the LSM space $\mathcal{L}_D$ of the texts in $\mathcal{T}_A$. Although these texts were not seen in the training corpus $\mathcal{T}_D$, we can invoke the same "folding" technique as used in standard latent semantic analysis (LSA): see, e.g., [9] and the references therein. Let us call $t_{A,q}$ $(1 \leq q \leq N_A)$ each entry in $\mathcal{T}_A$. Treating it as a regular pseudo-document,[1] we compute for this text the weighted counts (1) with $j = q$. The resulting feature vector, a column vector of dimension M_D, can be thought of as an additional column of the matrix W_D. Assuming the matrices U_D and S_D do not change appreciably, the SVD expansion (2) therefore implies:

$$t_{A,q} \simeq U_D\, S_D\, v_{A,q}^T\,, \tag{3}$$

[1]This entails, in particular, using the values ε_i observed during training for the words of $t_{A,q}$ that are part of $\mathcal{V}_D$. Words not present in $\mathcal{V}_D$ are ignored.

where the R_D-dimensional vector $v_{A,q}^T$ acts as an additional column of the matrix V_D^T. Thus, the represention of this text in the domain space can be obtained from $\nu_{A,q} = v_{A,q} S_D$. On Fig. 2, this process is called *latent folding*. The resulting set of vectors $\nu_{A,q}$ $(1 \leq q \leq N_A)$ can be viewed as functionally equivalent to the set of vectors $z_{D,j}$ $(1 \leq j \leq N_D)$, albeit for affective rather than domain data.

Now let L be the number of affective categories considered. We assume that human annotators have grouped all N_A texts in $\mathcal{T}_A$ into L subsets $\mathcal{T}_A^{(\ell)}$, one for each distinct emotion.[2] For each $1 \leq \ell \leq L$, the representation of the subset $\mathcal{T}_A^{(\ell)}$ in the space $\mathcal{L}_D$ is therefore prototypical of that particular emotion.

This forms the basis for computing the region associated with each affective category ℓ in the domain space. From all documents that have been labeled with the ℓth emotion, i.e., the subset $\mathcal{T}_A^{(\ell)}$, we gather the relevant representations $\nu_{A,q}$ in the domain space. We then compute the centroid:

$$\hat{z}_{D,\ell} = \frac{1}{|\mathcal{T}_A^{(\ell)}|} \sum_{\mathcal{T}_A^{(\ell)}} \nu_{A,q} \,, \tag{4}$$

as the average of the associated text vectors for each pertinent region in $\mathcal{L}_D$. This is turn defines the affective anchor representing each emotion ℓ $(1 \leq \ell \leq L)$ in the domain space. On Fig. 2, this process is labelled *centroid computation*.

The notation $\hat{z}_{D,\ell}$ is chosen to underscore the connection with $z_{D,j}$: in essence, $\hat{z}_{D,\ell}$ represents the (fictitious) text in the domain space that would be perfectly aligned with emotion ℓ, had it been seen the training collection $\mathcal{T}_D$.

IV. Latent Affective Embedding

A potential drawback of the above implementation is that (4) is patently sensitive to the distribution of words within $\mathcal{T}_A$, which may be quite different from the distribution of words within $\mathcal{T}_D$. In such a case, "folding in" the affective texts as described above may well introduce a bias in the position of the anchors in the domain space, which in turn could result in an inability to satisfactorily resolve subtle distinctions between emotional connotations. To remedy this situation, a possible solution is to build a separate LSM space from the affective training data. This leads to the latent affective embedding procedure illustrated in Fig. 3.

Referring back to the L subsets $\mathcal{T}_A^{(\ell)}$, we first generate a meta-corpus comprising only L documents, where the ℓth document is the concatenation of all documents in $\mathcal{T}_A^{(\ell)}$. From this meta-corpus, we then construct a $(M_A \times L)$ matrix W_A, whose elements $w'_{p,\ell}$ suitably reflect the extent to which each word or expression $w'_p \in \mathcal{V}_A$ appeared in each affective category c_ℓ, $1 \leq \ell \leq L$. This leads to the same form as (1), albeit with domain texts replaced by affective categories.

We then perform the SVD of W_A in a similar vein as (2):

$$W_A \simeq U_A \, S_A \, V_A^T \,, \tag{5}$$

[2]Note that, in the case of multiple emotional annotations, each text could conceivably contribute to several such subsets.

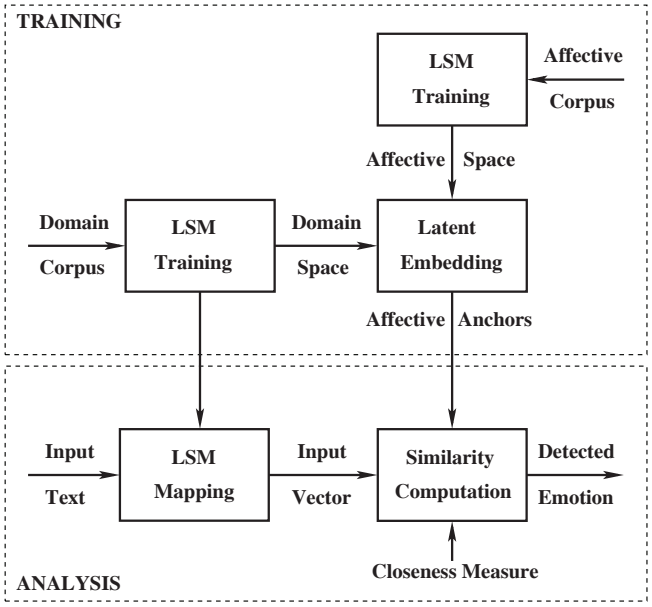

Fig. 3. *Affective Analysis Using Latent Affective Embedding.*

where again all definitions are analogous. This yields the second *LSM training* block depicted in Fig. 3. As before, both left and right singular matrices U_A and V_A are column-orthonormal, and their column vectors each define an orthornormal basis for the space of dimension R_A spanned by the $u_{A,p}$'s and $v_{A,\ell}$'s. We refer to this space as the *latent affective space* $\mathcal{L}_A$. The (rank-R_A) decomposition (5) encapsulates a mapping between the set of words w'_p and categories c_ℓ and (after appropriate scaling by the singular values) the set of R_A-dimensional vectors $y_{A,p} = u_{A,p} S_A$ and $z_{A,\ell} = v_{A,\ell} S_A$.

Thus, each vector $z_{A,\ell}$ can be viewed as the centroid of an emotion in $\mathcal{L}_A$, or, said another way, an affective anchor in the affective space. Since their relative positions reflect a parsimonious encoding of the affective annotations observed in the emotion corpus, these affective anchors now properly take into account any accidental skew in the distribution of words which contribute to them. All that remains to do is map them back to the domain space.

This is done on the basis of entities that are common to both the affective space and the domain space. Let $\mathcal{V}_{DA}$, $|\mathcal{V}_{DA}| = M_{DA}$, represent the intersection between $\mathcal{V}_D$ and $\mathcal{V}_A$. We denote their representations in $\mathcal{L}_D$ and $\mathcal{L}_A$ by $\lambda_{D,k}$ and $\lambda_{A,k}$, respectively $(1 \leq k \leq M_{DA})$.

The first step in deriving the cross-space transformation is to temporarily map both of these vectors to the unit sphere. Let μ_D, μ_A and Σ_D, Σ_A denote the mean vector and covariance matrix for all observations $\lambda_{D,k}$ and $\lambda_{A,k}$ in the two spaces, respectively. We first transform each vector as:

$$\bar{\lambda}_{D,k} = \Sigma_D^{-1/2} \left(\lambda_{D,k} - \mu_D \right) , \tag{6}$$

$$\bar{\lambda}_{A,k} = \Sigma_A^{-1/2} \left(\lambda_{A,k} - \mu_A \right) , \tag{7}$$

so that the resulting sets $\{\bar{\lambda}_{D,k}\}$ and $\{\bar{\lambda}_{A,k}\}$ each have zero mean and identity covariance matrix.

For this purpose, the inverse square root of each covariance matrix can be obtained as:

$$\Sigma^{-1/2} = Q\Lambda^{-1/2}Q^T\,, \tag{8}$$

where Q is the eigenvector matrix of the covariance matrix Σ, and Λ is the diagonal matrix of corresponding eigenvalues. This applies to both domain and affective data.

The next step is to derive the $(R_D \times R_A)$ cross-space transformation matrix Γ such that:

$$\bar{\lambda}_{D,k} = \Gamma\,\bar{\lambda}_{A,k}\,. \tag{9}$$

Details on how to proceed are provided in the Appendix.

Once the transformation Γ is computed, we need to apply it to the centroids of the affective categories in the affective space, so as to map them to the domain space. On Fig. 3, this process is referred to as *latent embedding*. We first project each vector $z_{A,\ell}$ $(1 \leq \ell \leq L)$ into the unit sphere, resulting in:

$$\bar{z}_{A,\ell} = \Sigma_A^{-1/2}\,(z_{A,\ell} - \mu_A)\,, \tag{10}$$

as prescribed in (7). We then synthesize from $\bar{z}_{A,\ell}$ a unit sphere vector corresponding to the estimate in the normalized domain space. From the foregoing, this estimate is given by:

$$\hat{\bar{z}}_{D,\ell} = \Gamma\,\bar{z}_{A,\ell}\,. \tag{11}$$

Finally, we restore the resulting contribution at the appropriate place in the domain space, by reversing the transformation (6):

$$\hat{z}_{D,\ell} = \Sigma_D^{1/2}\,\hat{\bar{z}}_{D,\ell} + \mu_D\,. \tag{12}$$

Combining the three steps (10)–(12) together, the overall mapping can be written as:

$$\hat{z}_{D,\ell} \;=\; (\Sigma_D^{1/2}\Gamma\Sigma_A^{-1/2})\,z_{A,\ell} \;+\; (\mu_D - \Sigma_D^{1/2}\Gamma\Sigma_A^{-1/2}\mu_A)\,. \tag{13}$$

This expression stipulates how to leverage the *observed* affective anchors $z_{A,\ell}$ in the affective space to obtain an estimate of the *unobserved* affective anchors $\hat{z}_{D,\ell}$ in the domain space, for $1 \leq \ell \leq L$. The overall procedure is illustrated in Fig. 4 (in the simple case of two dimensions).

V. Sentiment Prediction

To summarize, using either latent affective folding or latent affective embedding, we end up with an estimate $\hat{z}_{D,\ell}$ of the affective anchor for each category ℓ in the domain space $\mathcal{L}_D$. What remains to be described is how to perform sentiment prediction in that space.

The first step is to map into the space $\mathcal{L}_D$ any new input text to be processed. Proceeding along the lines of (3), each new text t is again treated as a pseudo-document, leading to:

$$t = U_D\,S_D\,v^T\,, \tag{14}$$

where, as before, the R_D-dimensional vector v^T acts as an additional column of the matrix V_D^T. The represention of the new text in the domain space is therefore obtained from $z = vS_D$. On Figs. 2 and 3, this process is called *LSM mapping*.

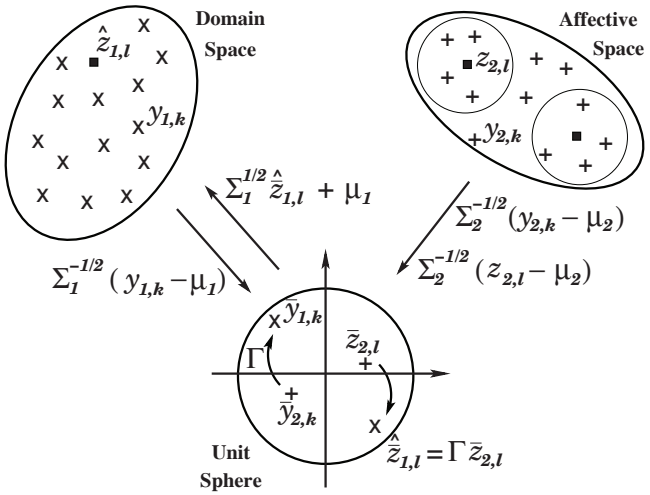

Fig. 4. *Illustration of Affective Anchor Embedding (2-D Case).*

The second step is to compare this representation to each affective anchor $\hat{z}_{D,\ell}$, bringing to bear a closeness measure consistent with the LSM paradigm. From [9], [11], among others, a natural metric to consider is the cosine of the angle between the two vectors. This yields:

$$\mathcal{C}(z, \hat{z}_{D,\ell}) = \frac{z\,\hat{z}_{D,\ell}^T}{\|z\|\,\|\hat{z}_{D,\ell}\|}\,, \tag{15}$$

for any $1 \leq \ell \leq L$. Using (15), it is a simple matter to directly compute the relevance of the input text to each affective category. Referring back to Figs. 2 and 3 one last time, this process is known as *similarity computation*. In essence, the resulting set of relevance scores (one for each affective anchor) leads to a quantitative assessment for the overall affective affinity of the text. It is important to note that in this assessment any word weighting is now implicitly taken into account by the LSM formalism.

VI. Experimental Evaluation

A. Test Database

The "Affective Text" task from SemEval 2007 was focused on the emotion classification of news headlines [4]. Headlines normally consist only of a few words and are often written by creative people with the intention to "provoke" emotions, and consequently attract the readers' attention. These characteristics make this kind of data particularly suitable for use in an automatic sentiment prediction setting, as a variety of emotional overtones tend to be present despite the brevity of each headline.

The test data accordingly consisted of 1,250 news headlines extracted from news web sites (such as Google news, CNN) and/or newspapers, and annotated along Ekman's standard $L = 6$ "universal" emotions (ANGER, DISGUST, FEAR, JOY, SADNESS, and SURPRISE [12]) by 6 different annotators. The

TABLE I
DISTRIBUTION OF EMOTIONAL MASS IN SEMEVAL-2007 "AFFECTIVE TEXT" TEST CORPUS.

Possible Emotion	Overall Probability of Occurrence
ANGER	11.0 %
DISGUST	7.0 %
FEAR	18.6 %
JOY	21.7 %
SADNESS	22.0 %
SURPRISE	19.7 %

TABLE II
SENTIMENT PREDICTION RESULTS ON SEMEVAL-2007 "AFFECTIVE TEXT" TEST CORPUS.

Approach Considered	Sentiment Recognition Rate
Baseline Word Accumulation	63.9 %
LSA (Specific Word Only)	67.5 %
LSA (With WordNet Synset)	68.2 %
LSA (With All WordNet Synsets)	67.8 %
Latent Affective Folding	70.3 %
Latent Affective Embedding	74.6 %

reader is referred to [4] for detailed information on data annotation, including studies on inter-annotator agreement. Taking into account the different degrees of emotional load assigned to each headline, the overall distribution of "emotional mass" observed across the entire test data is reported in Table I.

B. Baseline Systems

For baseline purposes, we selected two different kinds of systems: (i) based entirely upon manually selected vocabulary as in [7], and (ii) based on standard LSA trained on a large corpus of generic texts as in [13]. For (i), we followed a simple word accumulation strategy, which annotates the emotions in a text based on the presence of words from the WordNet-Affect lexicon [14].

For (ii), we implemented three conventional LSA-based systems, which differ solely in the way each emotion is represented in the generic LSA space: either based on a specific word only (e.g., JOY), or the word plus its WordNet synset, or the word plus all WordNet synsets labelled with that emotion in WordNet-Affect [4]. In all three cases, the underlying LSA space was based on the Wall Street Journal text collection, comprising about 86,000 articles.

C. Latent Affective Training

For the latent affective framework, we needed to select two separate training corpora. For the "domain" corpus, we selected a collection of about $N_D = 8,500$ relatively short English sentences (with a vocabulary of roughly $M_D = 12,000$ words) originally compiled for the purpose of a building a concatenative text-to-speech voice. Though not completely congruent with news headlines, we felt that the type and range of topics covered was close enough to serve as a good proxy for the domain.

For the "affective" corpus, we relied on about $N_A = 5,000$ mood-annotated blog entries from LiveJournal.com, with a filtered[3] vocabulary of about $M_A = 20,000$ words. The indication of mood being explicitly specified when posting on LiveJournal, without particular coercion from the interface, mood-annotated posts are likely to reflect the true mood of the blog authors [14]. The moods were then mapped to the $L = 6$ categories adopted in the affective description.

[3]Extensive text pre-processing is typically required on blog entries, to address typos and assorted creative license.

Finally, we formed the domain and affective matrices W_D and W_A and processed them as in (2) and (5). We used $R_D = 100$ for the dimension of the domain space $\mathcal{L}_D$ and $R_A = L = 6$ for the dimension of the affective space $\mathcal{L}_A$, and computed the affective anchors as per (4) and (13), respectively.

D. Experimental Results

Recall from [2]–[3] that emotion detection results obtained with the above setup, while encouraging, were not entirely satisfactory. Although the two latent affective mapping techniques provided significant improvements in performance compared to all four baseline approaches, the best F-measures observed were in the 30-to-35 range, which is arguably too low to assign a specific emotion with enough confidence.[4] Part of the problem is that all systems based on latent semantics tend to achieve high recall but fairly low precision, which is likely due to the fact that LSA/LSM is good at handling synonyms, but known to be otherwise hindered by polysemy [9], [15].

On the other hand, as we argued earlier, it may not be necessary to precisely identify the exact emotion in order to achieve satisfactory emotional congruence in TTS. Since what matters most is to avoid synthesizing speech with the wrong emotional quality, it may be enough to conform to the overall polarity of the input. After all, only a small percentage of input material may plausibly lead to salient congruence problems in the first place, primarily because they sit at the extreme of the affective range. In such cases sentiment analysis would be sufficient to enable the correct course of action.

To assess how well we could predict the most likely sentiment associated with each news headline in the test data, we thus adopted sentiment recognition rate as the evaluation criterion. The results are summarized in Table II for the various affective descriptions detailed above. We observe that the two latent affective techniques provide significant ($p < 0.001$ using the Wilcoxon test) reductions in sentiment error rate compared to all four baseline approaches. Moreover, it appears the techniques can be confidently applied to a large fraction of the input material.

[4]In terms of individual emotions, we observed the same general behavior as reported in [14], in that JOY yields the highest classification performance, followed by FEAR and SADNESS, then ANGER and SURPRISE, and finally DISGUST, which appears to be the most difficult to classify. Of course, the latter may be partly disadvantaged by its relative under-representation in the data (cf. Table I), which entails comparatively coarse scores within this emotional category.

Of the two latent affective implementations, latent embedding performs better, presumably because the embedded affective anchors are less sensitive than the folded affective anchors to any difference in word distributions that may exist between the two corpora. Both techniques seem to benefit from a richer description of the affective space, with positive repercussions on ensuing sentiment prediction.

VII. Conclusion

We have proposed a fully data-driven strategy for sentiment analysis in text. This strategy articulates around two coupled phases: (i) separately encapsulate both the foundations of the application domain and the overall affective fabric of the language, and (ii) exploit the emergent relationship between these two levels of semantic description in order to inform the sentiment prediction process. We address (i) by leveraging the latent topicality of two distinct corpora, as uncovered by a global LSM analysis of domain-oriented and emotion-oriented training documents. The two descriptions are then superimposed to produce the desired connection between all terms and affective categories. Because this connection automatically takes into account the influence of the entire training corpora, it is more encompassing than that based on the relatively few affective terms typically considered in conventional processing.

Empirical evidence gathered on the "Affective Text" portion of the SemEval-2007 corpus [4] confirms the effectiveness of the proposed strategy. Sentiment recognition performance with latent affective embedding is slightly better than with latent affective folding, presumably because of its ability to more richly describe the affective space. Both techniques outperform affectively weighted word accumulation, as well as standard LSA approaches based on expert knowledge of emotional keywords. Latent affective mapping thus appears to be a promising solution for automatic sentiment prediction, which bodes well as a first step in ensuring emotional congruence in TTS.

References

[1] J.R. Bellegarda, "Toward Naturally Expressive Speech Synthesis: Data–Driven Emotion Detection Using Latent Affective Analysis," in *Proc. 7h ISCA Speech Synthesis Workshop*, Kyoto, Japan, pp. 200–205, 2010.

[2] J.R. Bellegarda, "Latent Affective Mapping: A Novel Framework for the Data–Driven Analysis of Emotion in Text," in *Proc. Interspeech*, Makuhari, Japan, pp. 1117–1120, 2010.

[3] J.R. Bellegarda, "Further Analysis of Latent Affective Mapping for Naturally Expressive Speech Synthesis," in *Proc. ICASSP*, Prague, Czech Republic, pp. 5356–5359, 2011.

[4] C. Strapparava and R. Mihalcea, "SemEval-2007 Task 14: Affective Text," in *Proc. 4th Int. Workshop on Semantic Evaluations (SemEval 2007)*, Prague, Czech Republic, pp. 70–74, 2007.

[5] M. Schröder, "Approaches to Emotional Expressivity in Synthetic Speech," in *The Emotion in the Human Voice*, Vol. 3, K. Izdebski, Ed., San Diego, CA: Plural, 2008.

[6] M. Schröder, "Expressing Degree of Activation in Synthetic Speech," *IEEE Trans. Audio, Speech and Language Processing*, Vol. 14, No. 4, pp. 1128–1136, 2006.

[7] C.M. Whissell, "The Dictionary of Affect in Language," in *Emotion: Theory, Research, and Experience*, R. Plutchik and H. Kellerman, Eds., New York, NY: Academic Press, pp. 13–131, 1989.

[8] C. Ovesdotter Alm, D. Roth, and R. Sproat, "Emotions from Text: Machine Learning for Text–Based Emotion Prediction," in *Proc. Conf. HLT–EMNLP*, Vancouver, BC, pp. 579–586, 2005.

[9] J.R. Bellegarda, *Latent Semantic Mapping: Principles & Applications*, Synthesis Lectures on Speech and Audio Processing Series, Fort Collins, CO: Morgan & Claypool, 2008.

[10] M.W. Berry, "Large–Scale Sparse Singular Value Computations," *Int. J. Supercomp. Appl.*, **6**(1):13–49, 1992.

[11] C.D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, Cambridge, UK: Cambridge University Press, 2008.

[12] P. Ekman, "Facial Expression and Emotion", *American Psychologist*, **48**(4):384–392, 1993.

[13] C. Strapparava, A. Valitutti, and O. Stock, "The Affective Weight of Lexicon," in *Proc. 5th Int. Conf. Language Resources and Evaluation (LREC)*, Lisbon, Portugal, 2006.

[14] C. Strapparava and R. Mihalcea, "Learning to Identify Emotions in Text," in *Proc. 2008 ACM Symposium on Applied Computing*, New York, NY, pp. 1556–1560, 2008.

[15] S.M Kim, A. Valitutti, and R.A. Calvo, "Evaluation of Unsupervised Emotion Models to Textual Affect Recognition," in *Proc. Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, Los Angeles, CA, pp. 62–70, 2010.

[16] J.R. Bellegarda, P.V. de Souza, A. Nadas, D. Nahamoo, M.A. Picheny, and L.R. Bahl, "The Metamorphic Algorithm: A Speaker Mapping Approach to Data Augmentation," *IEEE Trans. Speech and Audio Processing*, **2**(3):413–420, 1994.

Appendix

Computation of the Transformation Γ

The cross-space transformation Γ in (9) is derived from a relative measure of how latent domain and latent affective spaces are correlated with each other, as accumulated on a common word basis. Since $\min(R_D, R_A) = R_A$, it is necessary to first project each $\bar{\lambda}_{D,k}$ in (6) into the unit sphere of same dimension as $\bar{\lambda}_{A,k}$ in (7). This assumes that the $(R_A \times R_D)$ associated projection matrix, P, has been properly designed so as to minimize any information loss.

Once this is done, we compute the (normalized) cross-covariance matrix between the two unit sphere representations, specified as:

$$K_{DA} = \sum_{k=1}^{M_{DA}} (P\bar{\lambda}_{D,k})\, \bar{\lambda}_{A,k}^{T}\,. \tag{16}$$

Note that K_{DA} is typically full rank as long as $M_{DA} > R_A^2$. Performing the (full-rank) SVD of K_{DA} yields the expression:

$$K_{DA} = \Phi\, \Omega\, \Psi^{T}\,, \tag{17}$$

where as before Ω is the diagonal matrix of singular values, and Φ and Ψ are both unitary in the unit sphere of dimension R_A. This in turn leads to the entity:

$$\Xi = \Phi \Psi^{T}\,, \tag{18}$$

which can be shown (see, for example, [16]) to represent the $(R_A \times R_A)$ least squares rotation matrix that must be applied (in that unit sphere) to $\bar{\lambda}_{A,k}$ to obtain an estimate of $P\bar{\lambda}_{D,k}$. It then remains to pre-multiply by P^T to get back to the unit sphere of dimension R_D, i.e.:

$$\Gamma = P^{T}\Xi\,, \tag{19}$$

is the $(R_D \times R_A)$ transformation matrix sought.