

【論文】

携帯電話を利用した吃音者用支援システムに関する研究

平田善之*・松本勝哉**

Support system for stutters by mobile phone

Yoshiyuki HIRATA*, Katsuya MATSUMOTO**

Abstract: There are various approaches to support handicapped persons by use of information system. One of the support system for stutters is named DAF(Delayed Auditory Feedback). It's not used commonly, because of expensive and difficult to use for beginner. The stutter persons have a problem that they cannot start to say the first word. Especially, it's serious at communication by the mobile phone. The support system for the stutter persons we presented in this paper was designed by use of the combination of i-appli (NTT DoCoMo), Web server and the database. We recognize this system is the middle way to our goal, but we think it passed though the first hurdle in this research.

Keyword : stutter, DAF, mobile phone, i-appli

1. 序論

障害者を支援するために情報システムを利用して様々な取り組みが行なわれ、障害者の生活の向上が図られている。言語障害である吃音に対する支援システムは、DAF(Delayed Auditory Feedback)装置があるが高価で訓練の必要があり、これは個人によって効果の差があるのであまり使われていない。

現在、携帯電話の普及により情報を交換する手段として電子メールが使われ、言葉が詰まる吃音者は電話の代わりにメールのやり取りによって容易に連絡を取り合えるようになった。しかし一方で携帯電話での会話も増え、最初の一言が言えない、自分の名前が名乗れないなどの解決していない問題も浮上してきた。

本研究では、吃音者の問題を解決するため、NTTDoCoMoのiアプリ、Webサーバ、及びデータベースを用いて吃音者を対象にした支援システムを構築した。吃音者側で自分の声の音声ファイルをあらかじめ携帯電話に保存登録しておき、必要に応じて相手に聞こえるように再生させることでこれらの問題の解決を試みた。携帯電話とWebサーバ、データベースとを連携させることでこれらの問題を解決できるシステム作成を目標とした。

2. 言語障害と本研究との関連

言語障害である吃音についての説明と吃音者の現状と問題点を述べる。また、吃音抑制装置であるDAFと今回のシステムの比較について説明する。

2.1 吃音

吃音とは話す言葉(音)が詰まったり、同じ言葉(音)をくり返すことをいう。アメリカの精神医学会が発行しているDSM-IV「精神障害の診断と統計マニュアル」によると成人の有病率は0.8%あり、吃音は発達障害に含まれる「コミュニケーション障害」のひとつで「通常、幼児期、小児期または青年期に初めて診断される障害」に分類されている。

2.2 吃音者の現状と問題点

吃音を治すため発声練習や呼吸練習、カウンセリングなど専門家による治療が行なわれてはいるが、吃音は人それぞれ症状が異なるため現在有効な治療方法が確立していない。吃音を矯正する装置(DAF:聴覚性遅延フィードバック、言語訓練装置)がある。DAFは個人によって効果の差や再発するといった結果もある。アメリカの研究では1/3の吃音者は確実に効果があり、1/3はやや効果(早口にならない、

* 工学研究科電気工学専攻2年

** 電気工学科

落ち着いて話せるなど)、約 70%に効果があったという報告がある。

吃音の問題点は、『声を出さなければ気付いてくれない』ということである。視覚障害者や聴覚障害者のように外見から判断することが難しく、言葉が詰まる場合とすらすら話せる場合があるため、話している相手が吃音を持っていることに気が付かない場合が多い。主な問題点として次のような例が挙げられる。

- ・電話で自分の名前が名乗れない。
- ・特定の言葉、最初の言葉が言えない。
- ・声が震えたり、うわずる。

これらの問題を解決するためのシステムとして、現在 DAF が使われている。

2.3 DAF と本システムとの比較^[1]

DAF と本システムの主な違いとして、DAF では吃音者本人が声を出して聞き取りながら話すのに対して、本システムは本人の音声を携帯電話に保存しておき必要に応じて再生させることである。DAF は吃音を矯正して治す機器であり訓練するための装置である。本システムは吃音者を支援するという面から、詰まる言葉を携帯電話に代わりに話してもらうというシステムにした。表 2.1 に DAF と本システムの比較について示す。

Table 2.1 Comparison between DAF and this system

	必要な機器	費用	その他
DAF	DAF 本体	高価	専門家のもとで訓練が必要
本システム	i アプリ対応の携帯電話	安価(ダウンロードの通信料)	機能・インターフェースに制限

3. 動作環境

本システムの構成にあたり必要な Web サーバ、Tomcat、MySQL、及び NTT DoCoMo の i アプリについて説明をする。また、i アプリと Web サーバの仲介役である Java サープレットを用いた連携についても述べる。

3.1 Web サーバ^[2]

Web サーバとは、HTML 文書や画像などの情報を蓄積しておき、Web ブラウザなどのクライアントソフトウェアの要求に応じて、インターネットなどのネットワークを通じて、これらの情報を送信する役割を果たす。初期の Web サーバは、あらかじめ用意し

ておいたファイルを送出する機能しか持たなかったが、最近では機能が増え、要求に応じてプログラムを実行し、結果をクライアントに送信する動的ページ生成の機能や、データベースと連携したトランザクション処理機能などを持つものも登場している。図 3.1 に Web サーバの動作を示す。中でも CGI や SSI などの拡張機能は歴史が古く、最も一般的に利用されている。Java 言語を利用した Java サープレットや JSP、Microsoft 社独自の技術である ASP などを利用した Web サイトも増えている。

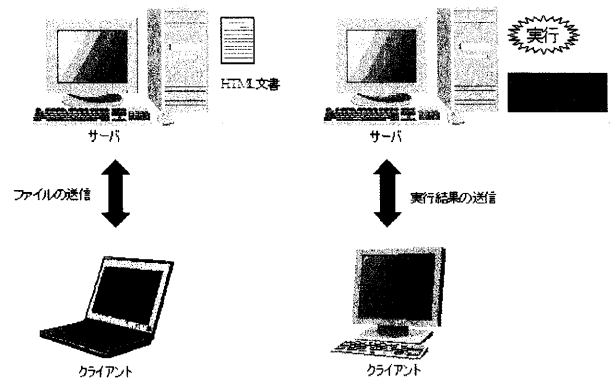


Figure 3.1 Role of Web server

3.2 Apache と CGI

Apache とは、NCSAHTTPD をベースに開発されたソフトである。その特徴として、軽快な動作と高い安定性を持ち、各 OS に対応するモジュールがフリーウェアとして配布されている。また、ユーザーからの要望によって、機能の追加・拡張が行なわれ、機能毎のモジュール化も徹底しており、ユーザーが新規機能を追加するための API も用意されている。

CGI は Common Gateway Interface の略語で、サーバマシン側で動くプログラムである。Web サーバが、Web ブラウザからの要求に応じて、プログラムを起動する。CGI を使うことによって、プログラムの処理結果に基づいて動的に文書を生成し、送信ができるようになった。図 3.2 に CGI の仕組みを示す。

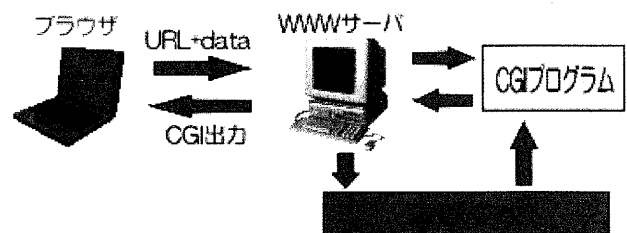


Figure 3.2 CGI process

CGI はプログラムが実行される分だけ、実行に時間がかかり Web サーバにも負荷がかかる。本研究の当初は CGI を使ったシステム構築をしていたが、サーバに負荷がかかるため Java サブレットを使うことに変更した。

3.2 Tomcat^[3]

Tomcat は、Jakarta Prokect が無償配布している JSP/サブレットを実行するためのアプリケーションで、代表的なアプリケーションサーバーのひとつである。Tomcat は単独で Web サーバとして動作することも可能だが、Apache や IIS のプラグインと動作できるようになっており、実際にはプラグインとしての利用が主流である。Tomcat の利点は、無償で使えることやオープンソースで Tomcat 自体のソースコードを見ることができることである。またポートは標準で 8080 を使用している。

3.3 MySQL^[4]

MySQL とは、TCX DataKonsultAB 社などが開発しているオープンソースの DBMS(DataBase Management System)である。その特徴を以下に記述する。

- ・MySQL は速度が非常に重視されており、MySQL の開発者は高機能なデータベースではなく、高速なデータベースを追求しており、速度面に影響のある機能の実装は極力避けている。そのため、検索やログインなど、至る所で非常に高速な処理を期待できる。データ件数が何千万にも及ぶ場合も高速にデータを扱うことができる。
- ・MySQL3.23.30からは正式に日本語に対応している。使用できる文字コードは EUC と SJIS である。値、およびテーブル名やフィールド名、LIKE 検索などに問題なく日本語を使用することができる。
- ・MySQL3.23.19からはライセンス形態が非常に柔軟になり、個人や社内で使用する場合 MySQL を無償で利用することができる。

本システムでは、利用者が吃音者を含む一般者であり、データが膨大になる可能性があるため、速度を重視した MySQL を採用する。

3.4 i-appli と i-appli の仕様(DoJa)^[5]

i-appli(i アプリ)とは、2001年1月に開始された、NTTDoCoMo の「i モード」対応携帯電話で利用できるアプリケーションサービスである。「503i」シリーズ以降の対応した携帯電話には、Java 言語で作成

したプログラムの実行環境(Java 仮想マシン)が搭載されており、Java 言語(の縮小版)で作成したアプリケーションソフトをダウンロードして実行することができる。i アプリはハードが携帯電話であるため次のような利点がある。

- ・アプリの配布が簡易である
- ・通信機能を標準装備している
- ・開発ツールが無償で提供されている

組み込み機器はそれぞれ特有の機能を持つため、それに対応できるように用途に応じた特殊の API(Application Programming Interface)が必要になる。この API をプロファイルと呼び、NTTDoCoMo の定める i アプリ専用のプロファイルを DoJa と呼ぶ。

また、i アプリを活用できる携帯電話は、i アプリの仕様と回線の種類によって以下の表のように分類できる(2005年3月現在)。

Table 3.1 The specification of an i-appli, and the kind of circuit

シリーズ	i アプリの仕様	回線の種類
901i シリーズ	DoJa4.0	FOMA
700i シリーズ	DoJa4.0LE	FOMA
900i シリーズ	DoJa3.5	FOMA
508i シリーズ	DoJa3.5	デジタル携帯電話(PCD方式)
505i シリーズ	DoJa3.0	デジタル携帯電話(PCD方式)

3.5 JAM ファイル

JAM(Java Application Manager)ファイルとは、携帯電話にインストールされている Java アプリケーションを管理する機能である。JAM の主な特徴として次のような機能がある。

- ・インストール前の検証・・・アプリケーションがダウンロード可能かを確認する。
- ・システムリソースの管理・・・スクラッチパッドの領域を確保し、ネットワークアクセスを制御する
- ・アプリケーションの状態管理・・・アプリケーションの状態(活性状態、サスペンド状態など)を管理する
- ・バージョン管理 制作日時・・・バージョンチェック・容量などにより、アプリケーションのバージョンアップの制御を行う

3.6 JAR ファイルとスクラッチパッド

アプリケーションを構成するクラスファイルとリソースファイルを圧縮形式で格納するための、標準

の Java アーカイブ機能によりパッケージ化したファイルを JAR (Java ARchive) ファイルと呼ぶ。i アプリ対応の携帯電話では、属性ファイルである ADF (Application Descriptor File) ファイルを先にサーバからダウンロードし、その i アプリが実行可能かどうかをチェックしてから、JAR ファイルをダウンロードする、という仕組みである。図 3.3 にダウンロードまでの流れを示す。

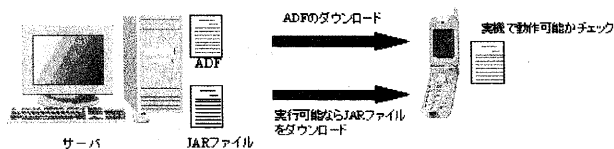


Figure 3.3 The download flow of an i-appli

また、スクラッチパッドと呼ばれる永続データを保存するためのデータ保存領域がある。JAR ファイルとスクラッチパッドの最大サイズを表 3.2 に示す。

Table 3.2 The maximum size of JAR and a scratchpad

	JARファイルのサイズ	スクラッチパッドのサイズ
901i・900i シリーズ	100K(102,400)バイト	400K(409,600)バイト
700i・506i・505i シリーズ	30K(30,720)バイト	200K(204,800)バイト

3.7 音声データ

i アプリでは、i メロディ形式と呼ばれるデータ形式の音声データを再生させることができ、NTTDoCoMo の端末でサポートされているメロディファイルは MLD ファイルと呼ばれる。MLD ファイルを作成するには、MIDI (Musical Instruments Digital Interface) または WAVE 形式のサウンドファイルを変換して作成する。MIDI は一種の音譜データで、各機種が持つ音色を使って演奏するためのデータである。そのため、ファイルサイズを小さくおさえることができる。WAVE は音声データで、ボイスも再生させることができるがファイルサイズが大きくなる。

3.8 Java と Java サブレット^[6]

i アプリの開発では Java というプログラミング言語を利用する。Java はサン・マイクロシステムズ社が開発したプログラム言語で、Windows や Mac、ザウルスや WindowsCE といったような PDA (携帯情報端末) にまで、様々な環境におけるアプリケーション開発に利用されている。Java は『特定の機種に依存することなく、基本的にはどのようなコンピュータで

も動作する』という特徴を持っている。Java によるプログラミングでは、Java バイトコードと呼ばれる中間形式にいったん変換し、実行時に Java 仮想マシン (JVM: Java Virtual Machine) と呼ばれるソフトウェアによって、そのプラットフォームで実行可能な形式に変換して実行するという方法をとっている。

Java サブレットは、サーバ側で動作する Java アプリケーションである。サブレットを追加することにより、Web サーバの機能を拡張することができる。CGI と同様の機能を Java で実現したもので、CGI ではリクエストの度に新しいプロセスが生成されるが、Java サブレットでは一つのインスタンスを共有するため負荷が少なく高速に実行できる。当初は CGI を利用したシステムを考えていたが、今回のシステムでは負荷が少ない Java サブレットを採用した。

3.9 Web サーバと i アプリの連携

i アプリでは、携帯電話機が持っている無線通信機能を用いて通信を行なうことができる。ただし、セキュリティ上の理由から次の制限が課せられる。

- ・使用可能なプロトコルは http と https のみ
- ・通信できる先は i アプリをダウンロードしたサーバのみ。

i アプリのネットワーク設定でサーバの URL を指定し、プログラム内で Java サブレットの URL を指定すれば利用可能になる。携帯電話から MySQL データベースへのテーブルの作成やファイルの登録・削除などは Java サブレットを介してアクセスを行なう。携帯電話から直接サーバへアクセスしてファイルをダウンロードするのは可能だが、サーバ内に置いてあるファイルに不正アクセスされる可能性があるため、ファイルをデータベースで管理することでセキュリティ面でも効果がある。また、ファイルが膨大になると本人の所有しているファイルが分からなくなるので、検索や集計ができるデータベースを利用することでファイル管理が容易になる。

4. 本システムの構成

4.1 システムの必要性

吃音で特に問題なのが電話での会話である。聞き手が目の前にいて言葉が詰まる場合、本人が話そうとしているのが相手から見ても一目でわかるが、電話の最初の一言が出ない場合、話し手 (吃音者) の姿が見えないので話そうとしていることがわからない場

合がある。

今回の研究では、最初の一言が出ないことに注目して、本人の代わりに携帯電話に話してもらおうというシステムを考えた。吃音を治す目的ではなく、言葉が出ない部分をシステムが支援することで上記のような問題を解決できると考えている。

4.2 システムの概要

本システムは携帯電話の i アプリ、Web サーバ、及び MySQL データベースから構成されている。システムの概要を図 4.1 に示す。また、本システムの携帯電話は NTTDoCoMo の FOMA を前提にして話を進めて行く。

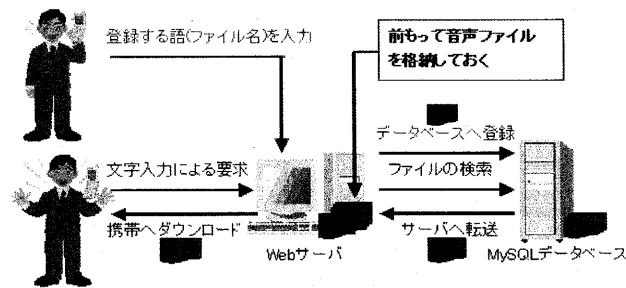


Figure 4.1 The support system for stutters

4.3 インターフェース

現在、主として使われている NTTDoCoMo の FOMA の画面サイズを表 4.1 に示す。携帯電話でアプリケーションを作成する場合、画面サイズの制約が大きいため操作性を考慮したインターフェースの作成が重要になる。

Table 4.1 Screen size of each model

機種	画面サイズ(pixel:横×高さ)
D900i	240×270
F900i	230×240
N900i	240×269
P900i	240×266
SH900i	240×252
D901iS	230×240
F901iS	230×240
N901iS	240×270
P901iS	240×270

今回の i アプリの画面構成は図 4.2 のようになる。操作性を早めるためなるべく画面をスクロールさせずにそれぞれを一画面で収まるように作成した。

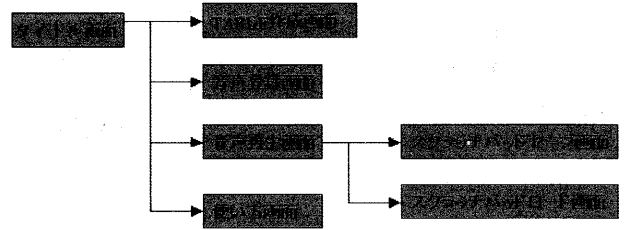


Figure 4.3 Screen composition of an i-appli

4.4 スクラッチパッド

FOMA のスクラッチパッドは 400K バイトある。FOMA の i アプリではスクラッチパッドを 16 分割できるようになり、本システムでは 2 個をダウンロード直後のデータ用とファイル名用に使い、残りの 14 個をデータとファイル名保存用として利用している。表 4.2 にスクラッチパッドの詳細を示す。

Table 4.2 Details of a scratchpad

スクラッチパッド番号	サイズ(byte)	利用方法
ScratchPad0	32768	最初のデータのダウンロード
ScratchPad1	256	最初のファイル名のダウンロード
ScratchPad2	32768	データ保存
ScratchPad3	256	ファイル名保存
ScratchPad4	32768	データ保存
ScratchPad5	256	ファイル名保存
ScratchPad6	32768	データ保存
ScratchPad7	256	ファイル名保存
ScratchPad8	32768	データ保存
ScratchPad9	256	ファイル名保存
ScratchPad10	32768	データ保存
ScratchPad11	256	ファイル名保存
ScratchPad12	32768	データ保存
ScratchPad13	256	ファイル名保存
ScratchPad14	32768	データ保存
ScratchPad15	256	ファイル名保存

4.5 データベースの構成

本システムでは、音声ファイルの管理のために MySQL を利用している。データベースはあらかじめ決められており(今回のデータベース名は test)、i アプリからデータベース test にアクセスして任意のテーブル名でテーブルを作成することができる。テーブルの作成の流れを図 4.4 に示す。

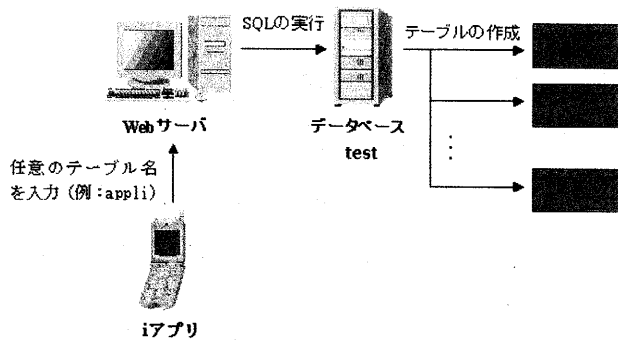


Figure 4.4 Creation of a table

テーブルの例として表 4.2 にテーブル名 appli を示す。今回は mld ファイルを扱うので blob を使用する。blob は大量のバイナリデータを格納することができるデータ型である。

Table 4.2 The contents of the table appli

	フィールド名	データ型
名前	name	varchar(32)
音声	voice	blob

4.6 データ処理の流れ

次に i アプリから MLD ファイルのデータベースへの登録・削除、携帯電話へのダウンロードまでのデータ処理の流れについて説明する。i アプリのネットワーク機能を使うことで、Web サーバの Tomcat と通信を行いデータベースにアクセスを行なう。i アプリでは 1 回の通信で送受信できるデータの大きさに制限があり、制限を越えるデータを送受信しようとする場合、何回かの通信に分割する必要がある。送受信できるデータ量の最大値を表 4.3 示す。

Table 4.3 Maximum of value which can be transmitted and received

	受信	送信
504i 以降	10KB	5KB
505i	20KB	10KB
900i	150KB	80KB

また、本システムで作成したプログラムの構成を図 4.5 に示す。Java サーブレットは Tomcat の webapps フォルダに格納され、Tomcat を起動させることで有効になる。

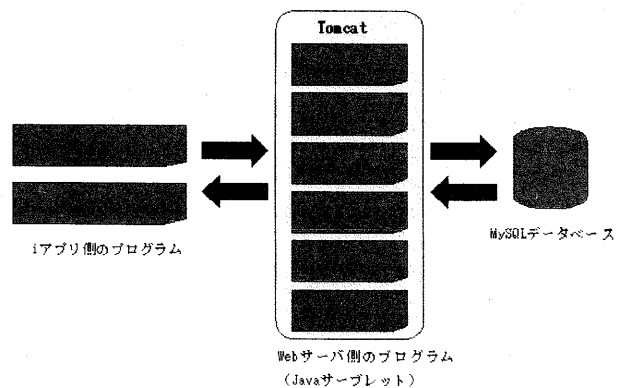


Figure 4.5 Composition of a program

4.6.1 MLD ファイルの登録・削除

MLD ファイルの登録までの流れを図 4.6 に示す。登録までのデータ処理の流れは次のようになる。

- ①ユーザーは i アプリから Java サーブレットに対して必要な MLD ファイルの登録、または不必要な MLD ファイルの削除を依頼する。
- ②Java サーブレットは指定されたファイル名がサーバ内にあるか検索を行なう。
- ③ファイル名と指定された語が一致した場合、Java サーブレットに転送される。
- ④Java サーブレットを介してデータベースに登録される。データベースに登録を行なった後ファイルのダウンロードが可能になる。

下図では例として『こんにちは』というファイル名で登録を行なう。

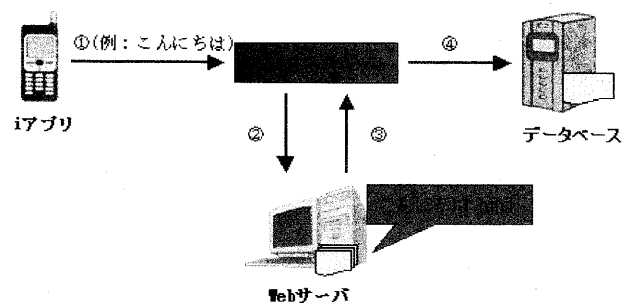


Figure 4.6 Registration of a MLD file

4.6.2 MLD ファイルのダウンロード

MLD ファイルのダウンロードの流れを図 4.7 に示す。ダウンロードのデータ処理の流れは次のようになる。

- ①ユーザーはダウンロードする語を Java サーブレットに依頼する。

- ②Javaサーブレットはデータベースに対して検索を行なう。
- ③指定された語とデータベース内にあるファイル名が一致した場合、Javaサーブレットに転送する。
- ④携帯電話へダウンロードが行なわれる。
- ⑤ダウンロードされるとiアプリのスクラッチパッドに保存され、ファイルは7つまで保存可能である。

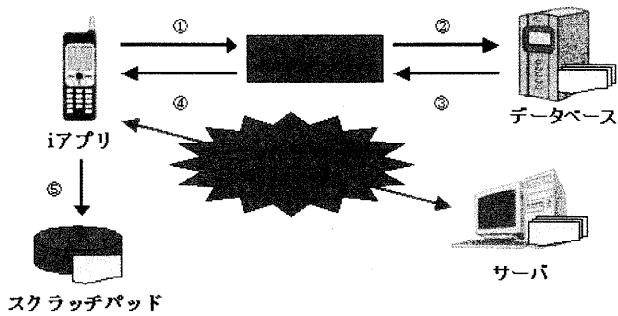


Figure 4.7 Download of a MLD file

4.7 本システムのアンケート調査

システムの利用対象者である吃音を持っている人達に対して20代30代以上の各10人にアンケートを行なった。性別、職業を問わず年齢別に調査を行い、見易さ、操作性、機能性を評価してもらった。それぞれ5段階評価で採点を行い、各評価の平均値を小数点第1位までとった。また、iアプリを利用したことがある人の割合をとることで、20代と30代以上の人達の間にもどれだけ差が出てくるのか考慮する。これらをまとめた結果を表4.4に示す。

Table 4.4 The evaluation of the system

年齢	見易さ	操作性	機能性	iアプリの利用者(%)
20代	3.4	4.3	3.6	40
30代以上	2.9	3.4	2.9	30

以上の結果より、30代以上に比べて20代の人達は携帯電話を使い慣れていることもあって、見易さ、操作性に関しては評価が高いことがわかった。見易さ、操作性に関しての要望やコメントには次のような意見があった。

- ・視認性は高いが、字が小さく見づらい(ボタンに書いてある文字が見えない)。
- ・画面が暗い、デザイン性よりも見易さ重視。
- ・もっと明るい色を使って欲しい。

機能性に関する評価は20代の方が若干高いが、年

齢に関係なくデータベースやテーブルといった知識がない人が多く、利用の初期段階ではわかりづらいといった意見が聞かれた。また、機能性に関する主な要望として、次のような意見があった。

- ・ひとつの画面で複数のファイルを再生させることができた方がよい。
- ・保存できるファイル数を増やして欲しい。
- ・音声再生までの操作を簡略化して欲しい。

システム作成の当初は、携帯電話に備わっているボタンに対応したアプリケーションを作成していたが、アプリケーション側のボタンやテキストボックスを小さく作る必要があり、ひとつの画面で表示できないときはスクロールしなければならぬ視認性、操作性が悪くなった。特に携帯電話のiアプリを使い慣れていない人にとっては操作性、視認性が重要だと考慮し、機能を分担させることでひとつの画面で単体の機能を持たせることにした。

保存できるファイル数に関しては、iアプリの保存できるメモリー数や仕様の制限からファイル保存の増大は難しい。今回のシステムでは、ひとつひとつのファイルのサイズが一定していないため、スクラッチパッドを16分割するときには余裕を持たせてメモリーを確保している。

音声再生までの操作の簡略化については、現時点ではロード画面を開き登録されているファイルをひとつだけロードし、再生画面に戻り再生させるという操作を繰り返す。本システムは電話の最初の一言を代わりに話してもらうという目的で、この目的であれば上記のような操作は問題ない。しかし、会話の途中で利用することがある場合には、一旦通話を止めて画面を見ながら操作しなければならないので問題が出てくる。これは最初の要望である『ひとつの画面で複数のファイルを再生させることができた方がよい。』に関連して、インターフェースの工夫が必要になる。

iアプリの利用者の割合に関しては、アンケートを取った人が皆DoCoMoユーザーではなかったことから、20代と30代以上の割合に差が出なかった。30代以上のiアプリ利用者は、40代・50代の人達がiアプリを使ったことがなく、この結果から、40代・50代の人達からの見易さ、操作性の評価が悪く平均値を下げる原因になったと考えられる。

4.8 本システムの検討と課題

4.7 で述べたように、本システムにはまだ改善のための課題がある。特に、30代以上の人達から多かった意見が、『全体的に文字が小さくボタンに書いてある文字が見にくい』ということである。i アプリを使ったことがない 40 代、50 代の人達が使いやすく見易い画面設計を考慮しなければならない。

もうひとつは、通話中に i アプリを同時に使うことができない点である。これは現在の携帯電話の性能上簡単に変えることはできないが、通話相手に i アプリで再生させた音声を伝える方法として、再生したデータを他のデバイスに一時保存しておき時間差を利用して相手に伝えることでこの問題点は回避できる。固定電話では、携帯電話を受話器の側に持っていき、音声を再生させることで相手に伝えることが可能である。図 4.10 に固定電話の場合と携帯電話の場合の方法を示す。

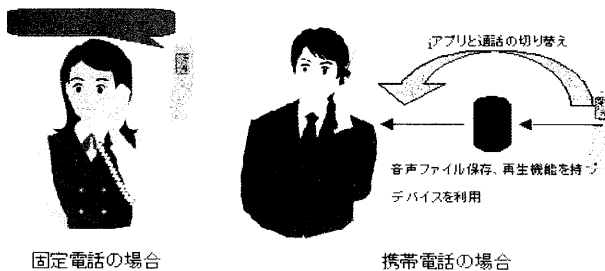


Figure 4.10 Combination with telephone and a Cellular phone

5章 結論

当初の目的である電話での最初の一言を支援する点については、携帯電話に7つの言葉を保存することができるので、会話の始めでよく使う言葉を登録しておく必要に応じて音声を再生させ相手に伝えることで目標は達成できたと思われる。また、携帯電話に保存されていない言葉をデータベースからダウンロードすることで個々の吃音者に適した言葉を選べる点についても利便性はある。本システムの現段階では、携帯端末から音声再生されるだけなので相手に言葉を伝える方法には限りがある。ここで提案したシステムは、吃音者用として作成したものはあるが、まだ完全ではなく様々な課題や問題点が残された。

本システムを吃音者の人達に使ってもらいアンケートを採った結果、システムにさらに解決すべき課題が出てきた。30代以上の人達になると携帯電話の i アプリに慣れていない人がいるので、画面(文字)

が見にくい・ボタンの操作が分からないなどの意見があった。また、保存できるファイルの数を増やして欲しい・音声再生までの操作を簡単にして欲しいなどの要望もあり、利用する人達にとって使いやすいインターフェースを考慮する必要があることが分かった。

システムの課題としては、携帯電話で再生させた音声ファイルを一時保存でき、再生できるような新たなデバイスを作成することが考えられる。

本システムでは最初に、電話で話す最初の言葉を携帯電話に代わりに話してもらうことによって吃音者を支援しようとした。しかし、会話の途中で言葉が詰まる場合には必ずしも使いやすいシステムとはなっていない。その場で使いたい言葉が携帯電話の中に入っていない場合は支援することができない。また再生までの操作に時間がかかってしまってスムーズな会話が成り立たないなどの問題もある。こうした問題を解決できるように、機能面の強化や操作性の向上が今後の必須課題である。

参考文献

- [1] <http://www.shinridock.com/daf1.htm> (2005-12)
- [2] 星野 努:「PHP+MySQLで作る最速Webシステム」, 1章 - 3章, 技術評論社(2003)
- [3] <http://e-words.jp/w/Tomcat.html>, IT用語辞典(2005-12)
- [4] 大久保 隆浩:「PHP・mysqlによる就職活動支援システムに関する究」, 論文 p.5(2005-12)
- [5] 布留川 英一:「i アプリゲーム開発テキストブック」, pp.17-19, 毎日コミュニケーションズ(2005)
- [6] <http://homepage2.nifty.com/S-Dream/homepagejava/java5.htm>, Java・i アプリ入門(2005-12)