GTM User Modeling for aIGA Weight Tuning in TTS Synthesis

Lluís Formiga
Universitat Ramon Llull, Spain

Francesc Alías
Universitat Ramon Llull, Spain

INTRODUCTION

Unit Selection Text-to-Speech Synthesis (**US-TTS**) systems produce synthetic speech based on the **retrieval** of previous recorded speech units from a **speech database (corpus)** driven by a weighted cost function (Black & Campbell, 1995). To obtain high quality synthetic speech these **weights must be optimized** efficiently. To that effect, in previous works, a technique was introduced for **weight tuning** based on evolutionary perceptual tests by means of **Active Interactive Genetic Algorithms** (**aiGAs**) (Alías, Llorà, Formiga, Sastry & Goldberg, 2006) **aiGAs** mine models that map **subjective preferences** from users by **partial ordering graphs**, **synthetic fitness** and **Evolutionary Computation (EC)** (Llorà, Sastry, Goldberg, Gupta & Lakshmi, 2005). Although **aiGA** propose an effective method to **map *single* user preferences**, as far as we know, the methodology to extract common solutions among different **individual preferences** (hereafter denoted as *common knowledge*) has not been tackled yet. Furthermore, there is an ambiguity problem to be solved when different users evolve to different weight configurations. In this review, **Generative Topographic Mapping (GTM)** is introduced as a method to extract common knowledge from **aiGA** models obtained from **user preferences.**

BACKGROUND

Weight Tuning in Unit-Selection Text-to-Speech Synthesis

The aim of **US-TTS** is to generate synthetic speech by concatenating the sequence of units that best fit the requirements derived from the input text. The speech units are retrieved from a **database** (**speech corpus**) which stores speech-units previously recorded by a professional speaker, typically.

Text-to-speech workflow is generally modelled as two independent blocks that convert written text into speech signal. The first block is named Natural Language Processing (NLP), which is followed by the Digital Signal Processing block (DSP). At first stage, The NLP block carries out a text preprocessing (e.g. conversion of digit numbers or acronyms to words), then it converts graphemes to phonemes. And at last stage, the NLP block assigns quantified prosody parameters to each phoneme guiding the way each phoneme is converted to signal. Generally, this quantified prosody parameters involve duration, pitch and energy. Next, The DSP block retrieves from a recorded **database** (**speech corpus**) the sequence of units that best matches the target requirements (the phonemes and their prosody). Finally, the speech units are ensembled to obtain the output speech signal.

The **retrieval** process is done by a dynamic programming algorithm (e.g. Viterbi or A* (Formiga & Alías, 2006)) driven by a cost function. The cost function computes the load of **selecting** a **unit** within a sequence as the sum of two weighted subcosts (see equation (1)): the target subcost (C^t) and the concatenation subcost (C^c). In this work, the C^t is considered as a weighted linear combination of the normalized prosody distances between the target-NLP predicted prosody vector and the candidate unit prosody vector (see equation). Otherwise, the C^c is computed as a weighted linear combination of the distances between the feature vectors of the speech signal around its concatenation point (see equation).

$$C(t_1^n, u_1^n) = \sum_{i=1}^{n} C^t(t_i, u_i) + \sum_{i=2}^{n} C^c(u_{i-1}, u_i) \quad (1)$$

$$C^t(t_i,u_i)=\sum_{j=1}^{p} w_j^t C_j^u(t_i,u_i) \tag{2}$$

$$C^c(u_{i-1},u_i)=\sum_{j=1}^{q} w_j^c C_j^c(u_{i-1},u_i) \tag{3}$$

where t_1^n represents the target units sequence $\{t_1, t_2,...,t_n\}$ and u_1^n represents the candidate units sequence $\{u_1, u_2,..., u_n\}$.

$$C_j^t(t_i,u_i)=1-e^{\left(\frac{P_j(t_i)}{\sigma_{P_j}}\right)^2} \tag{4}$$

$$C_j^c(u_{i-1},u_i)=1-e^{\left(\frac{P_j^R(u_{i-1})-P_j^L(u_i)}{\sigma_{P_j}}\right)^2} \tag{5}$$

Appropriate design of cost function by means of **weight training** is a crucial to earn high quality synthetic speech (Black, 2002). Nevertheless this concern has focused approaches with no unique response. Several techniques have been suggested for **weight tuning**, which may be spitted into three families: *i)* manual-tuning *ii)* computationally-driven purely objective methods and *iii)* perceptually optimized techniques (Alías, Llorà, Formiga, Sastry & Goldberg, 2006). The present review is based on the techniques based on human feedback to the training process, following previous work (Alías, Llorà, Formiga, Sastry & Goldberg, 2006), which is outlined in the next section.

The Approach: Interactive Evolutionary Weight Tuning

Computationally-driven purely objective methods are mainly focused on an acoustic measure (obtained from cepstral distances) between the resynthesized and the natural signals. Hunt and Black adopted two approaches in (Hunt & Black, 1996). The first approach was based on **adjusting the weights** through an exhaustive search of a prediscretized weight space (weight space search, WSS). The second approach proposed by the authors used a multilinear regression technique (MLR), across the entire **database** to **compute the desired weights**. Later, Meron and Hirose (Meron & Hirose, 1999) presented a methodology that improved the efficiency of the WSS and refined the MLR method. In a previous work (Alías & Llorà, 2003), introduced **evolutionary computation** to perform this tuning. More precisely, Genetic Algorithms (GA) were applied to obtain the most appropriate weight. The main added value of making use of GA to find optimal weight configuration is the independency to linear search models (as in MLR)and, in addition, it avoids the exhaustive search (as in WSS).

However, all this methods lack on its dependency on the acoustic measure to determine the actual quality of the synthesized speech, which in most part is relative to human hearing. To obtain better speech quality, it was suggested that user should take part in the process. In (Alías, Llorà, Iriondo, Sevillano, Formiga & Socoró, 2004) there were conducted preference tests by synthesizing the training text according to two different weights and comparing the obtained speech subjective quality. Subsequently, **Active Interactive Genetic Algorithms** were presented in (Llorà, Sastry, Goldberg, Gupta & Lakshmi, 2005) as one interactive **evolutionary computation** method where the user feedback evolves the solutions through *survival-of-the-fittest* mechanism. The solutions inherent **fitness** is based on the **partial order** provided by the evaluator; **Active iGAs** base its efficiency on evolving different solutions by means of **surrogate fitness**, which generalize the **user preferences**. This surrogate fitness and the **evolutionary process** are based on the following key elements: *i)* **partial ordering**, *ii)* induced complete order, and *iii)* surrogate function via ε Support Vector Machines (ε-SVM). Preference decisions made by the user are modelled as a **directional graph** which is used to generate partial ordering of solutions (e.g: $\hat{x}_1 > \hat{x}_2; \hat{x}_2 > \hat{x}_3 : \hat{x}_1 \rightarrow \hat{x}_2 \rightarrow \hat{x}_3$) (see figure 1). Table 1 shows the approach of global rank based on dominance measure: given a vertex v, the number of dominated vertexes $\delta(v)$ and dominating vertexes is computed. Using this measures, the estimated fitness may be computed as $\hat{f}(v)=\delta(v)-$ (v). The estimated ranking $\hat{r}(v)$ is obtained by sorting based on $\hat{f}(v)$ (Llorà, Sastry, Goldberg, Gupta & Lakshmi, 2005). The procedure of **aiGA** is detailed in algorithm 1.

However, once the global weights were obtained with **aiGA**, there was no single dominant weight solution (Alías, Llorà, Formiga, Sastry & Goldberg, 2006), i.e. each test performed by different users gave similar and different solutions. This fact implied that a second group of users had to validate the obtained weights.

Smart Home Research: Projects and Issues
Michael P. Poland, Chris D. Nugent, Hui Wangand Liming Chen (2009). *International Journal of Ambient Computing and Intelligence (pp. 32-45).*
www.irma-international.org/article/smart-home-research/37474