Sentence-based Sentiment Analysis for Expressive Text-to-Speech

DRAFT

*Alexandre Trilla, Francesc Alías, Member, IEEE

Abstract—Current research to improve state of the art Text-To-Speech (TTS) synthesis studies both the processing of input text and the ability to render natural expressive speech. Focusing on the former as a front-end task in the production of synthetic speech, this article investigates the proper adaptation of a Sentiment Analysis procedure (positive/neutral/negative) that can then be used as an input feature for expressive speech synthesis. To this end, we evaluate different combinations of textual features and classifiers to determine the most appropriate adaptation procedure. The effectiveness of this scheme for Sentiment Analysis is evaluated using the Semeval 2007 dataset and a Twitter corpus, for their affective nature and their granularity at the sentence level, which is appropriate for an expressive TTS scenario. The experiments conducted validate the proposed procedure with respect to the state of the art for Sentiment Analysis.

Index Terms—Sentiment analysis, feature engineering, text classification, expressive TTS synthesis

Copyright (c) 2012 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org

The authors are with the Grup de Recerca en Tecnologies Mèdia, La Salle, Universitat Ramon Llull, Quatre Camins 2, 08022 Barcelona, Spain (e-mail: atrilla@salle.url.edu; falias@salle.url.edu).

EDICS Category: SLP-LANG, SLP-UNDE, SPE-SYNT

I. Introduction

SPEECH researchers are increasingly focusing on the full range and variation of speech in order to signal the social and psychological aspects of a message. This means studying not just the conventional propositional or linguistic content but also affective states [1]. Future natural conversational interfaces, in line with the present needs of conversational interaction and everyday speech [2], [3], are hoped to achieve a major breakthrough in usability by integrating such expressive speech [4], in both the analysis and the synthesis of conversations. Therefore, the new generation of Text-To-Speech (TTS) systems should automatically deliver expressive cues when synthesising an affective message [5], [6]. Such niche of research can focus on "how" to render expression in speech or on "when" to do so [7]. This work is focused on the latter, since the detection and classification of the expression present in textual input is the requisite first step in the fully automatic generation of naturally expressive synthetic speech [5], [6], [8].

The basis of expressiveness in text and speech is quite diverse, non uniform and even overlapping [9], [10]. Some researchers in TTS tend to relate expression in speech with

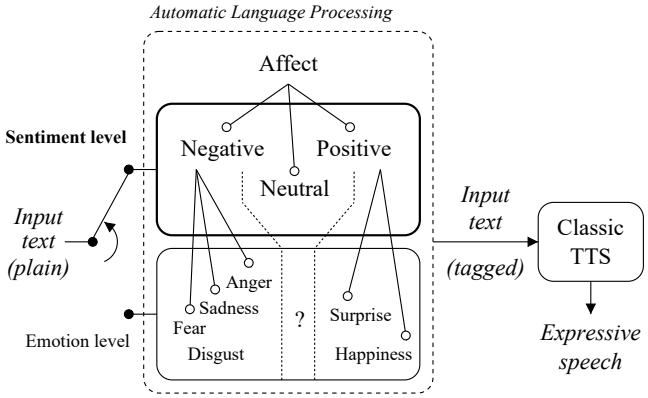


Fig. 1. Framework of the proposed expressive TTS system following [5], for automatically detecting textual affect-related information represented in the hierarchy of affect [19].

domain (i.e., the topic) in text [5], while others prefer to propose a direct bond with affect [11]–[16], and yet some others consider that both domain and affect are only disjoint subsets of expression [6], [7]. The adequacy of these different approaches might seem to rely on the field of application, e.g.: domain-based expressiveness was used in an advertising scenario [5], while affect-based expressiveness was used in news reading [6], [7], [11], [15], [16] and storytelling [12]. In this regard, the sole consideration of the domain as a reliable proxy for textual expression is more of a specific and partial solution to the problem [5], which leads to domain-transfer problems when a new domain is tested [17]. In contrast, accounting for the interaction between affect and text regardless of the domain seems to be more adequate for conversational interfaces [6], [15], [18], where a single conversation is usually composed of different topics.

Currently there are many challenges in translating human affect into explicit representations. One option is to presuppose the existence of some suitable taxonomy of affective states [6]. Such taxonomy typically represents the graded nature of affective categories, which is generally described in two levels of detail according to the hierarchy of affect [9], [19], i.e., sentiment and emotion levels, see Figure 1. Nevertheless, there is a lack of consensual definition of the different qualitative types (and degrees) of affective states to be considered [20]. For instance, in expressive TTS, text-based prediction of affect was tackled in [11] aiming to distinguish polarity levels only between "anger" and "happiness" emotions, while in [13] and [6] the focus was on detecting a broader range of prototypical emotions [19], typically named "The Big Six" [21] (happy, sad, afraid, angry, surprised and disgusted). However, the

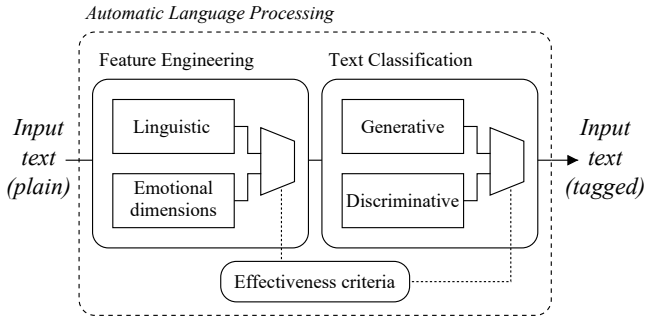


Fig. 2. Overview of the Sentiment Analysis framework under study, which considers both the diversity in the nature of the features extracted from the text and the diversity in the learning principles of the classifiers, and selects the most effective system for the problem at hand.

direct expression of these six stereotypical emotions per se is surprisingly unusual in conversational interactions [1]. Then, over this low level of emotion centred on the leaves of the hierarchy of affect (see Figure 1), the main meaningful distinction is the one between overall positive and negative expressions. This two-class categorisation problem (positive vs. negative) has traditionally been referred to as Sentiment Analysis (SA) by the Natural Language Processing (NLP) research community, see [18] for a comprehensive review. There are some authors who also consider a neutral sentiment at the same level of the hierarchy [19]. TTS applications make use of this neutral sentiment to generate usual messages that are not positive nor negative [15], [16], e.g., when reporting objective information such as a description.

The present work focuses on the categorisation of a plain input text to inform a TTS system about the most appropriate sentiment (positive, negative and neutral) to automatically synthesise expressive speech at the sentence level. Following [15], [16], we develop a SA approach adapted to the requirements of the problem at hand. We evaluate our approach on two English corpora labelled with sentiment on a sentence-by-sentence basis: the Semeval 2007 dataset [22] and a subset of the Twitter corpus [23]. We validate our proposal with respect to the state of the art and with SA problems with a somewhat different nature, respectively.

The paper is organised as follows. Section II refers to other works related to processing the affect of input text. Section III describes the procedure that is proposed to adapt the SA task to the TTS scenario. Section IV describes the experiments and analyses the results obtained. Finally, Section V draws conclusions and discusses future work.

II. Related Work

The prediction of affect in text is a topic that is mainly related to NLP, but it has also attracted the attention of the TTS synthesis research community. As far as we know, this work is one of the first attempts to adapt conventional SA methods to the TTS requirements.

In the TTS scenario, the granularity of the text under analysis is usually determined to be the sentence, as sentences are sensibly short textual representations with a rich expressive content [5], [11]–[13], [15], [16]. By regarding this sentence-by-sentence basis, natural expressive variations can be considered within the same paragraph [12].

Moreover, the conventional SA solutions borrowed from the NLP scenario may need to be adapted to the TTS environment because they are usually set to work with compilations of long texts that are not analysed at sentence-level [18], [24]. Some previous work has tackled this short text setting with heuristic approaches by affectively weighting the lexicon and then spotting keywords in the sentences, e.g., see [11], [16], [25], [26]. Other work, instead, proposed using Machine Learning (ML) methods to directly learn from previous example sentences [5], [6], [23], [27]–[29]. It is worth noting that previous work observed that the latter methods performed more effectively for the problem at hand [15], [22]. Therefore, this work focuses on direct ML methods.

Given that the information provided by a sentence is rather reduced, some approaches based on the latter ML methods also proposed using additional texts to infer further links with affect [6], [22]. Other works, instead, delved into the relevant characteristics of the available text of analysis without enlarging the data to process [5], [30]. In a TTS environment, which is expected to perform in real time, the SA task shall not overburden the TTS conversion process. What is more, collecting useful text for the problem at hand is difficult as it requires many human evaluators [22]. Due to resource limitations, experiments are restricted to existing labelled corpora. Hence, this work focuses on exploiting only the available short text of analysis. In any case, though, a comprehensive study of the size of the corpus and its impact on the computational performance is left for future works.

III. Sentiment Analysis for TTS purposes

A. Framework

This section focuses on reviewing and gathering a set of features suited to the addressed sentiment classification problem at hand. Firstly, different common features that are of use to denote the affect in text are described. In order to obtain them, we have developed EmoLib[1], which implements the framework depicted in Figure 2, but following a pipeline design pattern (see Figure 3) due to inter-feature dependencies and tagging sequentiality. EmoLib is a flexible framework for building prototypes that allows studying the appropriateness of different strategies to label affect in text [15], [16]. It allows incorporating offline expert knowledge derived from psychological studies as well as knowledge learnt from training examples. What follows is the description of the modules that build up the processing chain of EmoLib:

1) Lexical analyser: converts the plain input text into an output token stream. This module is produced with the JavaCC[2] parser generator. Additionally, this module spots the possible affective containers (content words), valence shifters such as negation words and intensifiers [18] and filters out "stop words" like function words [31].

[1] http://dtminredis.housing.salle.url.edu:8080/EmoLib/
[2] http://javacc.java.net/

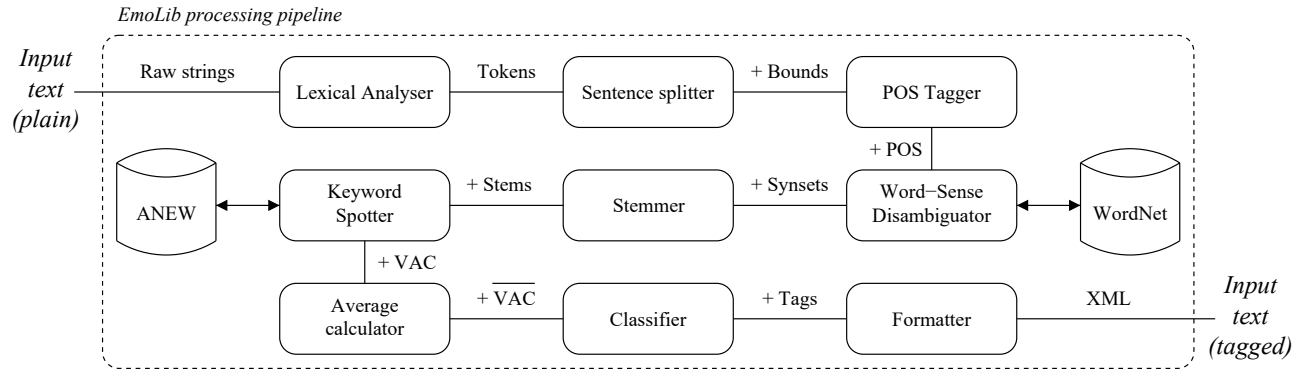


Fig. 3. Modular framework of the EmoLib processing pipeline. It provides the POS tag, synonyms, stems, emotional dimensions and predicted sentiment label to the tokens extracted form the plain input text. ANEW, which stands for "Affective Norms for English Words", is the dictionary of affect of use, and VAC, which stands for "Valence, Activation and Control", represents the emotional dimensions.

2) Sentence splitter: delimits the sentences through a binary decision tree following [32]. In general, periods, uppercase letters, exclamation points and question marks are good indicators of sentence boundaries.

3) POS Tagger: determines the function of nouns, verbs and adjectives (classes of words with a possible affective content [18]) within the sentence. A statistical approach is implemented using the Stanford log-linear POS tagger [33].

4) Word-Sense Disambiguator: resolves the meaning of affective words (i.e., nouns, adjectives and verbs) according to their context [31]. It uses a semantic similarity measure to score the senses of an affective word with the context words using the WordNet ontology [34]. Additionally, the module retrieves the set of synonyms for the resulting sense in order to expand the feature space [35].

5) Stemmer: removes the inflection of words for indexing purposes using the Porter stemming algorithm [36]. Semantically related words should map to the same stem, base or root form, and this should compensate for data sparseness.

6) Keyword Spotter: provides the emotional dimensions to the emotional words using the ANEW dictionary of affect [37]. It considers both the stems of the lexical instances as well as their POS tags. Words are mapped into a tridimensional space [9], also known as the circumplex, which defines the Valence (positive/negative evaluation), Activation (stimulation of activity) and Control (submissiveness) features (VAC) of the affect that they convey [9], [11], [16].

7) Average calculator: computes the averaged emotional dimensions for the text of analysis. In the current work, this is the arithmetic mean of the dimensions at the sentence level (i.e., a centroid-based approach) [9], [11], [16].

8) Classifier: predicts the most appropriate sentiment label according to the features extracted from the terms observed in the text, which is usually taken for a bag of words. The details of this module are thoroughly described in Section III-B, which regards the determination and representation of the relevant features, and in Section III-C, which describes the learning principles of the classifiers.

9) Formatter: presents the results in a usable form, which follows a XML specification [10], ready to be used by the TTS system that follows. For instance, the module can output the Speech Synthesis Markup Language [38] or the Emotion Markup Language [39].

B. Dimensionality Reduction and Weighting

In sentence-level Text Classification, the relative importance of features is of great relevance. But using all the features together directly often increases the size of the feature space without providing much satisfactory power (sparseness problem) [35]. Hence, selecting and weighting the most relevant features raises the discriminating properties of the data, thus improving the classification effectiveness [27], [31], [35], [40]. These methods are described as follows:

1) Term Selection: reduces the dimensionality of the feature space by selecting the most relevant features, which means discarding the ones that do not contribute significantly to the classification task [27], [31], [35]. This improves both the effectiveness of the classifier as well as its computational performance given the fewer number of features to process (see Section IV-C). In this work, three global Term Selection strategies have been considered: Mutual Information (MI), Chi-square (χ^2) and Term-Frequency-based Selection (TFS) [35]. MI selects term-based features that are not uniformly distributed among the sentiment classes because they are informative of their class. χ^2 measures how much expected counts and observed counts deviate from each other, and TFS directly selects the most frequent features.

2) Term Weighting: raises the discriminating power of certain features without reducing the dimensionality of the feature space. This process is somewhat complementary to (and dependent on) Term Selection with respect to the criteria of use. A persistent question regarding the weighting of terms is their representation of presence versus frequency [18], [24], [35]. Although the frequency of terms seems to be more useful as it naturally encodes the presence of terms, the use of binary weights denoting term presence/absence has comparatively performed better in sentiment analysis [18]. In this work, binary weights are evaluated, as well as a couple of enhanced frequency-based weights: the Inverse Term Frequency (ITF) [5], see Eq. 1, which weighs each term according to its prominence within the sentence, and the Relevance Factor

(RF) [30], see Eq. 2, which weighs the relevance of a term regarding the rest of categories.

$$ITF_t = \log \frac{\sum_{t' \in T} tf_{t'}}{tf_t} \quad (1)$$

$$RF_{t,C} = \log(1 + tf_t) \log_2 \left(2 + \frac{tf_{t,C}}{\max(1, tf_{t,\bar{C}})}\right) \quad (2)$$

where t represents the term, tf represents its Term Frequency, T represents the vocabulary (total number of different terms) and C represents the category of analysis (likewise $\bar{C}$ represents any different category).

C. Machine Learning

Regarding the learning strategy of the text classifiers, generative models explain the data, and if the model is correct, they should yield the best possible classification effectiveness rates [41]. Nevertheless, since the form of the actual model is unknown and the training sample does not generally cover the whole feature space, instead of proposing an endless amount of possible approximate models, task-centric approaches based on discriminating classes are evaluated [35].

Polynomial linear models are proposed in this work for their simplicity over their (more complex) nonlinear polynomial counterparts. Nonlinear models have more parameters to fit on a limited amount of training data and they are more prone to make mistakes for small datasets (see [5] for an empirical evidence of this phenomenon for Text Classification (TC) in a TTS-synthesis scenario). Instead, linear models might be preferable to separate the bulk of the data [31]. And with the high dimensional spaces that are typically encountered in text processing applications, the likelihood of linear separability increases rapidly [35]. What follows is the description of several classifiers with different learning principles that are considered for the study:

1) Multinomial Naive Bayes (MNB): probabilistic generative approach that builds a language model assuming conditional independence among the features. In reality, this assumption does not hold for text data [24], but even though the probability estimates are of low quality because of this oversimplified model, its classification decisions are surprisingly good [35]. The MNB combines efficiency (optimal time performance) with good accuracy, hence it is often used as a baseline in TC and SA research [31], [35].

2) Associative Relational Network - Reduced (ARN-R): word co-occurrence network-based approach, which constructs a Vector Space Model (VSM) with a term selection method "on the fly" based on the observation of test features [5]. This term selection refinement is reported to improve the classical VSM for modest-size sentence-based data in a TTS environment [5]. Dense vectors representing the input text and the class are retrieved (no learning process is involved) and evaluated by the cosine similarity measure. The basic hypothesis in using the ARN-R for classification is the contiguity hypothesis, where terms in the same class form a contiguous region and regions of different classes do not overlap [35].

3) Latent Semantic Analysis (LSA): similar to the VSM, but builds a latent semantic space by computing the Singular Value Decomposition (SVD) of the term-class matrix obtained from the VSM (i.e., constructing a low-rank approximation with its principal eigenvectors) [35]. The cosine similarity between the class vectors and the query text vectors (obtained by adding the observed term vectors) is used to make decisions in the reduced latent space. LSA has been used for emotion classification in a TTS scenario [6] as well as in TC and SA [22], [31].

4) Multinomial Logistic Regression (MLR): probabilistic discriminative approach that fits a set of exponential functions via the Maximum A Posteriori estimation [42]. MLR obeys the maximum entropy principle, therefore it does not make any further assumption beyond what is directly observed in the training data. Moreover, it makes no assumptions about the relationships among the features, and so might potentially be more effective when conditional independence assumptions are not met [24], also overcoming the sparseness problem. MLR has been used for SA in TTS and TC environments [15], [18], [24]

5) Support Vector Machine (SVM): maximum-margin discriminative approach that searches the hyperplane (decision surface in feature space) that is maximally distant from the class-wise data points. Since SVM is a binary classifier, a multicategorisation strategy has to be considered to deal with the three sentiment classes. SVM has shown to be superior with respect to other methods in situations with limited but sufficient training data [18]. SVM has been used in TC scenarios [30], [31] as well as in SA [18], [24].

D. Implications of the affect in text

In general, it is the semantics which provide a great deal of information with respect to the affect in text [18]. For the problem at hand, two approaches need to be considered in this regard: plain unigrams alone or a full set of diverse features [18], [24].

On the one hand, the former approach essentially leads to modelling words, which are plausibly conceived to be the smallest meaningful units of affect [43]. Words alone, modelled as unigrams, are obtained from the lexical instance of the tokens. Their consideration in isolation constitutes a simple Bag-Of-Words (BOW) model, which does not account for the order of words appearing in a text [31]. This BOW model is sometimes regarded to lack useful information, especially dealing with short texts in TTS [5]. The alternative approach is to increase the number of features selecting multiword patterns that are particularly discriminative [35]. In this regard, bigrams (i.e., the ordered co-occurrence of two unigrams) are also considered in the bag of features [24]. Bigrams are also reported to be of help to grasp stylistic traits and structural information (i.e., syntactic) in the text [5], [18]. This is regarded to be an alternative way to incorporate context [24], and with the inclusion of POS tags, the analysis is added some grammatical value [18]. Nevertheless, higher order n-grams are discarded as they do not appear to contribute much to the identification of affect in the text [18].

TABLE I
PROPERTIES OF THE SEMEVAL 2007 DATASET IN TERMS OF INSTANCE AND FEATURE COUNTS.

Instance properties	Counts	
Total (sentences)	1250	
Positive	174	
Neutral	764	
Negative	312	
With repeated words	46	
Without stop words	4	
Average length	7.53	
Feature properties	Unigrams	Bigrams
Total (n-grams)	8115	6865
Vocabulary	4085	6251
Frequent (≥ 5)	226	14

TABLE II
PROPERTIES OF THE TWITTER DATASET IN TERMS OF INSTANCE AND FEATURE COUNTS.

Instance properties	Counts	
Total (sentences)	3990	
Positive	1990	
Negative	2000	
With repeated words	1444	
Without stop words	576	
Average length	13.79	
Feature properties	Unigrams	Bigrams
Total (n-grams)	50849	46859
Vocabulary	7340	29676
Frequent (≥ 5)	1118	1032

TABLE III
SELECTED EXAMPLES OF THE STUDIED CORPORA SHOWING THE DIFFERENT ANALYSIS SCENARIOS.

Semeval 2007 dataset
Positive "The sweet tune of an anniversary"
Neutral "Bad reasons to be good"
Negative "Bombers kill shoppers"
Twitter dataset
Positive "had an amaazing day running sushi shower beach uno on the beach fun"
Negative "i couldn bear to watch it and i thought the ua loss was embarrassing"

IV. EXPERIMENTS AND RESULTS

To determine the most effective EmoLib configuration to adapt the SA framework to a TTS scenario, the main dataset of use in this work is the Semeval 2007 for its convenience for the problem at hand: sentence-based analysis on three categories of sentiment [22]. In addition, this dataset is sensibly small and unbalanced, which is challenging for the performance of SA (note that related works in TC for TTS synthesis have already managed such characteristics with success [5]). In this work we also evaluate a subset of a Twitter corpus to validate our proposal for two categories of sentiment. Moreover, since the size of the Twitter dataset is greater, it allows us to study the impact of having more evaluations for SA on the short sentence-by-sentence basis.

A. Datasets

1) Semeval 2007 dataset: This dataset consists of a compilation of news headlines (taken for short sentences with less than 8 words on average) drawn from major newspapers. Its design criteria highlight its typically high load of affective content written in a style meant to attract the attention of the readers [22]. In addition, its short-text form is adequate to evaluate SA in a TTS scenario where a single label represents the whole sentence [5]. This corpus is distributed in two sets: one for trial (training with 250 headlines) and the other for testing (containing 1000 headlines). This uneven distribution of its data is attributed to the competition conditions it was designed for. Nevertheless, considering the whole corpus as a single set (therefore containing 1250 headlines) is more appropriate for the following experimentation [6], [16].

An overall description of the properties of the entire dataset is shown in Table I. Note that the number of sentences (i.e., corpus instances) with words appearing more than once in a single sentence is scarce in the corpus (46 sentences out of 1250 yield a rate of 3.68%), and this figure even drops more if stop words are filtered out (0.32%). This fact shows that differentiating between the presence/frequency representation of the features is of little relevance for this data: in either case, the information is almost the same (this is strictly true for the 99.68% of the sentences in this corpus).

It is also important to note the richness of the vocabulary extracted from the data. Half the total number of unigrams yields the size of the whole unigram set (4085 unigrams), and in the case of bigrams, these counts are more similar (6251 bigrams). Hence, on average, each term only appears twice at most in the whole corpus. This lack of frequent features puts an extra difficulty to the identification of sentiment and therefore supports the proposal of weighting and selecting the most relevant ones.

2) Twitter dataset: This dataset consists of a compilation of tweets (taken for sentences with less than 14 words on average). This dataset is a subset of a bigger Twitter corpus [23], where the tweets that share the lexicon and sentiment label (which is based on the observed emoticons) with the headlines in the Semeval 2007 dataset have been selected. Hence, a similar "high load of affective content" characteristic can be used to describe it. Its sentence-based form is also adequate in a TTS scenario, but the greater amount of instances permits the study of the current approach of SA with a greater amount of data.

An overall description of the properties of this dataset is shown in Table II. Note that in this dataset, the number of tweets with repeated words is rather considerable (1444 out of 3990), so the assessment of the presence or the frequency of terms is well differentiated.

With the Twitter dataset, the richness of the vocabulary is a rather reduced compared to the Semeval 2007. Note that the total amount of unigrams is almost 7 times bigger than the size of the vocabulary, so words are frequently used and repeated over the Twitter corpus. This defines an analysis scenario different from the Semeval 2007, where in addition to having less instances per feature, the classification framework has to deal with the sometimes confusing neutral sentiment category. Table III reflects these differences through selected examples.

B. Experimental analysis

To determine the most effective EmoLib configuration (features and classifier) to adapt the SA framework to a TTS scenario, the following strategies are evaluated. On the one hand, the features of use contrast two approaches [18]: 1) the sensible agglomeration of traits in the vector space of features that are reported to be useful for SA, e.g., unigrams, bigrams, POS tags, stems, synonyms, emotional dimensions and negation flags, and 2) the sole consideration of unigrams as only the essential traits of sentiment in text. All unigrams and bigrams are weighted, representing the plain lexical instances of the observed words. POS tags are encoded by appending the POS to the unigrams such that words like "die_NOUN" and "die_VERB" represent two different dimensions in the feature space. If stemming is considered, the stems of the words are represented in the unigrams. Synonyms augment the dimensionality of the feature space as if they were observed in the text of analysis. Similarly, emotional dimensions augment the feature space with the overall sentence-level evaluations of valence, activation and control. If no emotion-signalling keyword if found in the text, default dimensions corresponding to the neutral sentiment are used. Finally, if an odd number of negation adverbs are detected in the sentence, a negation flag is set in the feature space.

The specific implementation in EmoLib of the TC methods to be evaluated are described hereunder:

- MNB uses Manning's TC definition for discrete features (binary weights) [35] and the Weka's general-purpose NaiveBayesMultinomial with continuous features [44].
- ARN-R is implemented following [5].
- LSA uses the SVD implementation provided by LingPipe[3] to construct a latent semantic space [45].
- MLR uses the Stochastic Gradient Descent optimisation procedure provided by LingPipe [42].
- SVM uses the Weka's Sequential Minimum Optimisation with a linear kernel and pairwise classification [44].

In TC it is customary to make use of the F_1 measure [31], [35] to compute the classification effectiveness rate. This unweighted effectiveness measure is needed to even the importance of each class regardless of instance imbalances, which are especially present in the Semeval 2007 dataset, see Table I. For all F_1 comparisons evaluated hereafter, the ANOVA test is applied to determine the statistical significance of the results. In order to estimate the F_1 measure, a 10-fold cross-validation procedure with macroaveraging is used for the two datasets (maintaining the class distributions in each fold) [31], [35].

In addition, a train-test procedure is also performed on the Semeval 2007 dataset following its original evaluation conditions [22]. This is to compare the results obtained with the procedure proposed in this work with the ones reported in the state of the art. In this alternative setting, the models are trained with much fewer instances than in the cross-validation section, see Table IV.

It is to note that the trial part of the Semeval 2007 dataset has a critical imbalance of instances: the size of the negative class is more than nine times bigger that the size of the positive class. This imbalance is much more abrupt than when using the whole corpus, making it more difficult to predict the class with the least generality, i.e., the positive, which only has 19 sentences. Also note that the relation between the vocabulary size and the total size of unigrams and bigrams is much greater for the trial part only than for the whole corpus (as is used in the cross-validation evaluation), which means that words appear a great deal less in this train-test setting.

[3] http://alias-i.com/lingpipe/

TABLE IV
PROPERTIES OF THE SEMEVAL 2007 DATASET (TRIAL PART) IN TERMS OF INSTANCE AND FEATURE COUNTS.

Instance properties	Counts	
Total (sentences)	250	
Positive	19	
Neutral	57	
Negative	174	
With repeated words	5	
Without stop words	1	
Average length	7.55	
Feature properties	**Unigrams**	**Bigrams**
Total (n-grams)	1638	1388
Vocabulary	1114	1290
Frequent (≥ 5)	22	1

TABLE V
AVERAGE F_1 RESULTS WITH THE WHOLE SET OF FEATURES USING 10-FOLD CROSS-VALIDATION (MEAN ± STD). IT CONSIDERS UNIGRAMS, BIGRAMS, POS TAGS, STEMS, SYNONYMS, EMOTIONAL DIMENSIONS AND NEGATION FLAGS.

Semeval 2007 dataset

Classifier	Term Weighting		
	Binary	ITF	RF
MNB	52.20 ± 4.30	51.09 ± 3.93	53.72 ± 6.54
ARN-R	46.14 ± 6.33	42.56 ± 5.10	51.28 ± 3.82
LSA	35.45 ± 6.64	38.10 ± 6.24	36.44 ± 8.28
MLR	54.32 ± 6.43	53.58 ± 6.72	54.66 ± 5.14
SVM	58.12 ± 4.15	55.20 ± 5.16	54.67 ± 5.53

Twitter dataset

Classifier	Term Weighting		
	Binary	ITF	RF
MNB	70.05 ± 1.21	69.14 ± 0.97	69.81 ± 1.34
ARN-R	55.39 ± 1.93	56.31 ± 2.36	68.29 ± 2.37
LSA	52.72 ± 2.78	51.40 ± 2.81	56.88 ± 2.12
MLR	72.33 ± 1.51	71.59 ± 1.26	72.66 ± 1.52
SVM	72.76 ± 1.76	71.61 ± 1.71	69.09 ± 1.79

C. Experimental results and discussion

1) Cross-validation evaluation: Table V shows the results obtained with the whole set of features. From the perspective of the term weighting strategy, little improvements are observed. For a given classifier, all the different term weighting configurations yield a similar effectiveness rate. Exceptionally, the ARN-R classifier shows a significant improvement for RF with respect to Binary and especially to ITF ($p=0.0004$). A similar behaviour is observed for the SVM and Binary-weighted features, but without significance ($p=0.2740$).

From the perspective of the classification strategy, it is to note that there seem to be two groups of classifiers according to the overall classification rates: the successful ones, which include the MNB, the MLR and the SVM, and the unsuccessful ones, which include the ARN-R and the LSA. It seems

TABLE VI
AVERAGE F_1 RESULTS WITH PLAIN UNIGRAM FEATURES USING 10-FOLD CROSS-VALIDATION (MEAN ± STD).

Semeval 2007 dataset

Classifier	Term Weighting		
	Binary	ITF	RF
MNB	53.30 ± 7.05	53.91 ± 5.11	54.52 ± 5.70
ARN-R	44.27 ± 6.82	39.88 ± 5.30	51.53 ± 5.32
LSA	35.74 ± 8.86	36.95 ± 6.28	34.15 ± 6.46
MLR	52.60 ± 7.57	52.86 ± 7.08	52.23 ± 6.25
SVM	53.56 ± 4.89	54.48 ± 7.09	50.80 ± 6.19

Twitter dataset

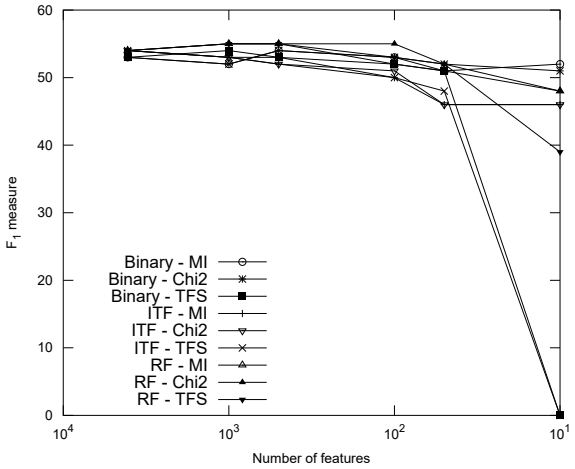
Classifier	Term Weighting		
	Binary	ITF	RF
MNB	71.14 ± 1.57	69.44 ± 1.53	70.65 ± 1.79
ARN-R	55.24 ± 2.31	57.80 ± 2.21	65.72 ± 2.38
LSA	53.87 ± 2.67	51.79 ± 2.65	56.87 ± 2.42
MLR	72.17 ± 1.76	71.37 ± 1.55	72.56 ± 1.82
SVM	70.64 ± 1.81	70.02 ± 1.88	69.32 ± 1.96

that for this textual data with a much larger size of features than examples, it is generally tricky to rely on the cosine similarity as a measure of relatedness (note that both the ARN-R and the LSA methods do it), regardless of the term-feature space of representation (it is especially adverse for the reduced space based on the principal components that the LSA method provides). Regarding the group of successful classifiers, it is to note that they all behave similarly. This may be attributed to overfitting issues, because all the classifiers operate on a very high dimensional space. Hence, they have a large amount of parameters to fit, which leaves them with a highly complex structure that is prone to overfit the data. In this regard, it is reasonable to wonder if the whole set of features gathered from the literature is appropriate in the setting of this work.

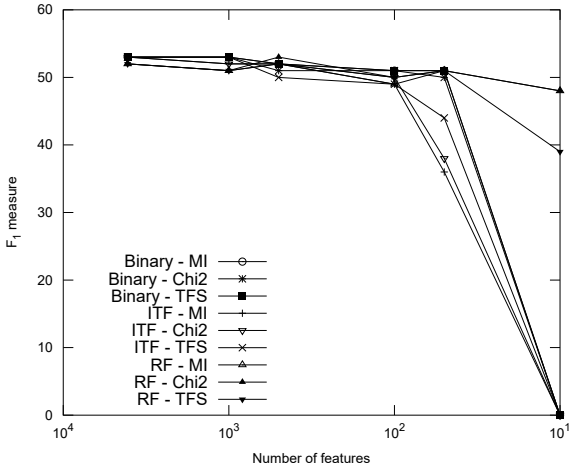
Overfitting may be reduced if the number of training examples is roughly proportional to the number of features used to represent the data [31]. In order to evaluate this hypothesis, only the essential affective information in text, i.e., plain words modelled as unigrams [43], is considered in Table VI. It can be observed that the resulting effectiveness rates essentially remain the same (for some classifiers like MNB they increase a little while decreasing for others like MLR and SVM, without significance, p=0.4667). This shows that the classifiers still overfit the data. Thus, it can be concluded that there is more than enough affective information in the words alone for the problem at hand.

Now the successful subset of learning strategies (i.e., MNB, MLR and SVM, along with the Term Weighting methods) is submitted to specific TC Feature Selection criteria (MI, χ^2 and TFS), to see if applying this dimensionality reduction method is of help to overcome the overfitting problem. The results are shown in Figure 4 for the Semeval 2007 data, and in Figure 5 for the Twitter data.

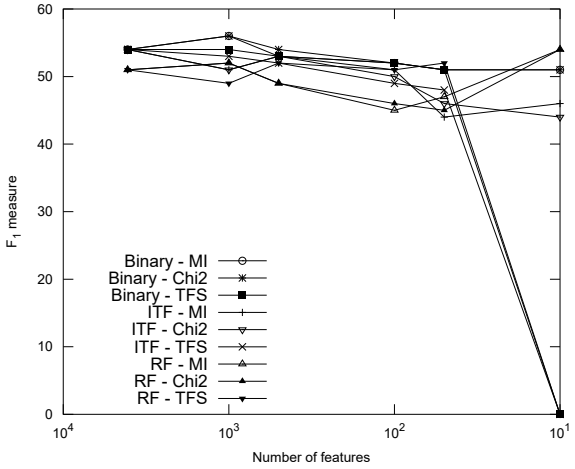
For the Semeval 2007 data shown in Figure 4a, it can be observed that the MNB behaves similarly to the full weighted feature space (averaged among the feature weighting methods, $F_1 = 52.34\%$) up to a reduction of two orders of magnitude. Then its effectiveness rates decrease considerably (even for some configurations the precision cannot be computed due to the lack of predicted labels for the class with the least



(a) Feature Selection for MNB



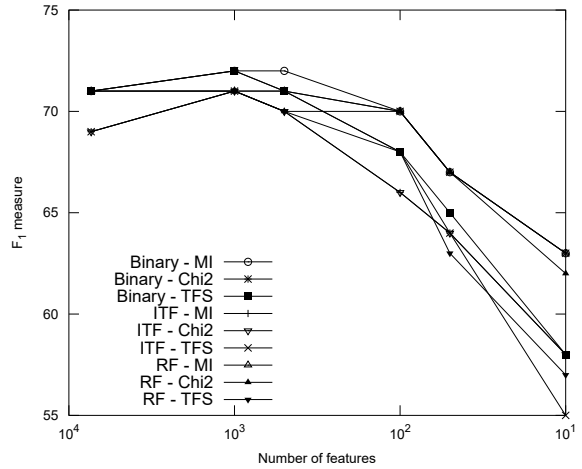
(b) Feature Selection for MLR



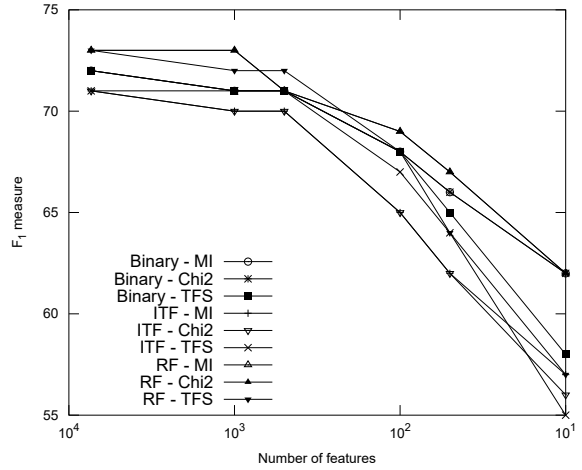
(c) Feature Selection for SVM

Fig. 4. Effectiveness rates for the Semeval 2007 dataset, obtained by 10-fold cross-validation, and using Feature Selection methods on unigrams applied to MNB, MLR and SVM with Binary-weighted features, ITF and RF. MI stands for Mutual Information, Chi2 stands for χ^2 and TFS stands for Term-Frequency-based Selection.

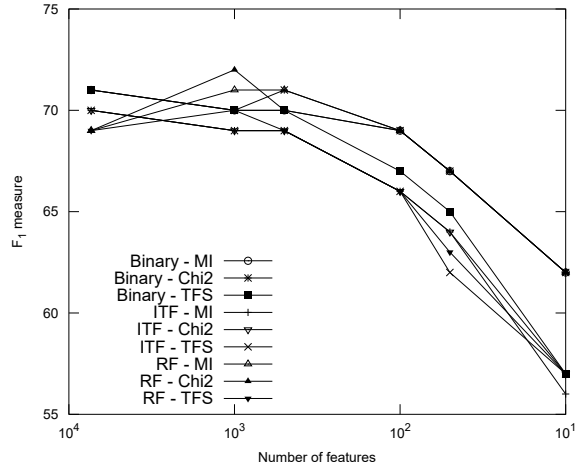
generality). Thus, MNB just learns enough knowledge from the overwhelming feature space to allow pruning it 100 times without affecting its effectiveness (for the best configuration,



(a) Feature Selection for MNB



(b) Feature Selection for MLR



(c) Feature Selection for SVM

Fig. 5. Effectiveness rates for the Twitter corpus, obtained by 10-fold cross-validation, and using Feature Selection methods on unigrams applied to MNB, MLR and SVM with Binary-weighted features, ITF and RF. MI stands for Mutual Information, Chi2 stands for χ^2 and TFS stands for Term-Frequency-based Selection.

that is RF weights with χ^2 selection, $F_1 = 54.75\%$).

As it can be observed in Figure 4b, the overall behaviour of the MLR resembles the MNB. Nevertheless, this classifier is more limited with respect to the final size of the feature space as it hardly can work under 100 features. This observation may reflect the need of a minimum amount of examples for a discriminative approach like the MLR, in contrast to the somewhat more enhanced robustness to a varying amount of features of a generative approach like the MNB, which is less affected by this aspect. Moreover, the highest effectiveness scores for the MNB are slightly better than for the MLR, which are also observed for the same system configuration, i.e., RF with χ^2 ($F_1 = 52.86\%$).

Figure 4c shows how the SVM performs very differently from the previous classifiers. For the SVM, the number of features affects its effectiveness unpredictably: any system configuration change produces a completely different result. For example, for MNB and MLR, a feature space reduction of an order of magnitude (1000 features) produces all classifiers to yield a F_1 variation within 53–55% (2% difference), while for SVM, it varies within 49–56% (7% difference, statistically significant, p=0.0044). And this behaviour is observed for the whole range of reduced features. However, the best configurations for SVM ($F_1 = 55.69\%$) are Binary weights with MI and χ^2 selections, which improve the former results with MNB by 1% (but non-statistically significant, p=0.2446).

Regarding the results for the Twitter data shown in Figure 5, the shape of the curve for the MNB (Figure 5a) and SVM (Figure 5c) is almost the same. It shows a bump of improvement around 1000 features for the MNB with binary-weighted features ($F_1 = 71.88\%$) and for the SVM with RF weights and χ^2 ($F_1 = 71.56\%$). The difference among the effectiveness rates for the SVM classifier is significant (p=0.0311) for 1000 relevant features. Regarding the MLR classifier, there is no improvement in its effectiveness, but its performance is maintained for 1000 features using RF with MI and χ^2 ($F_1 = 72.64\%$). For this classifier, there is no significant difference among the effectiveness rates for its different configurations (p=0.0762). These overall similar effectiveness trends validate the methodology proposed in this work for a different environment with more available data.

2) Train-test evaluation: The effectiveness of the classifiers with the whole set of features is shown in Table VII. It can be observed that again most of them yield similar rates, which may indicate overfitting problems, and none of them improves the best F_1 result published in the state of the art for sentiment classification with the Semeval 2007 corpus, which is set at 42.43% with a Naive Bayes classifier [22]. Exceptionally, the MLR could not predict the class with the least generality, which denotes the requirement of a minimum amount of examples for this classifier.

In contrast, the reduced feature setting with unigrams alone (see Table VIII) enables the classifiers to perform better, and this reveals the two groups of classifiers (the successful ones and the others) already observed with the whole dataset. Note that all the successful classifiers improve the baseline effectiveness rate at least by 2%, and they again appear to be the MNB, MLR and SVM. Specifically, the MNB with binary-weighted unigrams and the MLR with RF yield the best improvement margin with respect to the state of the art, which is of 7%.

TABLE VII
F_1 RESULTS WITH THE WHOLE SET OF FEATURES USING TRAIN-TEST VALIDATION. N/A STANDS FOR NOT AVAILABLE DUE TO NOT PREDICTING THE CLASS WITH THE LEAST GENERALITY.

Classifier	Term Weighting		
	Binary	ITF	RF
MNB	40.26	42.20	N/A
ARN-R	37.38	33.40	39.36
LSA	33.44	34.81	30.26
MLR	N/A	N/A	N/A
SVM	39.27	37.76	38.94

TABLE VIII
F_1 RESULTS WITH PLAIN UNIGRAM FEATURES USING TRAIN-TEST VALIDATION. N/A STANDS FOR NOT AVAILABLE DUE TO NOT PREDICTING THE CLASS WITH THE LEAST GENERALITY.

Classifier	Term Weighting		
	Binary	ITF	RF
MNB	48.89	45.41	N/A
ARN-R	37.26	32.32	42.25
LSA	37.71	37.63	31.96
MLR	N/A	N/A	49.26
SVM	45.30	36.83	N/A

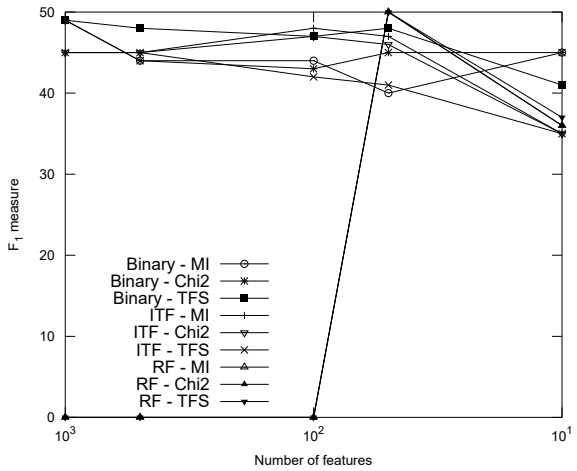
Next, the feature selection criteria are applied to the unigram-weighted space in order to improve the former effectiveness rates (see Figure 6). At first sight, note that only the MNB delivers some sort of trend while MLR and SVM behave unpredictably according to the number of selected features. This could be attributed to the simplicity of this successful generative approach in contrast to the complexity of the others. However, all the experiments coincide with having a maximum effectiveness rate of 50% when dealing with 50 features, and regardless of the feature selection method. This fact is observed for the MNB with RF, MLR with RF and SVM Binary weights. In the end, all these three methods improve the baseline F_1 rate in the state of the art (42.43%) by almost 8%.

In summary, the most effective procedure to adapt the conventional SA methods to the TTS requirements is to consider plain unigrams alone with a successful classifier like MNB, MLR or SVM. What is more, considering appropriate feature weighting and selection procedures, not only improves the effectiveness of the system a little, but also enhances its computational performance as it processes fewer feature dimensions.

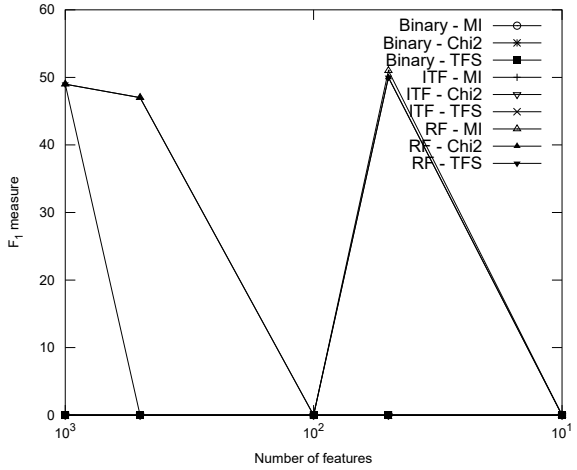
V. CONCLUSIONS

The identification of affect in text is a complex problem that has many facets to consider. In this work, we performed an exhaustive and comprehensive study to tackle a particular three-class sentiment analysis problem, at the sentence level, framed by a TTS scenario and without using additional textual data. As far as we know, this work is one of the first attempts to adapt conventional SA methods to the TTS requirements. Our experiments indicate that under such problem settings, the success of a good classifier such as MNB, MLR or SVM, greatly depends on the representation of the features, which helps the classifier to not overfit the data.

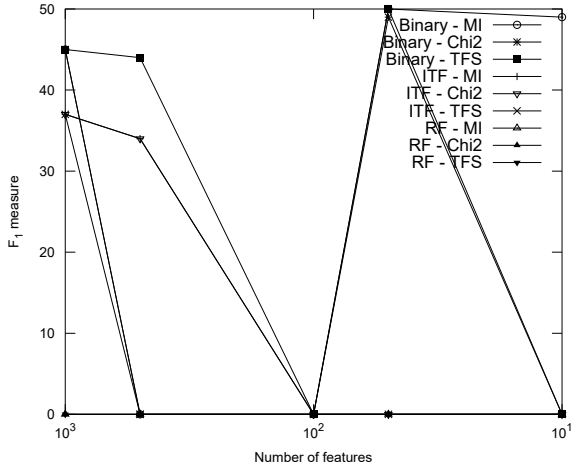
This work shows how considering the most relevant unigrams alone (with adequate weighting methods) results in better classification effectiveness compared to using additional features such as bigrams, POS tags, stems, synonyms, emotional dimensions and negations. We have evaluated our



(a) Feature Selection for MNB



(b) Feature Selection for MLR



(c) Feature Selection for SVM

Fig. 6. Effectiveness rates for the Semeval 2007 dataset, obtained by train-test validation, and using Feature Selection methods on unigrams applied to MNB, MLR and SVM with Binary-weighted features, ITF and RF. MI stands for Mutual Information, Chi2 stands for χ^2 and TFS stands for Term-Frequency-based Selection.

experiments with two corpora analysed with sentiment at the sentence level, but with differing amounts of available data and number of categories (evenly and unevenly distributed in the corpora). Although the results obtained display similar effectiveness trends for the various configurations, different effectiveness levels are observed according to the number of addressed categories and the amount of available data. For the particular problem tackled in this work, the successful classification strategies yield a similar F_1 effectiveness rate of 56% with the Semeval 2007 dataset, and 73% with the Twitter data. Finally, it is worth noting that setting the same evaluation conditions as the SA task for the Semeval 2007, the application of the sentiment analysis procedure proposed in this work improves the reported maximum effectiveness rates by 8%.

In our future work we will carefully study increasing the size of the training data regarding its computational performance, as it seems to smooth the effectiveness rate in SA. What is more, we plan to evaluate our results with a TTS system, with other languages and consider a temporal analysis for the evolution of a conversation.

REFERENCES

[1] N. Campbell, "Conversational Speech Synthesis and the Need for Some Laughter," IEEE Trans. Audio, Speech, Lang. Process., vol. 14, no. 4, pp. 1171–1178, Jul. 2006.

[2] M. Skowron, H. Pirker, S. Rank, G. Paltoglou, J. Ahn, and S. Gobron, "No Peanuts! Affective Cues for the Virtual Bartender," in Proc. of FLAIRS 2011, Palm Beach, Florida, USA, May 2011, pp. 117–122.

[3] R. Calix, S. Mallepudi, B. Chen, and G. Knapp, "Emotion Recognition in Text for 3-D Facial Expression Rendering," IEEE Transactions on Multimedia, vol. 12, no. 6, pp. 544–551, Oct. 2010.

[4] T. Wilson and G. Hofer, "Using linguistic and vocal expressiveness in social role recognition," pp. 419–422, 2011.

[5] F. Alías, X. Sevillano, J. C. Socoró, and X. Gonzalvo, "Towards High-Quality Next-Generation Text-to-Speech Synthesis: A Multidomain Approach by Automatic Domain Classification," IEEE Trans. Audio, Speech, Lang. Process., vol. 16, no. 7, pp. 1340–1354, Sep. 2008.

[6] J. Bellegarda, "A Data-Driven Affective Analysis Framework Toward Naturally Expressive Speech Synthesis," IEEE Trans. Audio, Speech, Lang. Process., vol. 19, no. 5, pp. 1113–1122, Jul. 2011.

[7] J. Pitrelli, R. Bakis, E. Eide, R. Fernandez, W. Hamza, and M. Picheny, "The IBM expressive text-to-speech synthesis system for American English," IEEE Trans. Audio, Speech, Lang. Process., vol. 14, no. 4, pp. 1099–1108, July 2006.

[8] C. O. Alm, Affect in Text and Speech. Saarbrücken, Germany: VDM Verlag, 2009.

[9] V. Francisco, R. Hervás, F. Peinado, and P. Gervs, "EmoTales: creating a corpus of folk tales with emotional annotations," Language Resources and Evaluation, vol. 45, pp. 1–41, Feb. 2011.

[10] M. Schröder, H. Pirker, M. Lamolle, F. Burkhardt, C. Peter, and E. Zovato, "Representing emotions and related states in technological systems," Emotion-Oriented Systems - The Humaine Handbook, pp. 367–386, 2011.

[11] G. O. Hofer, K. Richmond, and R. A. J. Clark, "Informed Blending of Databases for Emotional Speech Synthesis," Sep. 2005.

[12] C. O. Alm, D. Roth, and R. Sproat, "Emotions from text: machine learning for text-based emotion prediction," in Proc. of HLT'05. Morristown, NJ, USA: ACL, 2005, pp. 579–586.

[13] A. R. F. Rebordao, M. A. M. Shaikh, K. Hirose, and N. Minematsu, "How to Improve TTS Systems for Emotional Expressivity," in Proc. of Interspeech'09, Brighton, UK, Sep. 2009, pp. 524–527.

[14] M. A. M. Shaikh, A. R. F. Rebordao, K. Hirose, and M. Ishizuka, "Emotional speech synthesis by sensing affective information from text," in Proc. of ACII, Amsterdam, The Netherlands, Sep. 2009, pp. 1–6.

[15] A. Trilla, F. Alías, and I. Lozano, "Text classification of domain-styled text and sentiment-styled text for expressive speech synthesis," in Proc. of VI Jornadas en Tecnología del Habla (FALA2010). Vigo, Spain: RTTH, Nov. 2010, pp. 75–78.

[16] A. Trilla and F. Alías, "Sentiment classification in English from sentence-level annotations of emotions regarding models of affect," in Proc. of Interspeech'09, Brighton, UK, Sep. 2009, pp. 516–519.

[17] S. Tan and X. Cheng, "Improving SCL model for sentiment-transfer learning," in Proc. of NAACL-HLT. ACL, 2009, pp. 181–184.

[18] B. Pang and L. Lee, "Opinion mining and sentiment analysis," Foundations and Trends in Information Retrieval, vol. 2, no. 1-2, pp. 1–135, 2008.

[19] P. Shaver, J. Schwartz, D. Kirson, and C. O'Connor, "Emotion Knowledge: Further Exploration of a Prototype Approach," J. Pers. Soc. Psychol., vol. 52, no. 6, pp. 1061–1086, 1987.

[20] Z. Wu, H. M. Meng, H. Yang, and L. Cai, "Modeling the Expressivity of Input Text Semantics for Chinese Text-to-Speech Synthesis in a Spoken Dialog System," IEEE Trans. Audio, Speech, Lang. Process., vol. 17, no. 8, pp. 1567–1576, Nov. 2009.

[21] R. Cornelius, The science of emotion. research and tradition in the psychology of emotion. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[22] C. Strapparava and R. Mihalcea, "SemEval-2007 Task 14: Affective Text," Jun. 2007.

[23] A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," 2009. [Online]. Available: http://www.stanford.edu/~alecmgo/.../TwitterDistantSupervision09.pdf

[24] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," in Proc. of EMNLP'02, Philadelphia, PA, USA, Jul. 2002, pp. 79–86.

[25] C. Strapparava, A. Valitutti, and O. Stock, "The affective weight of lexicon," in Proc. of LREC'06, Genoa, Italy, 2006, pp. 423–426.

[26] S. Aman and S. Szpakowicz, "Identifying expressions of emotion in text," in Proc. of TSD'07. Pilsen, Czech Republic: Springer-Verlag, 2007, pp. 196–205.

[27] F. Sebastiani, "Text categorization," The Encyclopedia of Database Technologies and Applications, pp. 683–687, 2005.

[28] C. Strapparava and R. Mihalcea, "Learning to identify emotions in text," in Proc. of SAC'08. New York, NY, USA: ACM, 2008, pp. 1556–1560.

[29] S. M. Kim, A. Valitutti, and R. A. Calvo, "Evaluation of unsupervised emotion models to textual affect recognition," in Proc. of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text. Los Angeles, California: ACL, 2010, pp. 62–70.

[30] M. Lan, C. L. Tan, J. Su, and Y. Lu, "Supervised and Traditional Term Weighting Methods for Automatic Text Categorization," IEEE T. Pattern. Anal., vol. 31, no. 4, pp. 721–735, Apr. 2009.

[31] F. Sebastiani, "Machine learning in automated text categorization," ACM Comput. Surv., vol. 34, pp. 1–47, 2002.

[32] U. D. Reichel and H. R. Pfitzinger, "Text Preprocessing for Speech Synthesis," in Proc. of the TC-STAR Workshop on Speech-to-Speech Translation, Barcelona, Spain, Jun. 2006, pp. 207–212.

[33] K. Toutanova and C. D. Manning, "Enriching the knowledge sources used in a maximum entropy part-of-speech tagger," in Proc. of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora. Morristown, NJ, USA: ACL, 2000, pp. 63–70.

[34] N. Seco, T. Veale, and J. Hayes, "An Intrinsic Information Content Metric for Semantic Similarity in WordNet," in Proc. of ECAI'04, Valencia, Spain, 2004, pp. 1089–1090.

[35] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, MA, USA: Cambridge University Press, 2008.

[36] M. F. Porter, "An algorithm for suffix stripping," Program, vol. 14, no. 3, pp. 130–137, 1980.

[37] M. M. Bradley and P. J. Lang, "Affective Norms for English Words (ANEW): Stimuli, instruction manual, and affective ratings," Center for Research in Psychophysiology, University of Florida, Gainesville, Florida, USA, Tech. Rep., 1999.

[38] P. Baggia, P. Bagshaw, M. Bodell, D. Z. Huang, L. Xiaoyan, S. McGlashan, J. Tao, Y. Jun, H. Fang, Y. Kang, H. Meng, W. Xia, X. Hairong, and Z. Wu, "Speech Synthesis Markup Language (SSML) Version 1.1," W3C, Tech. Rep., Sep 2010.

[39] P. Baggia, F. Burkhardt, J.-C. Martin, C. Pelachaud, C. Peter, B. Schuller, I. Wilson, and E. Zovato, "Elements of an EmotionML 1.0," W3C, Tech. Rep., Nov 2008.

[40] Y. Dang, Y. Zhang, and H. Chen, "A Lexicon-Enhanced Method for Sentiment Classification: An Experiment on Online Product Reviews," IEEE Intell. Syst., vol. 25, no. 4, pp. 46–53, Jul.-Aug. 2010.

[41] T. M. Mitchell, "Generative and Discriminative Classifiers: Naive Bayes and Logistic Regression," Online draft, vol. 755, pp. 1–17, 2005. [Online]. Available: http://www.cs.cmu.edu/%7Etom/mlbook/NBayesLogReg.pdf
[42] B. Carpenter, "Lazy Sparse Stochastic Gradient Descent for Regularized Multinomial Logistic Regression," Alias-i, Inc., Tech. Rep., 2008.
[43] A. Batliner, D. Seppi, S. Steidl, and B. Schuller, "Segmenting into adequate units for automatic recognition of emotion-related episodes: a speech-based approach," Advances in Human Computer Interaction (AHCI), 2009.
[44] I. H. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques. San Francisco, CA, USA: Morgan Kaufmann, 2005.
[45] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," J. Am. Soc. Inform, Sci., vol. 41, no. 6, pp. 391–407, 1990.

Alexandre Trilla received the B.Sc. and M.Sc. degrees in Telecommunications Engineering by Enginyeria La Salle from Universitat Ramon Llull (URL), Barcelona, Spain, in 2006 and 2008, respectively, and the M.Sc. degree in Information Technology Management in 2010. He is currently pursuing the Ph.D. degree at URL focused on Statistical Natural Language Processing.

Since 2009, he has been an Assistant Teacher and Researcher at the Department of Media Technologies, Enginyeria La Salle (URL). He has participated in different R&D projects, and he has authored or coauthored several papers in national and international conferences and journals. His current research interests are Text Analysis, Statistical Natural Language Processing, speech technologies and Machine Learning.

Mr. Trilla has been a member of the International Speech Communication Association (ISCA) and he is currently a member of the Spanish Thematic Network on Speech Technology and the Spanish Thematic Network on Advanced Dialogue Systems.

Francesc Alías (S'05–M'07) received the B.Sc. degree in Telecommunications Engineering, M.Sc. and Ph.D. degrees in Electronics Engineering by Enginyeria La Salle from Universitat Ramon Llull (URL), Barcelona, Spain, in 1997, 1999, and 2006, respectively. From 2000 to 2004, he was a Ph.D. student granted by the Catalan Government (2000FI-00679).

From 2004 he works as a lecturer and researcher at La Salle (URL), being member of the Dept. of Communications and Signal Theory from 2004 to September 2007. Then, he moved to the Acoustic Section of the Dept. of Audiovisual Technologies and Multimedia to lead the audio and speech processing area, and later, he become scientific advisor on multimodal processing. He is currently the Director of the Human-Computer Interaction R&D area at La Salle.

Dr. Francesc Alías has leaded and participated in different R&D projects, and he has published more than 80 papers in national and international conferences and journals. He is member of the IEEE Signal Processing Society, the Spanish Thematic Network on Speech Technology, the Spanish Thematic Network on Advanced Dialogue Systems, and the ISCA Special Interest Groups on Speech Synthesis and Iberian Languages. He has been member of the Organizing or/and Scientific Committee of several scientific conferences, such as URSI 2004, SEPLN 2005, WCCI 2010, ICME 2011 and ICME 2012. He is currently reviewer of Information Sciences (Elsevier) and IEEE Transactions on Multimedia.