ICA-BASED HIERARCHICAL TEXT CLASSIFICATION FOR MULTI-DOMAIN TEXT-TO-SPEECH SYNTHESIS

Xavier Sevillano, Francesc Alías, Joan Claudi Socoró*

Enginyeria i Arquitectura La Salle. Universitat Ramon Llull
Department of Communications and Signal Theory
Pg. Bonanova 8, 08022 - Barcelona, Spain
{xavis, falias, jclaudi}@salleURL.edu

ABSTRACT

In the framework of multi-domain Text-to-Speech synthesis it is essential to *(i)* design a hierarchically structured database for allowing several domains in the same speech corpus and *(ii)* include a text classification module that, at run time, assigns the input sentences to a domain or set of domains from the database. In this paper, we present a hierarchical text classifier based on Independent Component Analysis (ICA), which is capable of *(i)* organizing the contents of the corpus in a hierarchical manner and *(ii)* classifying the texts to be synthesized according to the learned structure. The document organization and classification performance of our ICA-based hierarchical classifier are evaluated in several encouraging experiments conducted on a journalistic-style text corpus for speech synthesis in Catalan.

1. INTRODUCTION

In the last decade, concatenative Text-to-Speech (TTS) synthesis systems moved from diphone based approaches, with only one instance per unit, to *unit selection* based methods [1, 2, 3]. These methods make use of a large speech database of continuous read speech (e. g. 1 hour of speech, see [4] for a review), allowing multiple instances per unit. This corpus has to be designed to cover as much linguistic and prosodic variability as possible for a particular language or domain [5]. The characterization of the database is still an on-going research issue [4].

Unit selection TTS systems can produce sentences with high intelligibility and naturalness. However, this quality cannot usually be maintained throughout the whole sentence [6]. Thus, there is still a substantial amount of work necessary for the tuning of all parameters and features involved in the selection process [5]. As a first step, the unit selection systems have been applied to restricted domains giving high quality in synthetic speech within domain [6].

As another step towards enhancing the synthetic speech quality, we presented a new TTS system based on a multi-domain structured database [7]. This approach takes advantage of the speech quality obtained with the limited-domain approach without discarding a general purpose system. Thus, the searching space is reduced achieving high speech quality within the target domain. This multi-domain TTS architecture involves a text classification module and a hierarchically organized speech corpus.

*I would like to thank the Generalitat de Catalunya and the DURSI for their the support under grant 2000FI-00679.

Allowing for these requirements, we present a method based on Independent Component Analysis (ICA) for building a hierarchical thematic structure of the text corpus, and also for classifying the documents to be synthesized in the most appropriate domain of the corresponding speech corpus.

The training of our ICA-based hierarchical classifier is a partly supervised process, as it is developed on a manually labelled corpus. The aim of this process is to learn a hierarchical structure of thematic clusters of documents maximizing *(i)* the intra-cluster dependence, *(ii)* the inter-cluster independence and *(iii)* the documents' *recall* [8], choosing the best classifier configuration. The projection matrices and the independent components (IC) of the training documents at each level of the derived hierarchy constitute our ICA-based classifier. Testing the classifier, or categorizing new documents under the learned hierarchical structure, consists in projecting the test documents onto the ICA space and comparing the result to the training IC.

This paper is structured as follows: in Section 2 the fundamentals and the architecture of multi-domain text-to-speech conversion are presented. Section 3 deals with the description of the ICA-based hierarchical text classifier in terms of training and testing. In section 4 intensive experiments are described, and finally the conclusions of our work are discussed in Section 5.

2. MULTI-DOMAIN TEXT-TO-SPEECH (MD-TTS)

The main application of unit selection speech synthesis is a general-purpose TTS system (GP-TTS), which is able to produce *any* desired utterance from an input text [1, 2, 3]. Although the synthetic speech quality is usually very high, there are still bad synthesis examples in the unit selection GP-TTS conversion [6]. Therefore, in order to improve this issue, the unit selection process has been applied to limited domains (LD-TTS), achieving very high quality within those domains (see [4] for a review).

Furthermore, the GP speech database is usually designed to ensure that the recorded speech does not exhibit any strong speaking style, i.e. it sounds *neutral* [9]. As the TTS conversion heavily reflects the style and the coverage of the recorded database [5, 9], the synthetic speech quality decreases when the target domain of the input text mismatches the style coverage of the GP speech database [6, 10]. Chu et al. [10] also presented an approach for improving a GP-TTS system by incorporating some domain adaptation to improve the naturalness of speech.

Taking into account these ideas, we designed a multi-domain TTS (MD-TTS) system [7] (see Figure 1) in order to obtain high

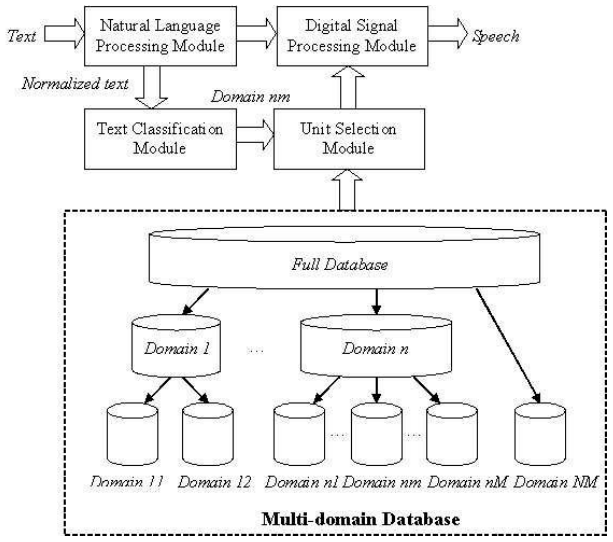

Fig. 1. Block diagram of the multi-domain unit selection TTS system with a hierarchical database and a text classification module.

synthetic speech quality (like the LD-TTS approach) in a GP-TTS framework. This architecture allows the coexistence of different *domains* in the same speech corpus: several emotions at the top level (sadness, happiness, etc.), various styles for each emotion (journalistic, literary, etc.) and diverse fields per style (politics, sports, ..., tales, poetry, etc.). Moreover, notice that while the vocabulary of each domain will certainly be specialized, it has to be designed with good phonetic and prosodic coverage [11].

Figure 1 shows the MD-TTS architecture, which involves including a text classification module for different domains [7], and designing a hierarchically structured database, which is essential for allowing several domains in the same speech corpus [9].

3. ICA-BASED HIERARCHICAL CLASSIFIER

The following paragraphs present a general description of ICA applied to text classification and the ICA-based hierarchization method for training and testing the classifier.

3.1. ICA in text classification

Text classification (TC) can be defined as the process of classifying documents into a set of predefined categories. These methods are often based on the vector space model (VSM) representation [8]. As a result, each document is defined as a vector of weights related to the terms composing the text. Before the final text representation, some pre-processing strategies are applied, looking for relevant linguistic components (stop listing and stemming) and significant semantic features (allowing dimensionality reduction) [8].

In the context of TC, the application of ICA is based on the assumption that a document collection (or corpus) is generated by a combination of topics [12, 13, 14]. Thus, a document is generated by the interaction of independent hidden random variables (thematic topics).

Using ICA in TC is related to Latent Semantic Analysis (LSA) [15]. This technique projects the documents onto a dimensionally reduced orthogonal space, extracting the K principal components from the data. This procedure is equivalent to the usual whitening preprocessing step that simplifies ICA algorithms [16]. Applying ICA on the LSA data yields K independent topics which generated the documents, allowing their classification.

3.2. Training the classifier: document organization

Once the training documents are represented in the K-dimensional LSA space, ICA is applied yielding K generating thematic topics (independent components, IC) which generated the collection. The projection of the LSA data onto the ICA space is computed by maximizing the third order moment (skewness) of the data [14]. We implement this process by means of a fixed-point algorithm (FastICA [16]), which presents fast and reliable convergence.

The value of the IC of each document is proportional to its relevance to the corresponding thematic topic. Therefore, sorting documents by their IC allows document categorization; that is, each independent component defines a cluster of documents related to the associated thematic topic.

Another key issue of the method is the choice of the space dimensionality (K). In the MD-TTS context, the document organization process is developed on a hand-labelled corpus, i.e. the *basic* number of topics (T) is known *a priori*. Nevertheless, no hierarchical information is available. By means of the ICA method the corpus is hierarchized (which is essential for MD-TTS synthesis), depending on the relation between K and T:

1. If $K = T$, the documents are assigned to the basic domains.
2. If $K < T$, domains are merged into upper level clusters containing the most statistically dependent domains.
3. If $K > T$, domains are split into sub-domains, creating lower level clusters.

Intuitively, the documents belonging to a thematically homogeneous domain tend to group in one cluster, which will be discovered for small values of K. On the other hand, documents from a more varied domain will be distributed in several clusters, which will only be discovered when the dimensionality of the space is increased. It is important to note that in a document collection there may exist more topics than those indexed by a human labeller [14].

In order to determine the lower hierarchical structure of the corpus (see figure 1), ICA is iteratively applied for increasing values of $K > T$, until the optimal number of clusters for each domain is found. At each step, the recall [8] of the clusters is calculated, selecting as the optimal number of clusters the one yielding the maximum recall for each domain.

As a summary, the results of training the ICA-based hierarchical classifier are the LSA projection matrices, the ICA separating matrices ($\mathbf{W}$) and the independent components of the training documents at each level of the hierarchy.

3.3. Testing the classifier: document classification

The document classification process follows three steps: first, the vectors representing the test documents are projected onto the training LSA space [15]. Secondly, these LSA components are mapped onto the ICA space by means of the separating matrices $\mathbf{W}$ obtained from the document organization process. And thirdly, the training and test IC are compared and sorted in order to classify the test documents.

4. EXPERIMENTS

The following experiments have been conducted on a collection of articles extracted from the Catalan newspaper AVUI, compiled during two periods of time in 2000 and 2003 [7]. This journalistic-style corpus is composed of 200 documents (2600 terms) divided into four thematic domains (fields): $D = \{$*politics* (60 documents), *society* (60 docs), *music* (40 docs), *theatre* (40 docs)$\}$. These documents are represented by means of their term frequency [8] in the VSM after stop listing and stemming.

The proposed ICA-based method for multi-domain corpus hierarchization has been evaluated in two stages: training (document organization) and test (document classification). The first stage consists of building a hierarchical classifier, which is evaluated by comparing its decisions to the labelled documents. In the second stage, the documents excluded from the training phase are used to test the classifier (test sets ranging from 5 to 20% of the corpus). Moreover, these train-and-test experiments have been developed following a *10-fold cross-validation* approach [8]. This is due to the *reduced* size of text corpora in the MD-TTS framework, when compared to larger document collections referred in TC literature (e.g. the Reuters collection [8]).

4.1. Document organization

The following experiments analyze the capabilities of the ICA algorithm for organizing the documents in the corpus hierarchically.

4.1.1. Merging domains

The first experiment consists of finding $K = \{2, 3, 4\}$ clusters ($K = 1$ is the full database), obtaining the upper levels of the hierarchy, which we named *macro-domains*, *superdomains* and *domains*, respectively. As we are looking for homogeneous clusters, recall is the parameter used for evaluating the performance of the document organization process (see table 1). According to the terminology defined in section 3.2, for $K = 2$ the documents are clustered into two macro-domains: *politics+society* and *music+theatre*. Furthermore, for $K = 3$ the documents are categorized under three superdomains: *politics*, *society* and *music+theatre* (see figure 3 for a graphical representation).

Recall	$K=2$		$K=3$		$K=4$	
% Training	***90***	***80***	***90***	***80***	***90***	***80***
Politics	1	1	.800	.814	.757	.650
Society	.648	.648	.898	.890	.832	.762
Music	.971	.961	.910	.869	.934	.786
Theatre	1	1	1	.987	1	.923

Table 1. Recall of the clusters obtained by the ICA algorithm over six of the different training configurations after *10-fold* averaging.

The TC algorithm presents a stable behaviour for the analyzed training configurations. Thus, the ability of hierarchization of the ICA-based method is demonstrated, which is essential in the MD-TTS framework as discussed previously.

4.1.2. Splitting domains

The second experiment evaluates the ability of the ICA method to classify documents when searching for subdivisions of original domains, which we called *sub-domains*. Figure 2 shows the evolution of the recall of each domain for $K > 4$, averaged across all the test configurations. The maximum recall in the *politics* domain is achieved for $K = 6$ (in average, .91), improving the result obtained when $K = 4$ (see figure 2). The optimal number of sub-domains corresponding to the politics documents is 4. On the other hand, *society*, *music* and *theatre* documents tend to group in one cluster though the dimensionality of the ICA space is increased, i.e. the maximum recall is achieved when $K = 4$.

These results are dependent on the corpus contents. Thus, in order to obtain a rich hierarchical organization for each domain, the MD-TTS corpus should be designed consequently.

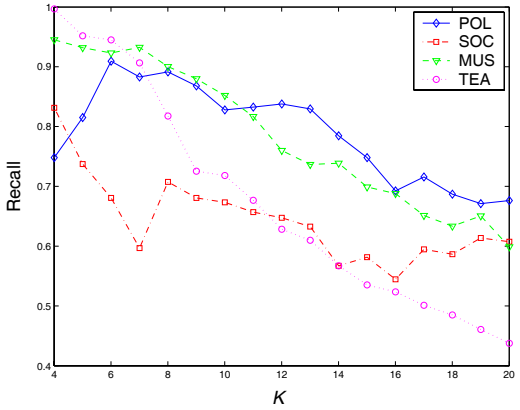

Fig. 2. Averaged recall per domain for $K \in [4, 20]$.

As a summary of the document organization process, figure 3 shows the final hierarchical structure obtained after averaging the 10-fold experiments using 80% of documents for training.

4.2. Document classification

The following experiment evaluates the *accuracy* [8] of the ICA-based classifier built after the training process. Table 2 shows the accuracy averaged through the 10 and 20% test experiments, related to the training processes evaluated in table 1. The classification of the test *society* documents presents a worse behaviour due to the heterogeneity of their contents, compared to the higher accuracies achieved for the rest of the domains.

Accuracy	$K=2$		$K=3$		$K=4$	
% Test	***10***	***20***	***10***	***20***	***10***	***20***
Politics	1	1	.867	.858	.850	.767
Society	.717	.725	.850	.833	.783	.675
Music	.950	.955	.925	.911	.925	.844
Theatre	1	1	1	1	1	1

Table 2. Accuracy of the hierarchical ICA-based classifier over six different test configurations after *10-fold* averaging.

Document classification in lower levels ($K > 4$) of hierarchy only deals with the politics documents (as discussed in section 4.1.2). The sub-domain accuracies (.950 for 10% and .917 for 20% test sets) notably improve the results achieved for $K = 4$ (domain level). Thus, the politics documents are better represented with

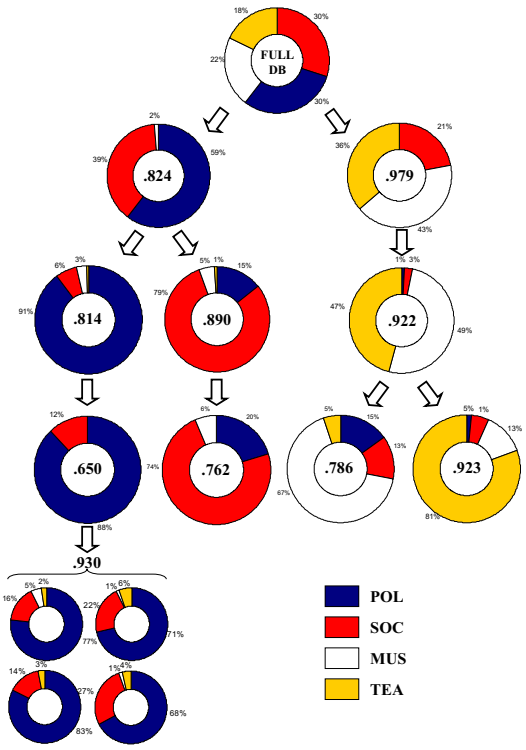

Fig. 3. Corpus hierarchical organization after *10-fold* averaging of the 80% training experiments. Recall is shown for each cluster.

four IC (each defining a sub-domain) than with just one domain. This is due to their diverse thematic contents. Moreover, good accuracies are still obtained when the number of training documents is notably reduced (at the most, training the classifier with 40% of documents per domain yields an average test accuracy of .81 for $K = 4$). Thus, the ICA-based method has presented a remarkable generalization ability in our experiments.

5. CONCLUSIONS

In this paper we have presented an ICA-based hierarchical document classifier for multi-domain Text-to-Speech (MD-TTS) synthesis. The experiments demonstrate its good performance for text corpus hierarchization. Moreover, the classifier achieves encouraging results for different training and testing configurations, even when few documents are available. This ability is essential in the MD-TTS context due to the reduced size of the corpus. In addition, as could be expected, the final hierarchical structure is highly dependent on the contents of the corpus.

Further studies will be focused on determining whether the derived hierarchical structure of the corpus is optimal after formal listening tests using the hierarchized MD-TTS database. Another key point is considering the reliability of the decisions taken by the classifier. For instance, if we try to classify documents belonging to none of the domains contained in the corpus, the relevance scores for all the thematic topics will be low. Thus, a more robust classifier would be built if this information was taken into account.

6. REFERENCES

[1] A.W. Black and P. Taylor, "Automatically clustering similar units for unit selection in speech synthesis," in *EuroSpeech*, Rodes, Greece, 1997, pp. 601–604.

[2] M. Beutnagel, A. Conkie, J. Schroeter, Y. Stylianou, and A. Syrdal, "The AT&T Next-Gen TTS system," in *Joint Meeting of ASA, EAA, and DAGA2*, Berlin, Germany, 1999, pp. 18–24.

[3] E. Eide, A. Aaron, R. Bakis, P. Cohen, R. Donovan, W. Hamza, T. Mathes, M. Picheny, M. Polkosky, M. Smith, and M. Viswanathan, "Recent Improvements to the IBM Trainable Speech Synthesis System," in *Proceedings of ICASSP*, Hong Kong, 2003.

[4] B. Möbius, "Corpus-based speech synthesis: methods and challenges," *Arbeitspapiere des Instituts für Maschinelle Sprachverarbeitung (AIMS)*, vol. 6, no. 4, pp. 87–116, 2000.

[5] A.W. Black, "Perfect Synthesis for all of the people all of the time," in *IEEE TTS Workshop 2002 (Keynote)*, Santa Monica, USA, 2002.

[6] A.W. Black and K. Lenzo, "Limited Domain Synthesis," in *ICSLP*, Beijing, China, 2000.

[7] F. Alías, I. Iriondo, and P. Barnola, "Multi-domain text classification for unit selection Text-to-Speech Synthesis," in *The 15th International Congress of Phonetic Sciences (ICPhS)*, Barcelona, August, pp. 2341–2344.

[8] F. Sebastiani, "Machine learning in automated text categorisation," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[9] A. Breen and P. Jackson, "Non-uniform unit selection and the similarity metric within BT's LAUREATE TTS system," in *The 3rd ESCA/COCOSDA Workshop on Speech Synthesis*, Jenolan Caves, NSW, Australia, 1998.

[10] M. Chu, C. Li, P. Hu, and E. Cahng, " Domain adaption for TTS Systems ," in *ICASSP*, Orlando, USA, 2002.

[11] J.M. Montero, R. Córdoba, J.A. Vallejo, J. Gutiérrez-Arriola, E. Enríquez, and J.M. Pardo, "Restricted-domain female-voice synthesis in Spanish: from database design to ANN prosodic modelling," in *ICSLP*, Beijing, China, 2000, pp. 621 – 624.

[12] C.-L. Isbell and P. Viola, "Restructuring Sparse High Dimensional Data for Effective Retrieval," *Advances in Neural Information Processing Systems*, , no. 11, pp. 480–486, 1999.

[13] T. Kolenda, L.K. Hansen, and S. Sigurdsson, "Independent Components in Text," in *Advances in Independent Component Analysis*. Springer-Verlag, 2000.

[14] A. Kabán and M. Girolami, "Unsupervised Topic Separation and Keyword Identification in Document Collections: A Projection Approach," Technical Report Nr. 10, Dept. of Computing and Information Systems, University of Paisley, 2000.

[15] S. Deerwester, S.-T. Dumais, G.-W. Furnas, T.-K. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *Journal American Society Information Science*, vol. 6, no. 41, pp. 391–407, 1990.

[16] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, John Wiley and Sons, 2001.