# UNSUPERVISED DECOMPOSITION OF PHONEME STRINGS INTO VARIABLE-LENGTH SEQUENCES, BY MULTIGRAMS

*Frédéric BIMBOT, Sabine DELIGNE, François YVON.*

ENST / Télécom Paris, CNRS - URA 820,
46 rue Barrault, 75634 PARIS cedex 13, FRANCE, European Union.

## ABSTRACT

The multigram model allows the automatic extraction of variable-length regularities in strings of symbolic units. In this paper, we assess the multigram model as a phonotactic model. In our experiments on the MALECOT corpus, the multigram model outperforms the classical n-gram model for the description and the prediction of phoneme strings, measured in terms of test set perplexity. We also show that the model can be used to automatically derive segmental speech synthesis units.

## 1. INTRODUCTION

A string of graphemes or phonemes can be viewed as the result of a complex encoding process which maps a message into a stream of symbols. This string of symbolic units is far from being random, as the encoding process is subject to various phonotactical, lexical and syntactical constraints. In particular, combinations of letters form lexical items, which themselves are arranged according to grammar rules.

These constraints are responsible for a significant degree of redundancy in natural language symbolic representations such as phoneme strings or word strings. For instance, in the phonemic transcription of a conversation, all phonemes are not equally likely, nor are their two-by-two combinations (bigrams), their three-by-three combinations (trigrams), and so on...

This redundancy is partly exploited by probabilistic language models, among which the *n-gram* model [1] is very popular in language engineering. However, the underlying hypothesis of this model is that the probability of a given linguistic symbol (phoneme or word) depends on its $n$ predecessors, $n$ being fixed a priori and supposed constant over the whole text.

In opposition, the ***n*-multigram** model, recently developed [2] and extended [3], is based on the hypothesis that the dependencies between symbols are of variable-length (from 0, i.e independency, up to length $n$).

The multigram approach was previously tested with success as a ***language model***, i.e a model of word dependencies within a sentence [2][3]. In this paper, we report its performance as a *phonotactic model*, and we assess its application for the automatic extraction of formal speech synthesis units.

## 2. THEORETICAL ASPECTS

### 2.1. Formulation

In this section, we denote as $A = \alpha_1 \cdots \alpha_t \cdots \alpha_N$, a string of $N$ linguistic symbols.

The conventional n-gram model assumes that the statistical dependencies between symbols are of fixed-length $n$ along the whole sentence. The *likelihood* of $A$ is then computed as :

$$\mathcal{L}_{gr}(A) = \prod_{t=1}^{t=N} p(\alpha_t \,|\, \alpha_{t-n} ... \alpha_{t-1}) \qquad (1)$$

where $p(\alpha_t \,|\, \alpha_{t-n} ... \alpha_{t-1})$ is the conditional probability of observing symbol $\alpha_t$ given that the history of $n-1$ symbols $\alpha_{t-n} ... \alpha_{t-1}$ has occured[1].

The $n$-multigram model makes a different assumption : under this approach, a stream of linguistic symbols is considered as the concatenation of independent variable-length sequences, and the likelihood of the whole string is computed as the sum (or the maximum) of the individual likelihoods associated to each possible segmentation.

Let $\Delta$ denote a possible segmentation of $A$ into $q$ sequences $s_1 \cdots s_k \cdots s_q$. For instance :

$$\begin{array}{lcl} s_1 & = & [\alpha_1\, \alpha_2], \\ s_2 & = & [\alpha_3\, \alpha_4\, \alpha_5], \\ s_3 & = & [\alpha_6], \\ \cdots & & \\ s_q & = & [\alpha_{N-2}\, \alpha_{N-1}\, \alpha_N], \end{array}$$

The $n$-multigram model computes the likelihood $\mathcal{L}_\Delta(A)$ of string $A$ for segmentation $\Delta$ as the product of the probabilities of the successive sequences composing $\Delta$ :

$$\mathcal{L}_\Delta(A) = \prod_{k=1}^{k=q} p(s_k) \qquad (2)$$

[1]Further in this paper, we recall the link that exists between the likelihood of a model and the explanatory capabilities of the model in terms of prediction.



| Input character strings | |
|---|---|
| 1 | blessedisthemanthatwalkethnotinthecounseloftheungodly... |
| 2 | buthisdelightisinthelawofthelordandinhislawdothhemeditatedayandnight |
| 3 | andheshallbelikeatreeplantedbytheriversofwaterthatbringethforthhisfruit... |
| 4 | theungodlyarenotsobutarelikethechaffwhichthewinddrivethaway |
| **Output 5-multigram decompositions** | |
| 1 | bless ed isthe man that walk eth not inthe couns el ofthe un godly ... |
| 2 | but his d e light is inthe law ofthe lord and inhis law do th he medit at e day and night |
| 3 | and he shall be likea t re e plant ed bythe river sof water that bring eth forth his fruit ... |
| 4 | the un godly are not so but are like the ch a f f which thew in d drive th away |

Figure 1: Old Testament - King James Version (Psalms - first 5 verses). Character string decomposition using a 5-multigram model. Variable-length regularities are extracted without any supervision.

Denoting as $\{\Delta\}$ the set of all possible segmentations of $A$ into sequences of maximum length $n$, the total likelihood of $A$ is :

$$\mathcal{L}_{\mu gr}(A) = \sum_{\Delta \in \{\Delta\}} \mathcal{L}_\Delta(A) \qquad (3)$$

A decision-oriented version of the model can provide a maximum-likelihood decomposition of $A$ as the segmentation $\Delta^*$ with highest individual likelihood :

$$\Delta^* = \mathrm{Argmax}_{\Delta \in \{\Delta\}}\ \mathcal{L}_\Delta(A) \qquad (4)$$

and

$$\mathcal{L}^*_{\mu gr}(A) = \mathcal{L}_{\Delta^*}(A) = \max_{\Delta \in \{\Delta\}} \mathcal{L}_\Delta(A) \qquad (5)$$

For instance, with $A = abcd$ ($N = 4$) and a conventional tri-gram model :

$$\mathcal{L}_{3-gr}(abcd) = p(a|\phi\phi)\, p(b|\phi a)\, p(c|ab)\, p(d|bc)$$

with $\phi$ denoting the null symbol, whereas for a 3-multigram model :

$$\mathcal{L}^*_{3-\mu gr}(abcd) = \max \left\{ \begin{array}{l} p([a])\, p([bcd]) \\ p([abc])\, p([d]) \\ p([ab])\, p([cd]) \\ p([ab])\, p([c])\, p([d]) \\ p([a])\, p([bc])\, p([d]) \\ p([a])\, p([b])\, p([cd]) \\ p([a])\, p([b])\, p([c])\, p([d]) \end{array} \right\}$$

The maximum term indicates the maximum likelihood segmentation $\Delta^*$, for instance : [ab][c][d].

### 2.2. Algorithm

The algorithm for estimating the multigram probabilities from a training corpus proceeds iteratively. After the sequence probabilities have been initialised by counting all co-occurences of symbols up to length $n$, a forward-backward procedure is implemented to refine these estimates. Once convergence is reached, a Viterbi procedure provides the maximum likelihood segmentation, either on the training set, or on a test set, as in Equation (4). A full formulation of the algorithm and additional details[2] can be found in [2] [3].

### 2.3. Illustration

Figure 1 shows the result of the multigram decomposition of an english text[3], from which all spaces between words were removed. The set of linguistic units, in this case, is composed of the 26 lower case letters of the alphabet, and the corpus on which the probabilities are estimated contains approximately 200 000 characters. After 10 training iterations of a 5-multigram model, convergence is obtained, and the dictionary of typical sequences contains approximately 1100 entries.

In Figure 1, we indicate sequence borders by a space. Some typical english words or morphemes are automatically extracted. Some frequent combinations of small words are often merged (*inthe, ofthe, inhis,...*), while rare words tend to be broken into smaller units (*t re e, ch a f f,...*). Occasionally, an unappropriate segmentation occurs (*river sof, thew in d, ...*). Nevertheless, it is quite clear that the multigram model, though using no prior knowledge, extracts variable-length regularities which are strongly correlated with the morpheme structure of the input text.

## 3. EXPERIMENTAL PROTOCOL

### 3.1. Motivation

The experiments reported in the rest of this paper are carried out on phoneme strings. Our experimental protocol is designed to assess objectively the multigram model as a description of syntagmatic aspects in phoneme strings, and to investigate its potential application as a tool for deriving variable-length segmental units for speech synthesis. In a first series of experiments, the multigram model is used to predict phoneme strings

[2]In particular, in what concerns the *pruning factor* [2].
[3]An excerpt from the Bible.

and evaluated in terms of perplexity. In a second experiment, it is used to build variable-length speech synthesis units, by merging diphones which frequently co-occur together. In this last case, the evaluation criterion is the reduction in the number of concatenations per sentence.

### 3.2. Database

Our corpus is the MALECOT corpus. It consists of approximately 200 000 phonemes (13 000 sentences) which were obtained by a manual phonemic transcription of informal conversations in the French language [4] [5]. We split our corpus into a *training set* (first 150 000 phonemes) and a *test set* (last 50 000 phonemes). The phonemic alphabet is composed of 35 symbols, namely : a, i, e, ɛ, u, o, ɔ, y, ø, œ, ə, ã, ɛ̃, ɔ̃, œ̃, p, t, k, b, d, g, f, s, ʃ, v, z, ʒ, m, n, ɲ, l, ʀ, j, w, ɥ. Spaces are removed from the corpus, so that the word borders are unknown.

### 3.3. Perplexity

As an objective measure of the multigram model ability in representing sequences of phonemes, we use the *perplexity* measure [1]. The perplexity of a model $\mathcal{M}$ on a string $A$ is defined as :

$$X = 2^H \quad \text{where} \quad H = -\frac{1}{N} \log_2 \mathcal{L}_{\mathcal{M}}(A) \qquad (6)$$

where $N$ is the length of string $A$ and $\mathcal{L}_{\mathcal{M}}$ the likelihood provided by the model, as in Equations (1), (3) or (5), for instance.

Consider now a string $B$ of length $N$ generated by a memoryless source[4], from an alphabet of $X$ equiprobable symbols. As the probability of each symbol is $\frac{1}{X}$, the perplexity of $B$ is $X' = 2^{H'}$ where :

$$\begin{aligned} H' &= -\frac{1}{N} \log_2 \mathcal{L}(B) \qquad (7) \\ &= -\frac{1}{N} \log_2 \left[\frac{1}{X}\right]^N = \log_2 (X) = H \end{aligned}$$

Hence $X' = X$. Perplexity can thus be viewed as the randomness in the data that can not be predicted by the model.

If two models provide different perplexity values on a same *test* corpus, the one with lower perplexity can be considered as more efficient in explaining the underlying process which generated the data, whereas a lower perplexity on the *training* set only indicates a better ability in rendering the particularities of the training data.

Our first set of experiments consists in comparing perplexity values provided by n-gram and n-multigram models[5], on the phoneme strings in the MALECOT corpus.

[4] The symbols emitted by a *memoryless* source are statistically independent from one another.

[5] Using Equations (1) and (5) respectively.

### 3.4. Average number of concatenations

Segmental speech synthesis generally uses acoustic diphone units[6] which are concatenated to each other in order to reconstruct a speech utterance. In practice, some speech synthesis defects come from discontinuites at the level of the concatenations.

The application of the multigram model to strings of formal diphones can extract sequences of diphones which frequently co-occur together. Diphones within such sequences can then be advantageously merged together into a longer *multiphone* unit. For instance, if the diphone sequence <as><sj><jɔ̃> has a high probability, a quadriphone unit <asjɔ̃> can be created and added to the list of segmental synthesis units, which will avoid two concatenations during the synthesis process, each time this group of phonemes will be met. However, the set of multiphone units must result from a compromise between the economy in concatenations and the number and volume of acoustic units in the segmental dictionary.

In our second set of experiments, we evaluate the benefit of multiphone units as the average number of concatenations per sentence in the MALECOT corpus as well as in terms of number of segmental units. We also give a rough estimate of the acoustic storage requirements, measured as minutes of speech[7].

## 4. RESULTS

### 4.1. Phoneme sequence modeling

Table 1 summarises the results obtained in terms of training and test set perplexity for $n$-grams and $n$-multigrams (with $1 \leq n \leq 5$)[8].

With more than 8000 entries versus less than 3000, the trigram (and 4-gram) model provides a lower prediction capability than the 5-multigram model (perplexity of 10.1 (or 10.0) versus 9.4). The n-multigram model also shows good generalisation properties from the training set to the test set.

Figure 2 depicts an example of n-multigram decompositions of a french sentence in its phonemic form, from our test set in the MALECOT corpus. Here again, the phoneme multigrams show a striking correlation with morpho-lexical elements, especially for $n = 5$. They could prove efficient as word- or subword-like units for speech recognition.

[6] An acoustic diphone can be understood as a *domino* composed of the transition between the "center" of a phone and the "center" of the next phone.

[7] On the basis of 50 ms for a border phoneme and 100 ms for an inside phoneme.

[8] A pruning factor of 1.0 was used for the n-multigram training. A Good-Turing estimate was used for unseen n-grams, whereas a fixed penalty was used for unseen n-multigram of length 1. See details in [2][3].



| model order | n-gram model | | | | | n-multigram model | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $n=1$ | $n=2$ | $n=3$ | $n=4$ | $n=5$ | $n=1$ | $n=2$ | $n=3$ | $n=4$ | $n=5$ |
| training set perplexity | 25.8 | 13.7 | 8.9 | 5.3 | 3.3 | 25.8 | 16.8 | 12.4 | 9.9 | 8.3 |
| test set perplexity | 25.8 | 13.9 | 10.1 | 10.0 | 15.7 | 25.8 | 16.8 | 12.7 | 10.7 | 9.4 |
| number of entries | 35 | 937 | 8455 | 28786 | 51197 | 35 | 352 | 1119 | 1891 | 2683 |

Table 1: MALECOT corpus : training set perplexity, test set perplexity, and number of entries for the $n$-gram and the $n$-multigram models ($1 \leq n \leq 5$), for phoneme string modeling and prediction.

i l e t e v i d ã d a j œ ʀ k i l f o d ʀ a i v ə n i ʀ
il e te vi dã d aj œʀ k il fo d ʀa i və ni ʀ
il ete vid ã da jœʀ ki lfo d ʀa i vən iʀ
ilet evid ã d ajœʀ k ilfo d ʀa iv əniʀ
ilet evidã dajœʀ kilfo d ʀa i vəniʀ

il le et te ev vi id dã ãd da aj jœ œʀ ʀk ki il lf fo od dʀ ʀa ai iv və ən ni iʀ
ile ete evi idã ãd daj jœʀ ʀk kil lfo od dʀ ʀa ai iv vən niʀ
ile ete evid dãd da ajœʀ ʀki ilfo od dʀ ʀa ai iv vəni iʀ
ile ete evi id dãd dajœʀ ʀki ilfod dʀ ʀa ai iv vəniʀ

Figure 2: French sentence : "il est évident d'ailleurs qu'il faudra y venir" (MALECOT corpus - test set). 1- 2- 3- 4- and 5-multigram phoneme segmentations and 2- 3- 4- and 5-multiphone decompositions.

### 4.2. Multiphone units

Figure 2 illustrates the result of multiphone decompositions on a test sentence. Here, the elementary symbol is a diphone, and a sequence of diphones is represented as a tri-, quadri- or quintiphone. Table 2 reports detailed results concerning the number, size and repartition of multiphone units obtained by the multigram model, for different orders[9].

| multiph. order | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| nb diph. | 759 | 567 | 582 | 583 |
| nb triph. | 0 | 1828 | 938 | 859 |
| nb quadriph. | 0 | 0 | 1365 | 612 |
| nb quintiph. | 0 | 0 | 0 | 879 |
| total | 759 | 2395 | 2885 | 2933 |
| missing diph. | 466 | 658 | 643 | 642 |
| grand total | 1225 | 3053 | 3528 | 3575 |
| ≈ vol. in mn | 2 | 8 | 12 | 14 |
| nb conc tr. set | 13.6 | 7.5 | 6.2 | 5.8 |
| nb conc test set | 13.5 | 7.7 | 6.6 | 6.3 |

Table 2: See text.

Table 2 shows, for instance, that the set of 5-multiphones (last column) is composed of 583 diphones, 859 triphones, 612 quadriphones and 879 quintiphones, i.e a total of 2933 units. As 35 × 35 = 1225 diphones are necessary to guarantee a 100 % coverage of any text, 642 other diphones must be added to the dictionary, which leads to a grand total of 3575 units, i.e less than 3 times the number of diphones. The 5-multiphone set would require approximately 7 times more space than the diphone set, to be stored in its acoustic form. In counterpart, it can be expected that a sentence could be synthesized with twice less concatenations, as the average number of concatenations per sentence on the MALECOT test corpus falls from 13.5 to 6.3. This should have a significant impact on synthetic speech quality.

[9] A pruning factor of 2.0 was used for this experiment.

## 5. CONCLUSION

The multigram model provides a powerful framework for the unsupervised description, decomposition and prediction of phoneme sequences, and an interesting tool for the automatic design of segmental speech synthesis units. More generally, it appears as a relevant approach for the modeling of natural language syntagmatic aspects, which are usually based on variable-length schemes.

### References

[1] F. JELINEK: *Self-organized language modeling for speech recognition.* Readings in Speech Recognition. Ed. A. Waibel, K.F. Lee, Morgan Kaufmann Publ. Inc., 1990.

[2] F. BIMBOT, R. PIERACCINI, E. LEVIN, B. ATAL: *Modèles de séquences à horizon variable: multigrams.* XXèmes JEP, 1994. To appear in IEEE-SP Letters, as: *Variable-length sequence modeling : multigrams.*

[3] S. DELIGNE, F. BIMBOT: *Language modeling by variable length sequences : theoretical formulation and evaluation of multigrams.* IEEE-ICASSP, 1995.

[4] A. MALECOT: *New procedures for descriptive phonetics.* Papers in Linguistics and Phonetics to the Memory of Pierre Delattre. Mouton, 1972.

[5] J.P. TUBACH, L.J. BOE: *Un corpus de transcriptions phonétiques: constitution et exploitation statistique.* Report ENST-85D001, 1985.