SYNTHÈSE PHONÉTIQUE PAR ORDINATEUR

C.Y. Suen*, C. Charbonneau**, M.G. Strobel** et L. Santerre**

* *Département d'Informatique*
Université Concordia

** *Université de Montréal*

ABSTRACT

This computer program transcribes a written text into a phonetic text. Using the keyboard of terminal, the user can type in a phrase and the program will transcribe it, word by word, into phonemes for voice output by a speech synthesizer. Transformation is accomplished by analyzing the spelling of the word using a series of rules.

RÉSUMÉ

Le programme, tel qu'actuellement conçu, permet la transcription phonétique d'une entrée écrite sur clavier de terminaux. L'utilisateur peut ainsi taper une phrase et le programme, mot par mot, la transforme en phonème pour être éventuellement envoyée à un synthétiseur vocal. Pour effectuer cette transformation, le programme compare, lettre par lettre le contenu du mot avec une série de règles.

Introduction

The transformation of written communication into speech has many potential applications. One of the more immediate uses is the feedback given to blind typists, who are unable to proofread their typing.

The problem can be defined as the transformation of the set of individual characters which form a word into the string of phonemes associated with the pronunciation of this word. The problem however has several conditioning parameters which severely constrain simple solutions.

Pronunciation of identical groups of letters changes with the meaning of the word (e.g. owe - owl; cereal - cerebral), and even words with the same meaning are pronounced differently according to their grammatical status (e.g. "read" in the present and past tense). This dependency on context brings the problem of automatic phonemisation close to the formidable difficulties encountered by automatic translation. We suppose that both approaches will benefit from each other's advances and eventually merge into a system of automatic simultaneous translation.

Machine Transcription of Words

The present paper describes a project designed to transform typewritten words (and sentences) into sounds uttered by a synthesizer. The language chosen is French, the input medium is information stored in a computer or a "standard" terminal keyboard, the output is destined to go to a Votrax synthesizer. The program is implemented in PASCAL and runs on the CDC CYBER/175 at the University of Montreal in either batch mode or interactive time-sharing mode.

First, rules of pronunciation for the French language were formulated from standard lexica and adapted to computer processing. The 280 rules were assembled in a table which associates letters and groups of letters with their phonetic equivalent. 37 phonemes are necessary and sufficient to express our dictionary of approximately 15,400 words. These phonemes are tabulated below.

PHONEMES	EXAMPLES
AE	pAtte
AA	pAs
AX	dE
EY	Eté
EH	trEs
IY	sI
Y	bIen
AO	bOl
OW	mOt
UW	tOUt
W	OUi
HY	crU
UY	nUit
EU	fEU
OE	jEUne
AN	grANd
EN	pIN
ON	pONt
UN	brUN
G	Gant
ZH	Je
NJ	siGNe
SH	CHose
B	Bien
K	Croix
D	Dame
F	Frite
L	Loi
M	Mes
N	Nier

Second, a method for selecting the appropriate rule of phonetization was developed. Each rule is composed of three constituents: (1) The target letter or set of letters; (2) the immediate morphological environment, i.e. the letters or characters preceding and/or following the critical set; and (3) the phonetic symbol corresponding to the rule.

For example in the ending '...tion' the rule

[t]ion=/s/

consists of the critical letter [t], the environment 'ion', and the phoneme /s/. Application of this rule leads to a pronunciation of 't' as 's' in every instance where 't' is followed by the set 'ion'.

The program analyzes a word letter by letter and searches for an exactly corresponding rule. Search strategies are based on clusters of rules pertaining to vowels, diphthongs, and consonants which are frequently found in the environment of the target set. The aim of the search is to assemble a string of phonemes derived from the equivalence rules which represent the word phonetically.

Conflicts between rules are resolved by retaining the phonetic rule which accounts for the longest character string, first in the set of the target characters, then also in the environment.

If the tie persists, the rule encountered first is retained. This hierarchical ordering of equivalent rules is based on heuristics and is difficult to rationalize. In some instances the higher frequency of occurrence gives a better performance for a rule, in others it turned out to be preferable to retain a rule close to an exception.

Performance

The performance of the system was tested by comparing each string of phonemes produced by the program with the phonetic representation of each word in the dictionary. The first version of the system made 163 errors on 15,367 words for a success rate of 98.93%.

All 163 errors were caused by exceptions from the rules. We therefore implemented a scheme where each word (after parsing from the sentence) was first looked up in a table of exceptions before it was submitted to analysis by the list of phonetic rules.

A test of this arrangement showed an improvement of the success rate to 99.23% (119 errors) at a cost of an increase in the delay time of approximately 10% in the interactive mode.

Particularly resistant to a satisfactory solution in terms of both efficiency and economics are errors produced by pronunciation rules which cannot be derived from the morphology of the word.

For example, the word "divergent" can signify in French "they diverge" (3 person, plural, present tense) or that a process is diverging (adjective). Only the semantic context can determine the correct pronunciation. Obviously the attempt to provide phonetic transformations in real time would be doomed to failure if it were indeed a necessary condition to first understand the meaning of a sentence.

Discussions

We are presently experimenting with a psycholinguistic or rather a psycho-accoustic solution.

Based on the observation that every speaker has his/her personal accent (particularly in a multi-ethnic society with a high immigration rate) it seems feasible to let the synthesizer produce a sound intermediate to the two conflicting pronunciations in those special cases. The listener can and does activate learning mechanisms which supply the missing information from the real world to aid the perception process. The physical characteristics of the intermediate sound which will lead to the most rapid acceptance or the least disturbance and distortion is presently under investigation.

Another problem is caused by the special characters used in the French orthograph such as the accents (aigu, grave, circonflex), the c-cedille, and the tréma. At present we signal to the program the presence of a special character by adding to the letter in question a digit from 1 to 5. This preliminary solution is dictated by the Display Code character set of our PASCAL compiler. Obviously it introduces unwanted complexity into our program in addition to the limitation that mixed alphanumerics cannot be treated. However equipment and software adapted to this special requirement should be forthcoming in the near future.

Acknowledgement

This research was supported by the Department of Education of Quebec.

References

1. M. Divay and M. Guyomard, "Conception et réalisation sur ordinateur d'un programme de transcription graphémo-phonétique du francais," Theses, Docteur en Troisième Cycle, L'Université de Rennes, Mathématiques et Informatique, April 1977.

2. J.-S. Lienard, "Analyse, synthèse et reconnaissance automatique de la parole," Thèse de doctorat d'Etat, Université Paris VI, April 1972.

3. J.-S. Lienard and D. Teil, "Les éléments phonétiques et la traduction automatique du message écrit en message parlé," Automatisme, no. 10, Oct. 1970.