平成 25 年度卒業論文

論文題目

電子書籍の見開きと検索機能の改善

神奈川大学　工学部 電子情報フロンティア学科
学籍番号　201002713
鴫原　善寿

指導担当者　木下宏揚　教授

目 次

第1章　序論

1.1　背景

スマートフォン等の急速な普及は、世界の移動体通信事業及びその関連産業の視界を一変させつつある。Apple 社は、iPhone、iPad の世界規模での販売拡大で一気に株式時価総額 1 位になり、Android OS を採用する端末を生産・販売する中国・韓国・台湾系企業が躍進している。また世界的に携帯電話市場の成長及び活性化を促しているといえる。なかでもスマートフォンの普及とともに現れた電子書籍にはさまざまな会社が力をいれた。電子書籍の市場には、普通の書籍大手であるアマゾン社だけではなく、最近は携帯式の音楽プレーヤーの iPod や携帯電話 iPhone で知られている Apple 社も参入しており、前者が kindle、後者が iPad という独自の専用機器を販売している。これはパソコンでもなく、携帯電話でもない、独自のコンセプトを持った電子機器であるといえる。電子書籍は

1.2　電子書籍のメリット・デメリット

電子書籍には従来使い続けてきた紙を媒体とした本とは大いにことなり、メリットとして大きく分けて以下の点がある。資源や電子書籍自体の機能の利点としては、印刷する必要がないため、 制作費（印刷・流通・製本・保管費など）が抑えられ、絶版がなくなる。資源が必要なくなる。文字のサイズなども変えることができ、フルカラーや文字の色を変えることもできる。また電子書籍なら音声や映像を簡単に連動させることができる。この機能を使えば、文中の知らない語彙や人物を調べることが出来たり、他のアプリケーションなどとの連動も期待できる。新しい情報発信法としても期待でき、情報の発信が容易になり国境を越え、作品を世界中に発信が可能となる。情報の受け方の変化として世界中どこにいても、いつでも、書籍の購入が容易となる。本棚がいらず、携帯性もあがるというメリットがあげらる。デメリットととしては、本の読みやすさなど、慣れや読み手側の感覚の部分にある。そのため、デメリットと感じる部分には個人差がある。共通となるのは電子デバイスについては、電子書籍のフォーマットが不統一であるため、専用端末でしか見ることができなかったり、電子デバイスをはじめとした再生機器の代金は初期費用のみであるが、かかってしまう。また、購入時のインターネット接続費用もかかる。よってデメリットは大きくわけて人の本にたいする慣れと、費用の点にあるといえる。電子書籍には専用端末側にバックライトや電子粉流体

が使われているものがあり、目に疲れがたまりやすいという身体的なデメリットに対しての技術が仕様されている。

1.3 電子書籍の改善点

デメリットとは言えないが紙からなる本との違いとして、従来の電子書籍では本のようにぱらぱらめくったりすることができない。また、検索機能をつかったところ、二語以上の検索ができなかった。ぱらぱらめくれるものもあるが、それはインターフェースの見かけ上の機能になってしまっている。二語以上の検索も一語の検索を二度行えばよい。よって、電子書籍の構造を理解し、書籍内の文字や画像などのデータを圧縮して格納することで処理能力をあげ、普通の書籍のようにぱらぱら読めるようにし、検索した一語から最も関連のある一語を抽出し、語含む一文のレイアウトを変えることでぱらぱら読んだときに目にとまるような電子書籍の構造を提案することを研究目的とする。格納の手段としてＢｕｒｒｏｗｓ　Ｗｈｅｅｌｅｒ　変換を扱う。検索方法の手段としてベクトル空間モデルを用いる。

第2章　基礎知識

2.1　電子書籍とは

電子書籍とは, 書籍や出版物の情報をデジタル化し, 印刷物の替わりに電子機器のディスプレイ上で閲覧可能なコンテンツの総称である。。電子書籍を閲覧するために用いられるハードウェアやソフトウェアは、電子書籍リーダーと呼ばれる。電子書籍リーダーの専用端末の代表的商品としては、「Kindle」や「sony Reader」などがあげられる。電子書籍は、EPUB や XMDF といった規格が複数あるため、電子ブックリーダーで閲覧するためには、電子書籍の形式をサポートしている必要がある。

2.1.1　電子ペーパー

紙の長所とされる視認性や携帯性を保った表示媒体のうち、表示内容を電気的に書き換えることができるもののことである。1970 年代に米国ゼロックス社のパロアルト研究所に所属していたニック・シェリドンが Gyricon と呼ばれる最初の電子ペーパーを開発した。Gyricon の構造は、半球を白、別の半球を黒に塗り分けた微小な球をディスプレイに多数埋め込んだものである。球の一部は静電気を帯びており、電界によって球を回転させることで白地に黒い文字を浮かび上がらせることができ、数千回の書き換えにも耐えた。低消費電力表示中に電力を消費しないか、又は極小で済み、書き換え時の消費電力も非常に少ない。応答速度電気泳動方式では非常に遅く動画用途には向かなかったが、電子粉流体では液晶よりも高速になっている。高い視認性紙と同じように反射光を利用して表示を行うため、視野角が広く直射日光に当たっても見易く、目に対する負担が少ない。暗所では別に照明が必要になる。薄くフレキシブル紙のように薄く作ることができる。

2.1.2　フォーマットについて

EPUB とは、電子出版業界の標準化団体である IDPF (International Digital Publishing Forum) によって標準化されたリフロー (再流し込み) が可能な電子書籍および出版物のための XML フォーマットのことである。IDPF が 2007 年 10 月に正式に EPUB を承認した後、2008 年までに主要な出版社で EPUB が採用されるに至る。EPUB フォーマットは多種多様なオープンソースのソフトウェアや市販

のソフトウェアを使って読み取ることができるだけでなく、主要なすべてのオペレーティング・システム、Sony や PRS などの電子書籍リーダー、そして Apple 社の iPhone などの小型機器でも使用することができるフォーマットである。また、EPUB のフォーマットには EPUB の使用を電子書籍に制限するような要素は何もない。無料で入手できるソフトウェア・ツールを使って、Web ページを EPUB としてバンドルしたり、プレーン・テキスト・ファイルを変換したり、既存の DocBook XML 文書を整形式の妥当な EPUB に変換したりすることも可能である。

2.1.3 EPUB と PDF との違い、およびお互いの利点

PDF は世界中で最もよく使用されている電子文書フォーマットである。書籍の出版者の観点からは、PDF には以下の利点がある。PDF ファイルでは、ピクセル単位で完全にレイアウトを制御することができ、段組みや、偶数ページと奇数ページでスタイルを変えるといった複雑な出力に対応したレイアウトにすることも可能。PDF を生成するには、Microsoft Office Word をはじめ、多種多様な GUI ベースの文書ツールを使用することが可能。PDF リーダーは至るところに普及していて、最近のほとんどのコンピューターにインストールされている。PDF では特定のフォントを組み込むことで、最終出力を正確に制御することが可能。 EPUB は以下の 3 つの IDPF 仕様からなるが、実際この 3 つをまとめて EPUB と総称して差し支えない。OCF (Open eBook Publication Structure Container Format): EPUB アーカイブのディレクトリー・ツリー構造およびファイル・フォーマット (ZIP) を指定している。OPS (Open Publication Structure): 書籍のコンテンツに使用可能なフォーマット (XHTML および CSS など) を中心に、電子書籍の共通語彙を定義している。OPF (Open Packaging Format): EPUB の必須メタデータとオプション・メタデータ、読み取り順、および目次について記述している。上記の他、EPUB は EPUB アーカイブ内に含まれる特定のタイプのコンテンツに対して、XHTML バージョン 1.0 や DAISY(Digital Accessible Information SYstem) などの標準も再利用している。 . 一方、ソフトウェア開発者の観点から見ると、PDF は以下の理由から、最適と言うには程遠いフォーマットである。簡単に習得できる標準ではないため、独自の PDF 生成コードを作成するのは容易でない。PDF は ISO (International Organization for Standardization) 標準となっているが (ISO 32000-1:2008)、従来から Adobe Systems 一社によって管理されている。PDF ライブラリーはほとんどのプログラミング言語で用意されているが、その多くは商用であり、GUI アプリケーションに組み込まれてる。そのため、外部プロセスによって制御するのは簡単ではなく、また、無料のライブラリーのすべてがアクティブに保守を継続していない。PDF 固有のテキストは、プログラムによって抽出および検索することができるが、Web で使いやすいフォーマットに簡単に、あるいは確実に変換できるようにタグが付けられている PDF はほとんどない。PDF の

リフロー (再流し込み) は容易ではない、つまり、小さな画面や大幅なレイアウト変更への対応が十分にはできない。

2.1.4 EPUB が開発者にとって扱いやすい理由

EPUB は、開発者にとって PDF が使いにくいとされるすべての点に対処できる。EPUB は、規定の順序でファイルが含まれる、単純な ZIP フォーマットのファイル (拡張子 .epub) である。この ZIP アーカイブを作成する場合、その方法に関していくつか注意が必要な要件があるが、それを抜かせば EPUB は以下のように簡単に扱えるようになっている。EPUB では、ほぼすべてのファイルが XML で構成されている。特殊なソフトウェアや専用のソフトウェアがなくても、標準的な XML ツールキットを使って EPUB ファイルを作成することができる。EPUB のコンテンツは、ほぼ XHTML バージョン 1.1 である。代わりに使用できるフォーマットは、視覚障害者用に書籍をエンコードするための標準である DTBook である。EPUBXML スキーマの大部分は、自由に利用できる公開済みの仕様から引用されている。　以上より重要な点は 2 つで、EPUB のメタデータは XML であること、そして EPUB のコンテンツは XHTML であることの 2 点である。文書の作成システムが Web 用の出力を生成するか、XML をベースとしているとしたら、そのシステムは EPUB も生成できる仕様である。

2.1.5 電子ブックリーダー

電子ブックリーダー（でんしブックリーダー）とは電子書籍を閲覧するための専用端末（デバイス）、および電子書籍データを表示する専用ソフトウェアである。電子書籍ビューワー、電子書籍専用端末、デジタルブックリーダー、E ブックリーダーとも呼ばれる。電子書籍出版社は、各スマートフォン向け（iOS、Ａｎｄｒｏｉｄ）に電子書籍ビューワーアプリを提供し、無料でほぼすべての電子書籍を読む事が可能となっている。読書感は、専用端末には負けるものの、新たに端末を買う必要がないというのは大きなメリットである。また、スマートフォンだとさすがに画面が小さく読みづらい場合、汎用タブレット端末も同様に電子書籍ビューワーアプリが出ているため、そちらを利用すれば大きな画面で読む事も可能であるが、電子ブックリーダーを使うと以下の特徴があるため、読みやすさに違いがある。・表示部に eInk などの省電力な電子ペーパーが使われはじめている。・大容量で低価格となったフラッシュメモリの採用で、多数の電子書籍を格納できる。・バッテリーの性能の向上と電子回路の省電力化技術によって、長時間使用が実現された。特に電子書籍専用端末に向いた最新技術には新たな種類の電子ペーパーがあり、これまで以上に省電力で高コントラストの表示が実現される。

2.1.6 レイアウト

電子書籍のレイアウトには2つの種類があり、リフロー型と固定(フィックス)レイアウト型の二つである。どちらが適切かは、どんな本を作りたいかで変わる。

リフロー型

電子書籍には自由に文字の大きさやフォントを変更できる機能があるが、これが出来る本は全てリフロー型の本である。小説やエッセイ、参考書などの、読み物系は主にこのリフロー型である場合がほとんどである。文字の大きさの変更などに応じて、1ページに収まる文章の量が増えたり減ったりと、レイアウトが状況に応じて変化するというのがこのリフロー型の特徴である。また、そのカスタマイズ性に優れている反面、思い通りに見せる事は不可能になる。文章だけの本であれば、あまり問題は生じないが、ある文章の直後に表示される挿絵がある場合、それを同じページ内に表示したいという事があったとした時、読者がフォントや文字の大きさをカスタマイズする事で、その挿絵が出てくるページのレイアウトがズレて、挿絵は次のページになってしまう事が十分起こりえるので、そのあたりを完全にコントロールしてこだわれないのがリフロー型の弱点である。

固定レイアウト型

完全に一つのページの中に表示する画像や文字の位置を固定してしまう方式である。これは主にマンガや絵本、画集や写真集だったり、画像を中心に扱う書籍に使われる事が多いレイアウト方式である。絵を中心に本の流れが作られるタイプの本だと、勝手にレイアウトが変わられては作者の意図通りに読者が読む事が難しくなる為、それを避ける為にはレイアウトを固定するしか方法はないため、固定レイアウト型は作者がイメージした通りの形を保つ事が出来るが、読者にとってはカスタマイズの自由性がなくなるという捉え方になってしまう。

2.1.7 空間ベクトルモデル

検索結果として得られた文書集合は、質問に対し一様な度合で得られるわけではなく、必ずしも完璧な結果が得られるわけではない。ここで最も完成度が高い検索方法として最優良検索を用いる。最優良先検索は、質問に対し得られた文書をどれくらいの割合で適合しているかという度合で優先度を付けていく方式である。最優良先検索にはブーリアンモデル、確率分布モデルなどがあるが、今回の研究では空間ベクトルモデルを用いる。空間ベクトルモデルとは、情報検索を行うためのアルゴリズムのひとつである。一語を一つの次元に対応させるベクトル空間を用いたとき、一語一語はベクトル空間内の点として扱う。多次元のベクト

ル空間上に配置し、検索対象のベクトル表現と検索語のベクトル表現の相関量を互いのベクトルのコサイン、内積、距離等で計算し値を付ける。この数値化のよって関連度を求める方式である。

2.1.8 クエリー (query)

データベース管理システムに対し、処理の要求を文字列で表したものを指す。データの更新や検索、削除等の命令をシステム上で発行する際に使われる。検索機能を使う場合、検索クエリといい、検索を行う際に実際にユーザーが入力する単語、複数語のことを指す。

2.1.9 共起頻度

五語処理の分野において任意の文書や文などの限定した範囲において、ある文字列とある文字列が、同時に出現することをいう。また文字列の共起頻度を単語単位のみではなく、任意の n-gram 単位で集計することができる。この集計されたものは必ずしも、単語同士の共起関係とは言えない。

2.1.10 n-gram

任意の文字列、文書などにおいて、任意の n 文字列が連続した文字列のことである。1 文字の場合は unigram、2 文字の場合は bigram、3 文字の場合は trigram、4 文字以上の場合からは 4-gram、5-gram と表現されることが多い。

2.1.11 再現率 (recall)

再現率とは情報工学の分野においてシステムが検索結果や判定結果などとして出力した結果が、あらかじめ用意した正解となるデータと比較してどのくらい網羅しているかを表す指標である。情報検索の結果において、R を検索された適合文書の数、C を全文書の中の正解文書の数とすると、再現率は R/C で得られる。

2.1.12 適合率 (precision)

システムが出した結果において、検索対象の文書群の中から、正しく検索されたかどうかの文書の割合を出す。出力したすべての結果の中に、どれだけ正解が含まれているかの割合を指す。再現率同様に R を検索された適合文書の数、N を検索結果の文書の数とすると適合率は R/N で得られる。

2.1.13 F 値 (F-measure)

適合率をあげると再現率が下がる。また、再現率を上げれば適合率が下がる傾向がある。そのため、F 値という尺度をよく用いる。F 値は適合率と再現率の調和平均である。再現率と適合率で扱った R、N、C の相加平均で割ったものに相当する。F 値が高いことが性能が高いことを指す。

第3章　提案手法

3.1　提案手法

3.1.1　tf-idf法

空間ベクトルモデル tf-idf 法を用いる。ある単語の文書内での出現頻度を TF（Term Frequency）といい、ある単語が含まれている文書の頻度を df(document frequency) という。またこの d f の逆数を idf(inverse document frequency) という。単語の出現頻度が高いことと、その出現頻度が高い単語が含まれる文書頻度が少ないという二つの条件を満たすことで単語の重み（重要性、優先度）を特徴付ける。単語の出現頻度を指す tf と単語を含む文書頻度の逆数を指す idf を掛け合わせたものとなる。

tf ×　idf　　tf 単語の出現頻度 idf df の逆数

大域的重み付け

idf(inverse document frequency）は df の逆数となり。この場合は対数を使う。対数をとることによって、ほとんどの文書で出現する単語の影響を小さくするためである。

idf = log(N/df) + 1 idf df の逆数 N　　全文書数 df　　単語が含まれている文書頻度

ここで+1 をしているのは idf の最小値が 0 なので tf-idf 法を用いたとき、tf の値によらず 0 になってしまうために+1 をしている。

文書正規化係数ある単語を含む文書が長い場合、しまうため、結果長い文書に含まれる単語の方が重みが強くなってしまう。よって調整が必要であるためコサイン正規化を行う。

3.2　文字列データ圧縮

文字列に対するデータ圧縮法で、近年、大規模文字列データを扱ううえで、データ圧縮は特に重要な技術として注目されている。データ圧縮とは、データの内容や性質を保ったまま別の表現方法で表すことで、元の表現より少ないデータ量で表現することである。圧縮の実用的な側面として、データをいかに小さく表現できるかだけでなく、いかに効率よく圧縮、復元できるか、またデータをすべて見てから圧縮するのではなく、前から順に圧縮できるかといったさまざまな目標

があるが、ここでは電子書籍のインターフェース上の見開きの速さではなく、データ圧縮によりデータの構造を簡潔にし、インターフェースを見かけ上の速さではなく、アルゴリズムを組み立てた見開きの速さを目的とする。

3.3 圧縮と牽引の融合

簡潔データ構造は大規模な文字列な文字列データ解析において重要な役割を果たす。簡潔データ構造は、操作が高速でありながら作業領域量が小さいという二つの性質を兼ね備えたデータ構造であり、データ圧縮と索引の技術が組み合わさったものである。これまでの計算機の演算処理性能はムーアの法則にしたがって向上し続ける一方、記憶装置のアクセス速度の向上速度はそれに追いつかず、演算速度と記憶装置のアクセス速度の差は広がっていった。こうしたトレンドから、近年のデータ処理においては計算回数をへらすよりも、いかに上位の高速な記憶階層で計算できるかが、重要となっている。

もう一つデータ処理の方法の観点からみた場合、多くの解析の場面では同じデータに対して繰り返し、さまざまな解析を行うという特徴がある。情報検索はその代表例であり、同じデータに対し、さまざまな検索操作を繰り返し適用する。この場合、データをあらかじめ分析しておき、索引を利用して解析に必要なデータのみを参照できるようになる。最初の一回の索引構築は必要だが、索引さえできれば、毎回の解析操作を高速化できる。

3.4 ＢＷＴ変換

Burrows Wheeler 変換は Burrows,Wheeler によって考案された文字列の可逆変換である。ＢＷＴの定義を示す。定義 (Burrows Wheeler 変換)　文字列 T[0,n) とその接尾辞配列 SA[0,n) が与えられたとき、ＢＷＴは次のように定義される。

$$T[i] = T[SA[i] - 1]SA[i] > 0$$

$$T[n-1]SA[i] = 0$$

3.5 ＢＷＴ変換を用いた圧縮

アルファベット集合Σ＝ {0,1,2,,, σ-1} からなる長さ n の文字列 T とし、ＢＷＴ後の文字列をＴｂとする。このＴｂに対し、次に三つの変換を適当する。・先頭移動方によりＴｍ=mtf(Tb) に変換

・連長圧縮によりＴｒ=rle(Tm) に変換

・Tr をエントロピー符号化

3.5.1 先頭移動法

先頭移動法により、任意の文字が連続しやすい文字列から、小さい文字列が出現しやすい文字列に変換される。

3.5.2 連長圧縮

連長圧縮では、連続する文字を文字とその繰り返し回数のペアで記録する。この繰り返しのことを連長と呼ぶ。この連長の符号化には正整数の符号を使うことができる。連長が小さいものが多いものであればγ符号やδ符号などそれに合わせた符号を使う。連長圧縮は単純でありながら有効であり、多くのデータ圧縮の前処理、または後処理で利用されることが多い。ここでは任意の位置の文字に圧縮したままアクセス可能な連長圧縮を考える。これは連長の符号化に完備辞書を組み合わせる。文字列T中で先頭、もしくは前の文字と異なる (—T[i] ≠ T[i-1]) 場合はB[i]=1、そうではなく前の同じ文字である場合はB[i]=0であるようなビット列Bを考える。そしてBに対する完備辞書を構築する。また各繰り返しの専用文字をUに格納する。この際T[i]はU[rank(B,i+1)-1]として求められる。このような完備辞書を利用した連長圧縮のサイズは疎なビット列に対する完備書籍を利用した場合、Tの長さがn、アルファベット種類数をσ、連長の数をγとしたとき、rlg σ+rlg(n/r)+2rビットであり、定数時間でアクセスできる。

3.5.3 エントロピー符号化

全体の文字の頻度表を用いて、ハフマン符号、もしくは算術符号などをもちいて符号化する。出現率の高い値に短いbitを、出現率の低い値に長いbitを割り当てる符号化方式である。 文字の符号化では、すべての文字に対し8bitや16bitの固定長の符号を均等に割り当てているが、実際の文章では、頻繁に使われる文字もあれば、めったに使われない文字もあるため、出現頻度に応じて異なる長さの符号を割り当てれば、データ全体をコンパクトに符号化できる。こうした方法をエントロピー符号化と言う。

第4章　提案

4.1　提案

何度やっても式と図が出てこないのでもう一度乗せなおします。

参考文献

[1] 電子ペーパー　－ねらいと開発の現状－
面谷　信
東海大学　工学部　未来科学技術共同研究センター

[2] web 上の情報からの人間関係ネットワークの抽出
人口知能学会論文誌　2005

[3] XML データベースの自然言語検索技術
真鍋　俊彦　國分　智晴
東芝レビュー　Vol.64 No.2 (2009)

[4] 検索キーワードを含む最小 XML 部分文書抽出のための索引手法
DEWs2007C1-4
三木　健士　横田　治夫
東京工業大学　大学院　情報理工学研究科　計算工学専攻
東京工業大学　学術国際情報センター

[5] 語の共起の統計情報に基づく文書からのキーワード抽出アルゴリズム
Vol.43 No.SIG 2(TOD 13) 情報処理学会文誌：データベース Mar.2002
絹谷　弘子　波多野　賢治　吉川　正俊　植村　俊亮

[6] 印象語に基づく Web ページデザインシステムの構築とその評価
社会法人　電子情報通信学会　信学技報
TL2007-44 (2007-12)
黒田　英憲　小澤　朋之　亀田　弘之
東京効果大学　コンピュータサイエンス学部
東京工科大学大学院　バイオ・情報メディア研究科

[7] 特定分野を対象とした連想検索のための書籍の索引部を用いたメタデータ空間生成方式
電子情報通信学会論文誌 2005/4 Vol.J88-D-I No.4
中西　崇文　岸本　貞弥　櫻井　鉄也　北川　高嗣

[8] 書籍の目次と索引を利用した専門用語ネットワークの構築
社会法人　情報処理学会　研究報告　2006-FI-84

IPSJ SIG Technical Report 2006-NL-175
2006/9/12
石塚　隆男
亜細亜大学　経営学部

[9] 小説テキストを対象としたジャンル推定と人物抽出
馬場こづえ　藤井　敦　石川　徹也
筑波大学図書館情報専門群
筑波大学大学院図書情報メディア研究科

[10] 異分野データベース群を対象とした意味的検索空間統合方式とその実現
Vol.43 No.SIG 5(TOD 14) 情報処理学会論文誌：データベース　June 2002
石原　冴子　清木　康

[11] 目次情報などを利用した図書・文献検索方式
投稿論文
長尾　真