INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Adaptive Latency for Part-of-Speech Tagging in Incremental Text-to-Speech Synthesis

Maël Pouget[1,2], *Olha Nahorna*[1,2], *Thomas Hueber*[1,2], *Gérard Bailly*[1,2]

[1]CNRS/GIPSA-Lab, Grenoble, France
[2]Univ. Grenoble Alpes/GIPSA-Lab, Grenoble, France
[1,2]firstname.lastname@gipsa-lab.fr

Abstract

Incremental text-to-speech systems aim at synthesizing a text 'on-the-fly', while the user is typing a sentence. In this context, this article addresses the problem of the part-of-speech tagging (POS, i.e. lexical category) which is a critical step for accurate grapheme-to-phoneme conversion and prosody estimation. Here, the main challenge is to estimate the POS of a given word without knowing its 'right context' (i.e. the following words which are not available yet). To address this issue, we propose a method based on a set of decision trees estimating online whether a given POS tag is likely to be modified when more right-contextual information becomes available. In such a case, the synthesis is delayed until POS stability is guaranteed. This results in delivering the synthetic voice in word chunks of variable length. Objective evaluation on French shows that the proposed method is able to estimate POS tags with more than a 92% accuracy (compared to a non-incremental system) while minimizing the synthesis latency (between 1 and 4 words). Perceptual evaluation (ranking test) is then carried in the context of HMM-based speech synthesis. Experimental results show that the word grouping resulting from the proposed method is rated more acceptable than word-by-word incremental synthesis.

Index Terms: Incremental speech synthesis, natural language processing, classification, TTS, part-of-speech

1. Introduction

Text-to-speech (TTS) systems are now able to produce very high-quality synthetic voice. They can be used as a substitute voice by people with severe communication disorders (such as patients with Parkinson's disease or ALS). However, TTS-based communication lacks interactivity since the synthesis is generally triggered on a per-sentence basis. Therefore, the listener (i.e. the communication partner) has to wait for a complete sentence to be typed down. This increases drastically the communication latency and often results in some frustration for both the listener and the system user. Incremental TTS (iTTS) [1, 2, 3] aims at improving this interactivity issue by delivering the synthetic voice 'on-the-fly' (*i.e.* while the user is typing the target sentence) with almost the same quality as a conventional (*i.e.* non incremental) TTS.

The main challenge in iTTS is to perform the two main steps of a conventional TTS, that are text analysis (often referred to as natural language processing, NLP) and waveform generation, when considering only a limited lookahead. In other word, the iTTS paradigm assumes that the synthesis of a given word can rely only on its 'left-context' (*i.e.* the words before it) and that almost no 'right-context' (*i.e.* the words after it) is available. In our previous study [4], we focused on the waveform generation step in the context of HMM-based speech synthesis. We proposed a method for building HMM voices using models trained with limited and adaptive lookahead. In this paper, we focus on the text analysis step, and in particular on Part-Of-Speech (POS) tagging. This step consists in assigning a lexical category to each word (e.g. noun, verb, etc.), based on both morphological analysis and syntactic constraints, *i.e.* its relationship with left- and right-adjacent words in the sentence. POS-tagging is critical for grapheme-to-phoneme conversion but also for prosody estimation since the syntactic structure of the sentence is actually derived from the POS tags.

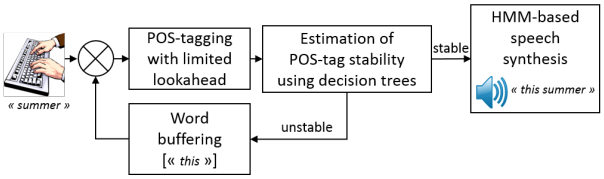

Figure 1: *Overview of the proposed iTTS architecture with adaptive latency for robust online POS-tagging*

In [5], Beuck et al. propose four strategies for performing POS-tagging incrementally, in the context of NLP. These strategies as well as their use in the context of iTTS can be briefly summarized as follows:

- estimating POS tags using left-context only. In iTTS, this results in a zero delay for delivering the synthetic voice but some POS tags may be inaccurate.
- considering a fixed size lookahead (typically 2 or 3 words) for disambiguating POS tags. In iTTS, this results in a constant latency but likely more accurate POS tags.
- recalculating the POS of a given word already tagged when more right-context becomes available (a system allowing such behavior is referred by the authors as a non-monotonic system). In iTTS, the synthesis has to be postponed until the POS tag can no longer be modified. As discussed later, this is the core idea of the iTTS architecture proposed in this article.
- considering multiple hypothesis for each new available word. In iTTS, this will require to propagate such ambiguities to the signal processing module. This approach seems interesting but is not considered in the present study.

In line with the third strategy, we propose a method for estimating POS tags accurately in the context of iTTS while

http://dx.doi.org/10.21437/Interspeech.2016-165

minimizing the lookahead (and thus maximizing the reactivity of the synthesizer). The proposed method (described in Section 2) is based on a set of decision trees estimating online whether a given POS tag is likely to be modified when more right-contextual information becomes available. Each decision tree models the stability of a POS tag for a given left-context and a given lookahead. In the present study, we consider an adaptive lookahead between 0 and 2 words. The synthesis of a word is triggered as soon as the stability of its related POS-tag is guaranteed. This results in delivering the synthetic voice in word chunks of variable length (i.e. adaptive latency). A general overview of the proposed architecture for a so-called "adaptive-latency iTTS" is presented in Figure 1. The proposed method is evaluated both objectively and perceptively, in the context of our HMM-based iTTS system for French [4] (Section 3).

2. Proposed method

2.1. POS-tagging in incremental TTS

Many approaches have been proposed in the literature to address automatic POS-tagging in conventional (i.e. non-incremental) TTS (see [6], ch. 10 and [7] for reviews). Modern taggers are almost all based on the two following steps: (1) the extraction of one or several hypothesis for each word considered separately from its context and (2) a global optimization which aims at alleviating ambiguities by making use of the large-span context. POS-tagger such as TnT [8] or Festival [9] use second order Markov models with states representing the tags and outputs (i.e. observations) representing the words (and thus state transition probabilities modeling pairs of tags). The POS-tagger used in this study for French language, called COMPOST [10], is based on the same approach. Following the formulation used in [8], the most likely tag sequence $[\hat{c}_1, \ldots, \hat{c}_T]$ associated with the word sequence $[w_1, \ldots, w_T]$ of length T is defined such as:

$$\underset{[c_1,\ldots,c_T]}{\arg\max}\{[\prod_{t=1}^{T} P(c_t|c_{t-1}, c_{t-2})P(w_t|c_t)]P(c_{T+1}|c_T)\} \quad (1)$$

where c_{-1}, c_0, and c_{T+1} are beginning/end sentence markers, $P(w_t|c_t)$ is related to tag estimation without taking into account any contextual information and $P(c_t|c_{t-1}, c_{t-2})$ refers to a 3-gram model providing prior information on the current tag c_t given the tags of the two previous words (c_{t-1} and c_{t-2}). These probabilities can be derived from relative frequencies estimated on large text corpora. In the framework of Markov modeling, Equation (1) is typically solved using the Viterbi algorithm.

Such formulation assumes that the final tag c_{T+1} is known without any ambiguity (as well as c_{-1} and c_0). In conventional TTS, it often corresponds to a "End-of-sentence" marker such as a period. However, such assumption can not be made when processing the input text incrementally. Thus, the POS-tagging technique needs to be adapted. Here, we propose to solve (1) for the word sequence $[w_1, \ldots, w_t]$ each time a new word w_t is made available (e.g. when the user presses the space bar), using the forward-backward rather than the Viterbi algorithm. The associated tag c_t is defined as the one that maximizes the forward probabilities $\alpha_t(j,k) = P(c_1, \ldots, c_{t-1} = j, c_t = k|w_1, \ldots, w_t)$ over all the possible N tags. This forward probability can be calculated using the well-known recursive expression:

$$\alpha_t(j,k) = \sum_{i=1}^{N} \alpha_{t-1}(i,j)P(c_t = k|c_{t-1} = j, c_{t-2} = i) \quad (2)$$

assuming that word w_{t-2} was given the tag i and a transition between states j and k at times $t-1$ and t. Each previous word w_k of $[w_1, \ldots, w_{t-1}]$ is then tagged by calculating the posterior probabilities:

$$P(c_k = k|w_1, \ldots, w_t) = \sum_{j=1}^{N} \alpha_k(j,k)\beta_k(j,k) \quad (3)$$

with $\beta_k(j,k)$ the backward probability given by:

$$\beta_t(j,k) = \sum_{i=1}^{N} \beta_{t+1}(i,j)P(c_t = k|c_{t+1} = j, c_{t+2} = i) \quad (4)$$

2.2. Evaluation of POS tag stability using decision trees

The POS-tagging procedure presented in the previous section is sub-optimal since an uncertainty remains on the final tag c_t. Indeed, its online estimation relies only on the left-context and therefore may sometimes be incorrect. Moreover, if c_t is incorrect, the backward propagation may influence in a bad way the tags further left (i.e. $[c_1, \ldots, c_{t-1}]$). To alleviate this potential negative effect, we propose a method for estimating the stability of a POS tag, in a given (syntactic) context, that it how it is likely to be modified when more right-context becomes available.

The proposed method is based on a set of 3 binary decision trees. Each decision tree models the stability of a POS tag for a given lookahead (i.e. right-context) of 0,1,or 2 words. Input features are composed of a sequence of 3 consecutive tags $[c_{t-2}, c_{t-1}, c_t]$ calculated incrementally, together with their associated probabilities $[P(c_{t-2} = i|w_1, \ldots, w_t), P(c_{t-1} = j|w_1, \ldots, w_t), P(c_t = k|w_1, \ldots, w_t)$ given by Equation (3). The output feature is a binary value indicating if the POS tag calculated incrementally matches the one estimated from the complete left and right context. In other word, for each tree, the set of yes/no questions partitions the training set regarding to the following rules: "Is $(c_t|w_1, \ldots, w_{t+L}) = (c_t|w_1, \ldots, w_T)$?" where L is the considered lookahead and T the number of words in each training sentence. Note that c_{-1} and c_0 are set as an explicit Beginning-of-Sentence class with a probability equal to 1. As an example, let us consider the French sentence "*Cet été, les enfants vont à la mer*" ("This summer, the children will go to the sea"). Training input observations are built by successively sending the following chunks to the POS-tagger: "*Cet*", "*Cet été,*", "*Cet été, les*", "*Cet été, les enfants*", "*Cet été, les enfants vont*", etc. and by storing the successive POS tags for each word, with their respective probabilities.

2.3. Adaptive latency iTTS

As already mentioned, a POS-tagging error can have important consequences on the grapheme-to-phoneme conversion as well as on the prosody. With this consideration in mind, we propose a new iTTS architecture in which the synthesis of a given word w_t is delayed until the stability of its associated POS-tag (determined using the procedure describe in Section 2.1) is guaranteed. This stability is assessed using the decision trees presented in Section 2.2. The proposed algorithm for triggering the synthesis is presented in Algorithm 1. This procedure results in delivering the synthetic voice in word chunks of variable length, introducing a variable latency but maximizing the POS-tagging accuracy. The maximum latency which can be obtained using this procedure is 3 words. This happens when a given POS tag is still classified as "unstable" even when considering a 2-words lookahead.

Data: $[w_{t-2}, w_{t-1}, w_t]$, $[c_{t-2}, c_{t-1}, c_t]$
waiting list : Typed words, not synthesized yet.
if w_{t-3} *is in waiting list* **then**
  $Synthesize(w_{t-3})$
if w_{t-2} *is in waiting list* **then**
  if $IsStable(c_{t-2})$ *(2-word lookahead)* **then**
    $Synthesize(w_{t-2})$
  else
    Put w_{t-2}, w_{t-1}, w_t in waiting list **return**;
if w_{t-1} *is in waiting list* **then**
  if $IsStable(c_{t-1})$ *(1-word lookahead)* **then**
    $Synthesize(w_{t-1})$
  else
    Put w_{t-1}, w_t in waiting list **return**;
if w_t *is in waiting list* **then**
  if $IsStable(c_t)$ *(0-word lookahead)* **then**
    $Synthesize(w_t)$
  else
    Put w_t in waiting list **return**;

Algorithm 1: Proposed algorithm for scheduling the incremental synthesis of chunks of words based on the stability of the POS-tagging.

3. Experiments

3.1. Objective evaluation

The proposed method was evaluated in the context of our incremental HMM-based speech synthesis system [4], which is based on the NLP front-end COMPOST [10] and the HTS toolkit [11]. The corpus used for training the decision trees was extracted from the two French books "Notre-Dame de Paris", by Victor Hugo and "Le tour du monde en 80 jours", by Jules Verne. This corpus consists in 20154 sentences (290801 words). The corpus was divided into a training set (2/3 of the corpus : 13436 sentences, around 193000 words) and a testing set (1/3 of the corpus : 6718 sentences, around 98000 words). The training of the decision trees was done using Matlab (*classregtree* package).

First, we evaluated the performance for each of the 3 decision trees considered *independently*. That is, their ability to evaluate whether a POS tag is likely to be modified when considering more right-context (i.e. a lookahead of 0, 1, or 2 words). The performance was measured by calculating the *accuracy* (Acc), defined as

$$Acc = (TP + TN)/(TP + TN + FP + FN)$$

where TP, TN, FP, and FN are respectively true positives, true negatives, false positives and false negatives. Results are presented in Figure 2.

First, let us discuss the performance in terms of POS tag correctness as a function of the lookahead (that is the raw performance of the NLP front-end COMPOST considered in this study). With no lookahead, around 40% of the POS tag are badly estimated (i.e. $(TN+FP)/(TP+FN+TN+FP)$). As expected, the performance increases with the lookahead, with 9% of error when considering 1 word, and less than 2% when considering 2 words. These results show that (1) POS-tags can be accurately estimated online when considering at least a lookahead of two words, (2) a new strategy was in fact needed to achieve lower latency. Let us now discuss the ability of the decision trees to evaluate the stability of a POS tag.

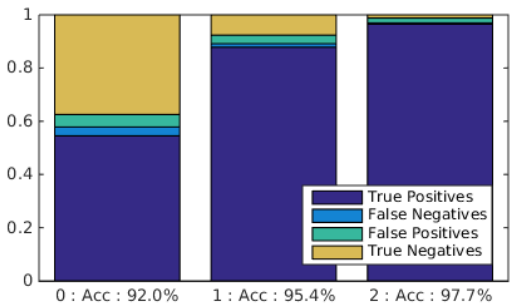

Figure 2: *Objective evaluation of the decision trees (considered independently) estimating the stability of a POS tag as a function of the lookahead (for left to right: 0, 1, and 2 words).*

With no lookahead, the stability of the POS tag was correctly assessed in 92% of the cases (i.e. $(TP + TN)/(TP + TN + FN + FP)$). Among these decisions, in 37% of the cases, it was rightly decided to postpone the synthesis since the stability of the POS was not guaranteed (TN). On the contrary, in 55% of the cases, the POS tag was considered to be stable so that the synthesis could be triggered confidently (TP). In 8% of the cases, the stability of the POS tag was wrongly assessed, resulting either in a synthesis triggered too soon and with an erroneous POS tag (FP) or with an unnecessary latency (FN). As expected, the number of such errors (i.e. $FP + FN$) decreases when the lookahead increases, with $\sim 4\%$ for a 1-word lookahead and $\sim 2\%$ for a 2-word lookahead.

Then, we evaluated the performance of the complete system, that is when the 3 decision trees are used jointly as shown in Algorithm 1 (in other words, the decision of the "no lookahead tree" conditions the decision of the "1-word lookahead tree", etc.). Figure 3 displays the distribution of the test data as a function of the delay needed to guarantee the POS tag stability. For each considered lookahead (0, 1 and 2, resulting in a maximum latency of 3 words), we also represent the remaining errors (FP, in yellow), that is the amount of words for which the synthesis has been wrongly triggered instead of being delayed. In 60% of the cases, the synthesis is triggered immediately (no lookahead) with 92% of the POS tag correctly estimated. In more than 30% of the cases, a lookahead of 1-word is needed to estimate the POS-tag with 95.4% accuracy. Finally, in 5% of the cases, the synthesis is delayed by at least 2-words (with more than 97.7% accuracy). When considering the combined accuracy of all decisions performed with a maximum latency of 3 words, the proposed adaptive latency approach performs a robust online POS-tagging with $\sim 90\%$ correlation with respect to the non incremental tagging.

3.2. Perceptual evaluation

The proposed iTTS system with adaptive latency delivers the synthetic voice in groups of words (between 1 and 4 words). This may result in a singular word grouping (i.e prosodic phrasing). To assess the quality of this grouping, we conducted a perceptual evaluation based on a ranking test. A set of 14 sentences extracted from the Combescure corpus [12] was synthesized using our HMM-based iTTS system for French [4] and 4 different strategies of word grouping (resulting in a total of 56 stimuli to rank):

- "WG1: One word per group" which corresponds to a

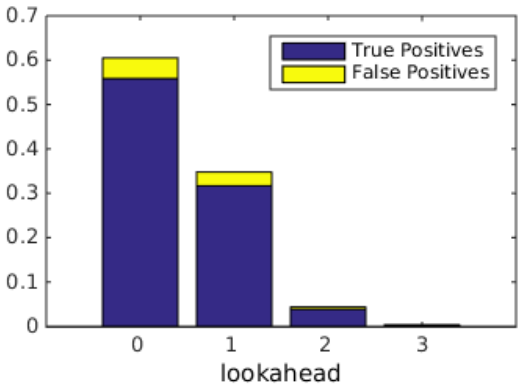

Figure 3: Distribution of the test words as a function of the lookahead needed to guarantee the stability of the associated POS tags.

word-by-word synthesis using no lookahead (e.g "This - summer, - the - children - etc.")

- "WG2: Random word grouping" obtained by replacing the output of each decision tree by a random binary value. This strategy is used as a reference condition (e.g "This - summer, the - children - etc.").
- "WG3: Expert-based word grouping" where 3 human experts were asked to delimit manually the most natural boundaries of each prosodic phrase, based on the semantic (e.g "This summer, - the children - will go - to the sea").
- "WG4: Adaptive latency iTTS" which is the word grouping resulting from the proposed method (e.g "This summer, - the children will - go to - the sea").

The duration of the silence between each word group is constrained so that the 4 versions of each sentence have all the same length (with minimum silence duration between each word chunk set arbitrarily to 300 ms). The listening test was done online by 20 native speakers of French, with no particular expertise in speech processing. The participants were asked to do the test in a quiet environment, with headphones. The presentation order of the stimuli was randomized for each participant. For each sentence, the participant were asked to score the different stimuli on a Mean-Opinion-Score (MOS) scale ranging from 1 to 5 (a set of 5 labels "very bad, bad, middle, good, very good" was nevertheless displayed in order to help the subject in the ranking process). The participant was allowed to play each stimulus several times. The statistical significance of the ranking score was assessed using Beta regression, considering the position of the stimulus on the scale as the variable to explain, the word grouping strategy as the explanatory variable (4-level factor), and both the *subject ID* and *sentence ID* as random effects (an Anova test was not suitable since the variable to explain was bounded).

As expected, the most natural word grouping is the one proposed by human experts (WG3), which can indeed rely on high-level semantic knowledge. Interestingly, the "one word per group" strategy (i.e. the strategy that leads to the most reactive system) was considered less acceptable than the random grouping (which was the reference condition). This result shows the importance of prosodic phrasing in incremental text-to-speech, where a tradeoff between reactivity and naturalness have to be found. Finally, and more importantly, the proposed adaptive-latency iTTS was ranked second. It was assessed significantly

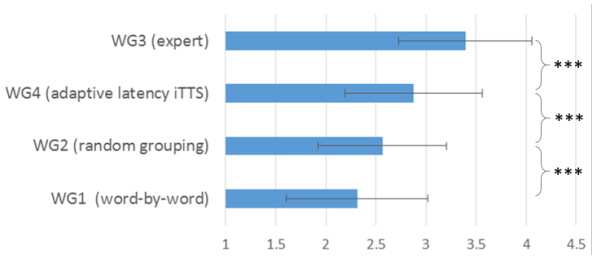

Figure 4: *Results of the perceptual listening test. Mean position on the X-axis of ranked samples with standard deviations, averaged across the listeners, for each word grouping strategy ("one word per group" (WG1), "random" (WG2), "expert" (WG3) and "adaptive-latency iTTS" (WG4, proposed method)(*** denotes statistical significance)*

better than the random grouping (and word-by-word synthesis), but also significantly lower than the expert-based strategy. This demonstrates the interest of the proposed approach while letting some room for improvements. To illustrate a possible limitation of the proposed method, let us focus on one stimulus which was ranked as "bad" by most listeners: the sentence "Il arrive en retard en ce moment" ("he arrives late these days"). For this stimulus, the expert-based word grouping (WG3) was *"Il arrive - en retard - en ce moment"* whereas the word grouping resulting from the analysis of POS tag stability gave *"Il arrive - en retard en - ce moment"*. This result in a non-natural prosodic phrasing, notably due to the third chunk "en retard en". It corresponds to the POS sequence "Preposition Noun Preposition" which is not a common prosodic unit in French. Therefore, the proposed word grouping strategy based on the sole POS-tag stability is an interesting but perfectible approach.

4. Conclusions and Perspectives

This article introduced a method for robust POS-tagging in the context of incremental Text-to-speech synthesis. The core idea is to assess 'on-the-fly' whether a POS tag in a given left-context is likely to be modified when more right-context becomes available, and if yes, to postpone the synthesis. This results in a new iTTS architecture where the synthetic voice is delivered in word chunks of variable length. Objective evaluation showed that almost 90% accuracy of true positives can be obtained with a adaptive lookahead between 0 and 3 words, for French.

Although demonstrating the pertinence of this morphosyntactic parsing for effective incremental speech synthesis, the perceptual evaluation of the resulting prosodic phrasing led to contrasting results. Future work will focus on improving this prosodic phrasing. Among other perspectives, we will notably combine the proposed approach with the predictive incremental parsing technique, recently proposed in [13]. Finally, as an incremental TTS synthesizer is primarily designed for casual conversation, we will also evaluate the performance of the proposed adaptive latency POS-tagger on other kind of text data, such as text-messages or tweets.

5. Acknowledgments

This work was funded by the project *SpeakRightNow* (AGIR program, Université Joseph Fourier, http://www.gipsa-lab.fr/projet/SpeakRightNow/). The authors would like to thank Sylvain Gerber for his help in the statistical analyses.

6. References

[1] J. Edlund, "Incremental speech synthesis," in *Proceedings of Swedish Language Technology Conference*, Stockholm, Sweden, 2008, pp. 53–54.

[2] D. Schlangen and G. Skantze, "A general, abstract model of incremental dialogue processing," in *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, Athens, Greece, 2009, pp. 710–718.

[3] T. Baumann and D. Schlangen, "The INPROTK 2012 release," in *Proceedings of the 12th Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Stroudsburg, PA, USA, 2012, pp. 29–32.

[4] M. Pouget, T. Hueber, G. Bailly, and T. Baumann, "HMM training strategy for incremental speech synthesis," in *Proceedings of Interspeech*, Dresden, Germany, Sep. 2015, pp. 1201–1205.

[5] N. Beuck, A. Köhn, and W. Menzel, "Decision Strategies in Incremental PoS Tagging," in *Proceedings of NODALIDA 2011*, Riga, Latvia, 2011, pp. 26–33.

[6] C. D. Manning and H. Schütze, *Foundations of statistical natural language processing*. MIT Press, 1999, vol. 999.

[7] K. Toutanova and C. D. Manning, "Enriching the knowledge sources used in a maximum entropy part-of-speech tagger," in *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, Stroudsburg, PA, USA, 2000, pp. 63–70.

[8] T. Brants, "TnT: a statistical part-of-speech tagger," in *Proceedings of the sixth conference on Applied natural language processing*. Seattle, WA, USA: Association for Computational Linguistics, 2000, pp. 224–231.

[9] P. Taylor, A. W. Black, and R. Caley, "The Architecture of the Festival Speech Synthesis System," in *Proceedings of the 3rd ESCA Workshop on Speech Synthesis*, Jenolan Caves, Australia, 1998, pp. 147–151.

[10] G. Bailly and M. Alissali, "Compost : un serveur de synthèse de parole multilingue," *Traitement du Signal*, vol. 9, no. 4, pp. 359–366, 1992.

[11] K. Tokuda, Y. Nankaku, T. Toda, H. Zen, J. Yamagishi, and K. Oura, "Speech Synthesis Based on Hidden Markov Models," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1234–1252, May 2013.

[12] P. Combescure, "listes de dix phrases phonétiquement équilibrées," *Revue d'acoustique*, vol. 56, 1981.

[13] N. Beuck, A. Köhn, and W. Menzel, "Predictive incremental parsing and its evaluation," in *Computational Dependency Theory*. Kim Gerdes, Eva Hajičová, Leo Wanner, 2013, vol. 258, p. 186.