

Supporting IRers through Development of an IR Data Check Application

Kunimasa Yamada*

Institute for the Advancement of Higher Education, Hokkaido University

IR データチェックアプリの開発による IR 担当者の支援

山田 邦雅**

北海道大学高等教育推進機構

Abstract — The Japanese situation of institutional research (IR) in the university is different from that of the United States because a campus-wide database system has not been constructed. Therefore Japanese IRers expend substantial manpower on efforts to collect data from each faculty member and consolidate them. For IR to demonstrate its true value by utilizing data, I think support of IRer is necessary. The most inefficient work of the Japanese IRer is the basic tasks that should be performed by a computer. Therefore I, a beginner in programming, developed an application that checks for incomplete IR data for IRers. When I checked the IR data of the last year with this application, I found that the values of 4.3% of the cells were incomplete and our IRer understood them. I also checked whether this application was user-friendly by using Fogg's consumption model.

(Accepted on 31 January, 2017)

1. はじめに

日本における大学 IR (Institutional research) は急速に普及し始め、何らかの形で IR 活動を行うようになっている大学が急増している (文部科学省 2012)。2014 年度からは、日本私立学校共済・振興事業団による私学助成において IR の実施が評価されるようになり、高等教育機関全体として IR が推進される傾向にある。また、IR は 2002 年の中央教育審議会答申「大学の質の保証に係る新たなシステ

ムの構築について」を受けて重視されてきたデータによるエビデンスをサポートするものとしての期待が大きく、文脈的に受け入れやすいものであったという背景もあった。

このように、現在の大学 IR には強い追い風が吹いているということはできるが、内情はそれほど楽観できるものではない。それは費用対効果の観点で評価すると、まだ必ずしも効果が勝っているとは言えない状況が見え隠れしているからだ。もちろんデータの集中管理の利点が完全に否定されることは

*) Correspondence: Institute for the Advancement of Higher Education, Hokkaido University, Sapporo 060-0817, Japan
E-mail: yamada@high.hokudai.ac.jp

***) 連絡先: 060-0817 札幌市北区北 17 条西 8 丁目 北海道大学高等教育推進機構

ないであろうが、日本の大学において IR の有用性が認められるかどうかは、これからにかかっている。

2. IR 担当者の現状

大学 IR は米国の取り組みを追従する形で導入されたものであるが、日本固有の問題によって米国よりも低次元での葛藤が起こっている。それは「どのように活用するか」ではなく「データを用意する」というレベルの業務である。

まず、日本では大学の学部縦割り型構造により、データは大学内で分散しており、集めるだけで大きな負担となってしまう。米国の IR 業務に携わる柳浦（2013）は米国の大学において「学内のデータは各部署から集めてくるものではなく、予めデータベース上に集約されているもの」としており、スタート地点からして大きく異なっている。また、日本の IR の現状について「ある部局はデータをシェアすることを拒否もしくは無視したり、またはデータを提供することには理解を示したものの、無数の書類を提出した上でようやく必要なデータが受け取れる、というようなつれない対応が当初の IR に対する反応であった。そのような状況下では、日本の初期段階の IR は実践段階に移行することは至難の業であったといえる。」と日本における IR の導入期の実情を述べている。

しかし、一度各部局とのコネクションができてしまえば昨年度と同様に業務を進められ、負担は軽減されるはずである。ところが、これで障壁がなくなったわけではない。柳浦は「現時点では、多くの大学が、各部局のデータ担当者から、部局の所有するデータベースから抽出されたデータを CSV ファイルで受け取り、そうして集まったファイルを IT 担当者が組み合わせて串刺しデータを作り、IR がそれを受け取ってデータ分析を行う。いわば、ひとつの分析を行うまでに、複数人の手を経ている。これでは非効率であるし、ファイル作成過程でエラーが出やすい。」と指摘している。つまり、データが統一されたデータベースに集約されている、または IR 担当者が各部局のデータベースに直接アクセスしてデータを抽出できる状況ではなく、データをもらっ

て自分で繋げている状況では整形に大きな労力を要してしまうのである。特に他大学との比較を目的の 1 つとして構成されるコンソーシアム型 IR では、一定のデータクリーニングが行われていることを前提としてデータを受領し、基準を満たさなければ受け付けないケースもあり、IR 担当者の負担はさらに大きくなる可能性がある。

P. T. Terenzini (1999) は IR 担当者に求められるスキルとして technical intelligence, issue intelligence, contextual intelligence の 3 つを示しているが、これは米国での話であり、日本ではその前段階にあたるスキルや忍耐が求められるのが現状である。また、この 3 点は IR オフィスのメンバーや大学教育研究センターのような部署の者が支援することも可能であるが、データの準備については上述したように多くの人を経してしまうことで非効率化する要素があるため、支援という介入が難しいのである。

3. IR 担当者の支援

このように、まだ黎明期にある日本の大学 IR であるが、費用対効果としての費用（労力）が大きくなりすぎて、活用実績での真価が評価される前に衰退してしまうのでは高等教育の発展においてマイナスである。活用実績で勝負する土俵に上げるため、日本の大学においてデータベース的整備がなされるまでの間、IR 担当者の支援を行うことは重要であると考えている。

また、1 つの大学が IR を上手くこなすことができても IR 文化の育成にはならない。汎用的な観点での支援を視野に入れて IR 担当者の支援を考えたい。北海道大学は大学 IR コンソーシアム（大学 IR コンソーシアム 2012）を教学 IR のベースの 1 つとしている。このようなコンソーシアム型 IR における IR 担当者全体像を視野に入れての支援を検討する。

まず、IR 担当者全体を対象としたときの 1 つの問題点は、担当者が多様かつ流動的であることである。「平成 26 年度大学における教育内容等の改革状況について」（文部科学省 2012）によると、IR に専任教員を置く大学は 6.3% しかない。また、IR 担当者は事務職が中心であり、数年周期で配置部署が入れ替

わる大学も多い。それゆえ、今日の日本の大学における IR 担当者は様々なレベルの人材によって構成されているのが現状である。これはコンピューターリテラシーでいうと MS Excel の基本操作程度のレベルまでしか求めることはできないと考えた方がよい。現に、米国のコミュニティーカレッジの IR 担当者ですら、MS Excel, MS Access, SPSS, SAS, ERP のソフトウェアで上級者と回答したのは Excel が 79.3%であったのに対し、他は全て 4 割にも満たない (Hagedorn & Coogan 2007)。

実際、本学の教学 IR 担当者のデータ整形業務では、要求されたデータの定義に合わせることに苦勞が多く、その確認も目視で行わざるを得ない状況である。例えば、Excel においてセルの値を「1」とすべきところが「1 」のように余分なスペースが混ざっている場合や、そもそもデータの定義を勘違いして理解している場合には、指摘されなければ修正できるものではない。しかもこのようなデータの不備は頻繁に発生する。

4. アプリの開発

IR データ整形業務において効率の悪さが顕著なものは、コンピューターが得意とする単純作業を人間が実施している場合である。例えば、余分なスペースを見つけなければならない場合、検索操作で簡単にできるが、実際のデータ整形では余分なスペースがあるかもしれないと思うまでにかなりの時間を要するのである。このように、検索という操作だけではなくチェックすべき項目が他にもたくさんあり、そのラインナップも自動化することが大きな支援となるのである。

そこで、本取り組みでは、揃えるべきデータの書式に対して考えられるチェック項目のラインナップを全レコードに対して行い、あり得ない値を検出してセルを指定する形で報告するアプリの開発を行い IR 担当者を支援してみた。なお、本取り組みは、アプリ開発を専門とせず趣味的にプログラミングを行う程度の初級レベルの筆者が作成するという制限の基で行うものである。そもそも、アプリ開発は高額なものであり、そのような大きな資金を前提とした

特異な状況は想定しない。

5. 支援アプリの機能

アプリの機能や特徴を以下に述べる。

1) フォーマット準拠の専用アプリ

特定の IR データに対する専用アプリにすることでほとんどの操作を自動化できるため、ユーザーに求める操作を極力排除した。これにより、メニュー、ボタン、文字入力用テキストボックスなどを一切持たないインターフェイスを実現している。これは使い方の習得も操作も必要としないことのあらわれである。実際の利用法は、データファイルの CSV または XLSX ファイルをドロップするだけでチェックが行われ、フォーム上に表示されたチェック結果はウインドウを閉じるとテキストファイルに保存されるようにした。

また、専用アプリとすることで、IR システム側が要求する基準を超えた厳しいチェックを行うことができ、データクリーニングの不備により撥ねられることを防止することはもちろんのこと、それを越えたデータの信頼性向上を行うことができる。

2) 凡ミス防止

一般に、受入れ側のシステムはデータの書式のみが受け入れ基準となっていることが多く、ヒューマンエラーが入る余地が大きい。たとえば、学生の生年が 1800 であっても 4 桁の実数であれば受け入れてしまうような場合である。この対策として、書式だけではなく数値として範囲が制限できる場合は、その範囲内かもチェックするようにした。また、「生年」と「入学年」のような項目間の整合性チェックを入れることでさらなる強化を行った。

3) 即時結果表示

近年は CPU の処理能力が高く、初級レベルのプログラマによるコーディングでも実用レベルの処理

能力を持ったアプリを作ることができることが多くなった。ソフトウェア開発は素人が行えるものではなかったが、2020 年からは小学校でプログラミング教育が必修化され、リテラシー化されてゆく可能性がある。専門ではなくリテラシーレベルのプログラミング能力で小規模開発ができ、また、使用レベルの処理速度をもつことができる時代になってきている。

また、プログラミングを行うにあたって、利用できる豊富なライブラリが取り揃えられている場合があり、積み木のように組み合わせる形でのプログラミングが可能となってきた。プログラマ側の書くコードが減ることに伴って、下手なコードも減少し速度を上げる結果につながっている。

また、.net framework が LINQ というデータ抽出表記法に対応したことにより、SQL と同レベルでデータの抽出式をコードに記述できるようになったため、初級プログラマが簡単にデータ処理プログラミングできるようになった。LINQ が無ければコードの肥大化、パラメータの扱いによるバグなどで、初級プログラマへの負担は大きいものとなる。

このように、十分な処理速度が確保できるため、手順を 1 つに集約して利用者の操作を排除し、また、一度に全結果を一覧表示するような処理も可能である。

4) 無料

アプリの作成は、業者に依頼すれば言い値的に高額なものとなる。また、保守が必須になる性格のものであるため、なおさら高額な契約となってしまう。

現在まで一般の人は大きな負担となる単純作業があったとき、たとえそれがコンピューターの得意とする処理であっても自らのアプリ開発を行うという選択肢はなかったであろう。プログラミングリテラシーの普及によっては、大きな単純作業負担とアプリ開発を天秤にかけられる可能性もでてくる。今回は、初級レベルの筆者による開発が可能であった。

5) スタンドアロンで独立に使える

IR は組織の機密情報を扱い、基本的に外部に出す

ことが禁じられているため、アプリはどのような形態でもよいわけではない。特に、コンソーシアム型 IR の IR 担当者全体の支援を考えたとき、外部サーバを経由するクラウド型は不適である。そもそも、IR に関してはグローバルネットワークから分離されたパソコンで処理されることは十分にあり得る。完全にスタンドアロン型で構成することが望まれる形である。

6) インストール不要

IR を扱うアプリといえばデータベースと連携したものが一般的であろう。しかし、データベースソフトウェアの扱いや SQL の使用は現在の IR 担当者の守備範囲からは外れるものと考えた方がよい。

幸運なことに、今回構成しているアプリはデータチェックアプリであるので、データベースに含まれるデータ全てを扱う性格のものではなく、データベースの入り口におけるフィルターに過ぎない。一度に扱うものはフィルターを通るデータだけであるため、近年のメモリの大容量化を考慮するとデータをデータベースに溜めることなくメモリ上で扱えるレベルである。

これにより、データベースソフトのインストールを必要とせず、データチェックアプリ自体もインストールの必要のない実行ファイルで構成できる。

現在の IR は一般に大学の事務職員がデータの準備を行っているが、事務用パソコンには勝手にアプリをインストールすることはできない場合がある。極端な話、実行ファイルの場合は USB メモリ上で実行することもできる。

6. アプリの使用結果

実際に本アプリを使って昨年度作成した IR データをチェックしてみた。使用したデータは北海道大学の総合理系の学生 340 人分の教学 IR データで、データのフィールド数は 162 である。よって、CSV データにおけるセル数は、55,080 あることになる。チェック時間は 2 秒以内に完了し、計 2368 セル (4.3%) のエラーを発見した。

内訳は、余分なスペース：1307 セル、項目間の整合性のエラー：934 セル、想定される規格から外れた値：53 セル、値が入るべきフィールドがブランク：55 セル、フィールド名の不備：12 セル、不良レコードの可能性の指摘：4 セル、レアケースにつき要確認：3 セルであった。

実際に、データクリーニングは行き届いていない現状がわかるであろう。

7. フォッグ式行動モデルによる検証

本取り組みは、コンソーシアム型 IR における IR 担当者全体を対象者と想定して行っている。しかし、現時点では自大学でのみの使用であるので、一般に使用されるものになるかどうかはわからない。そこで、ユーザーに好んで使用されるアプリとしての要素を兼ね備えているかを理論になぞって確認しておく。

田嶋 (2016) は、スマートフォンアプリを起動する消費者行動をフォッグ式行動モデル (Fogg) に当てはめて考えている。フォッグ式行動モデルによると、ユーザーが行動を起こす要因は3つあり、動機 (Motivation)、行動の容易さ (Ability)、トリガー (Triggers) としている。さらに、動機は「快楽／苦痛」「希望／恐怖」「社会的容認／否定」、行動の容易さは「時間」「お金」「肉体的負担」「精神的負担」「繰り返し」「社会的逸脱性」、トリガーは「スパーク」「ファシリテーター」「シグナル」によって構成されている。

本アプリが利用者に IR 担当者に適合したものであるかどうかをフォッグ式行動モデルをベースに検証することにする。

1) 動機

アプリ起動の動機

・苦痛／快楽：

手動によるデータのクリーニングの負担感／アプリで一括検索

・恐怖／希望：

データクリーニングの不備で受け付けられず残

業の恐怖／アプリで受付レベルを超えたチェック

・社会的容認／否定：

凡ミスを含んだデータ公表への羞恥心／アプリにより個人でチェック

2) 行動の容易さ

アプリ起動の障壁

・時間： 瞬時に結果を得られる

・お金： 無料

・肉体的負担： ユーザーに求められる操作なし

・精神的負担： 使い方を覚える必要なし

・繰り返し： エラーが一覧化され、継続的操作は不要

・社会的逸脱性： 学外と隔離して利用可能

3) トリガー

アプリ起動のためのトリガー

・スパーク： (動機を刺激する時)

作成したデータをその場で瞬時にチェックできる

・ファシリテーター： (動機は十分高まったが行動障壁がある時)

アプリ起動後、ファイルをドロップするだけで使用できる

・シグナル： (行動障壁もクリアしている時)

目につくデスクトップに実行ファイルを置くことができる

このようにフォッグ式行動モデルによって検証してみると、IR 担当者が利用するに十分な動機、行動の容易さ、トリガーを備えているものと考えられる。今後のコンソーシアム型 IR における利用拡大が期待できるであろう。

8. まとめ

本アプリを作成・使用してみて、コンピューターによる一括チェックの便利さ、書式依存で専用アプリとすることによる完全自動化の利便性は非常に高く、IR 担当者の支援として一定の成果を上げられた

と思う。実際に、本学の IR 担当者は本アプリの使用により業務がスムーズに進んだと答えている。

一方、開発側としてはアプリ開発の素人ならではの問題点を感じた。

まず、コンソーシアム IR のような IR 担当者の集団を対象としてアプリを作成する場合、一定の責任が発生せざるを得ないであろう。これは、業者ではなく個人、ましてや初級プログラマが作成する場合、マネジメント、ソフトウェアチェックが甘く、バグの発生は当然の結果である。アプリがある場合とない場合での業務量を考えて、一定の不備は受け入れる姿勢で利用しなければならないであろう。

バグチェックの難しさが予想される。もし、このアプリを他大学で使用する場合、データの性格上、バグが発生したデータをアプリ開発者に提供できないであろう。その場合、バグの修正が難しくなってしまう。多くのバグを含むであろう素人によるアプリ開発においては問題である。

最後に、初級プログラマが作成することに関して、やはり手間は大きいと言わざるを得ない。上級プログラマであれば非常に簡単なレベルのアプリであると思われるが、自分のキャリアの中でプログラミングが必要になったことが無い筆者の場合、“文字の色を変える”ような基本的なコーディングすらインターネットで調べながら行うことになる。作ってしまった後の保守としての微調整は趣味レベルで対応できるが、アプリの作成自体は IR の仕事として対応するレベルとならざるを得なかった。

参考文献

- 大学 IR コンソーシアム (2012), <http://www.irnw.jp/history.html>
- 田嶋勝利 (2016), 「アプリ企画, DAU に悩む人へ—アプリを起動してもらふ不変の条件」, SmaList ウェブサイト, <http://smaListblog.com/>, 2016 年 2 月 8 日
- 文部科学省 (2012), 「大学における教育内容等の改革状況について (平成 26 年度)」, http://www.mext.go.jp/a_menu/koutou/daigaku/04052801/_icsFiles/afieldfile/2016/12/13/1380019_1_1.pdf
- 柳浦猛 (2009), 「アメリカの Institutional Research IR とはなにか?」, 国立大学法人における授業料と基礎的教育研究経費に関する研究第 12 章, 国立大学財務・経営センター
- 柳浦猛 (2013), 「アメリカ大学事情 Vol.3 アメリカから見た日本のインスティテューショナル・リサーチ その 1 —「学生串刺しデータファイル」に思うこと, 及び SQL のすすめ。」, http://www.postsecondaryanalytics.com/jp_blog003/, 2013 年 4 月 22 日
- Fogg, B. J., BJ Fogg's Behavior Model, <http://www.behaviormodel.org/> (2016 年 8 月 1 日観覧)
- Hagedorn & Coogan (2007), 柳浦猛 (2009) を参照
- Terenzini, P. T. (1999), “On the Nature of Institutional Research and the Knowledge and Skills it Require,” *New Directions for Institutional Research* 104, 21-29