データサイエンスへの適用を想定した情報理論教育

【論　　文】

データサイエンスへの適用を想定した情報理論教育

伊　藤　則　之

1. はじめに

情報理論は，1948 年のシャノンの論文 “A Mathematical Theory of Communication”［1］が起源となっている。この論文のなかで説明される最初の図では，情報源で発生した情報は，通信路を通じて伝送され，受信側でその情報を受け取るという通信システムがモデル化されている。情報は符号化されて通信路を伝わり，受信側に到達したら復号化されて情報として受け取られる。受け取った情報により，受信側での不確からしさが減少すると解釈される。また，通信路の途中には，ノイズ源が定義され，伝送中にノイズが入り込む可能性を想定している。こうした情報理論では，個々の情報が持つ情報量，情報源から発生する平均情報量としてのエントロピー，情報を通信路に送るための符号化など，情報に関する基本的な理論を学ぶことになる。

情報を扱う情報科学やデータサイエンスを学ぶ学生がこの情報理論を学び，学んだ情報理論をその後に学ぶものに有効なかたちで生かせるように教育することが必要である。特に，データサイエンスの世界におけるデータ，このデータと情報理論が扱う情報の関連性を十分に意識することが重要と考える。筆者は情報理論を学び，その後に社会に出て情報理論の知識を実際の業務で生かすことができた。情報理論を学ぶ際，なぜ情報理論を学ぶのか，学んだ情報理論をその後に学ぶものにどのように生すことができるのか，情報理論を学ぶ学生にはこうした点についても具体例を示しながら教育することが必要であると感じてきた。そこで，本論文において，データサイエンスへの適用を想定した情報理論の教育について，具体的な例を示しながら教育方法を提案したい。

2. 情報量と相互情報量

情報理論では，情報の発生の源である情報源が出発点となる。情報源としては，文字，音声，画像など様々あるが，一番扱いやすい記号の系列による情報源から出発するとわかりや

すい。特に，究極に単純化したものとして，0または1からなる記号列から情報理論を説明することが多い。2進数1桁，つまり0か1かの1桁の情報量は1ビットと定義される。1ビットは，高等学校の情報の教科書でも2進数1桁に相当すると説明されている。しかしながら，情報理論では，2進数1桁の情報の情報量が必ずしも1ビットではなく，0または1が発生する確率と関連付けられて説明されるという点が重要である。

0と1の発生確率が等しい場合，この2進数1桁の情報量が1ビットとなる。たとえば，0と1の2つの目のみのサイコロがあって，そのサイコロを振って，どちらの目が出たかを教えてもらうとき，1ビットの情報を受け取ったことになる。しかし，10,000回振って1回の確率で1の目が出るサイコロを考える。このサイコロを振って，0の目が出たことを教えてもらったとしても，もともと0の目が出る可能性が非常に高いので，受け取った情報量は0に近いものとなる。逆に，1の目が出たことを教えてもらった場合，1の目が出る可能性が1/10,000と非常に小さい。そのため，発生する確率が小さいことが発生したという情報を受け取ったとき，その情報量は大きい値となる。確率xを持つ事象Xが発生したとき，これが与える情報量yは$-\log_2(x)$と定義される。この$y=-\log_2(x)$という式をWindowsの電卓のグラフ計算機能を使ってグラフにすると図1のようになる。確率xは0から1までの範囲の値をとるため，図1を見るとxが1，つまり確率が1の場合に情報量が0となり，確率が0に近づくほど情報量が大きくなることがわかる。情報量の説明では，直観的に理解してもらう例を示すと同時に，式やグラフで定量的に理解してもらうことが重要と考える。発生確率が小さい貴重な情報の情報量は多く，発生確率の大きな当たり前の情報の情報量が少

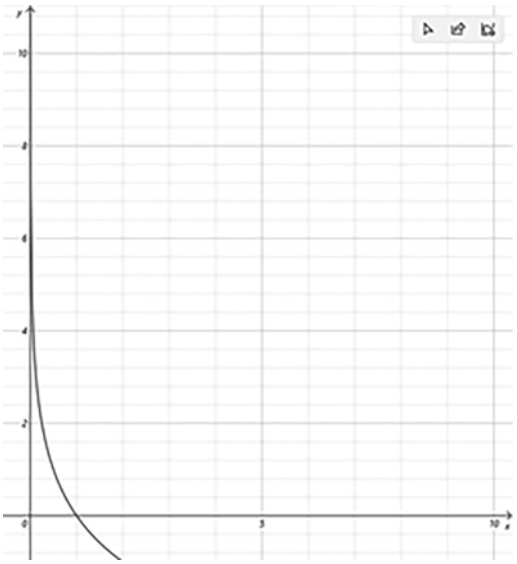

図1　$y=-\log_2(x)$のグラフ

ないというように，情報源で発生する事象の確率とその事象の持つ情報量の関係を理解してもらうことが重要と考える。なお，確率 x を持つ事象 X が発生したとき，これが与える情報量 $y=-\log_2(x)$ は自己情報量と呼ばれることは，のちに出てくる相互情報量と対比するためという説明も必要となる。

情報理論では，情報源で発生した情報は送信側から記号の列 $a_1,a_2,a_3,...,a_n$ として通信路に送り出され，受信側で記号の列 $b_1,b_2,b_3,...,b_n$ として情報を受け取るというモデルを前提としている。そして，a_i の発生する確率を $P(a_i)$ とし，b_j の発生する確率を $P(b_j)$ とする。このとき，通信路にはノイズが入り込むことを前提としているため，送信側で a_i を送り出し，受信側で b_j を受け取ると考え，このときの確率を $P(a_i,b_j)$ とする。また，送信側で a_i を送り出したという条件の下で，受信側で b_j を受け取る条件付き確率を $P(b_j|a_i)$ とする。逆に，受信側で b_j を受け取ったという条件の下で，送信側で a_i を送っている条件付き確率は $P(a_i|b_j)$ となる。つまり，受信側で bj を受け取ったという条件の下で，送信側で ai を送信したことが知らされたとき，$-\log2(P(ai|bj))$ だけの情報量を受け取る。この情報量を I(ai|bj) と表記する。また，受信側で何を受け取ったかに関係なく，送信側で ai を送信したことが知らされると $-\log2(P(ai))$ だけの情報量が得られる。この情報量を I(ai) と表記する。このとき，$I(a_i)-I(a_i|b_j)$ が a_i と b_j の相互情報量 $I(a_i;b_j)$ と呼ばれるものになり，$I(a_i;b_j)=\log_2(P(a_i,b_j)/(P(a_i)P(b_j)))$ となる。この相互情報量は，ai と bj が全く独立な事象の場合では 0 になり，その逆の場合は I(ai) となる。つまり，相互情報量は，a_i と b_j との依存関係の大きさを示す指標となる。

情報理論では，シャノンが提唱した通信システムのモデルにおいて，送信側と受信側という 2 つの場所での事象について相互情報量を定義している。情報理論を学生に学んでもらう際には，a_i と b_j との依存関係の大きさを示す指標である相互情報量は，送信側と受信側という 2 つの場所での事象についてだけでなく，様々な応用範囲があることも合わせて学んでもらうことが重要である。たとえば，テキストマイニングにおいて，まずは単語の出現頻度の解析が重要になるが，そのあとで単語の意味的な類似性や係り受けの関連性をこの相互情報量を用いて分析することが可能であることを学生に示すことにより，その後の学びのなかで相互情報量の応用能力も身につけることができる。文中で 2 つの単語 a, b がそれぞれ出現する確率，また，単語 a の前後 n 単語以内に単語 b が出現する確率などを分析して相互情報量を算出することにより，2 つの単語の意味的な類似性や係り受けの関連性を調べることができる。さらに，2 つの単語の間の相互情報量を用いることにより，相互に関連する単語類をまとめるグルーピングなどにも利用することも示すことにより，学んだ学生は相互情報量の考え方を適用する範囲を広げることになる。

3.　エントロピー

熱力学のなかでエントロピーという考え方があり，原子や分子の乱雑さを表す指標とされる。エントロピー増大の法則，つまり放っておくと乱雑さは増大するという法則であり，筆者がこれを学んだとき，自分の部屋の散らかりようが日々ひどくなることを思い浮かべてこの法則に納得したものである。このエントロピーの考え方は情報理論のなかにも存在する。コインを投げて表裏のどちらが出るか，この事象から得られる情報量の平均値をエントロピーと呼んでいる。このエントロピーは，その事象が起こった結果によって取り除かれる不確からしさということもできるし，未来予測の難しさの程度や，また受け取ったときに得られる情報量の期待値と考えることもできる。

コインを投げて表と裏の 2 種類の記号のどちらかが出る事象について，1 回の試行で得られる情報の平均値を考える。表の出る確率を p とすると，裏の出る確率は 1－p となる。1 回の試行で表が出た場合に得られる情報量は $-\log_2(p)$ となり，裏が出た場合に得られる情報量は $-\log_2(1-p)$ となる。コインを 1 回投げたときに得られる情報量の平均値，つまりエントロピー $H(p)$ は，$-p\times\log_2(p)-(1-p)\times\log_2(1-p)$ となる。この p を x にした関数をグラフ化すると，図 2 のようになる。$p=0.5$ のときに，$H(p)$ が最大になる。これは，表と裏の 2 つの事象の発生確率が 0.5 で同じときに，エントロピーが最大の 1 となることを示

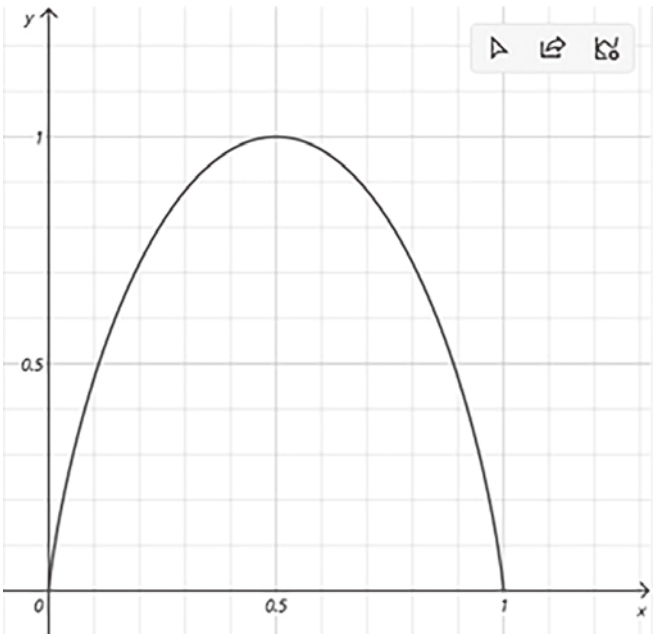

図 2　$y=-p\times\log_2(p)-(1-p)\times\log_2(1-p)$ のグラフ

している。p が 0 に近づくか，または 1 に近づくと H（p）は最小値の 0 に近づいてゆく。

宝くじを 10 枚購入して，1 等がどの番号になったかという抽選結果を情報として受け取ったとき，1 等が当たる確率は非常に小さいので，その情報が持つ平均情報量，つまりエントロピーは 0 に近い。一方，下一桁の番号が一致すれば当選する 7 等の場合，下一桁がどの番号になったかという抽選結果を情報として受け取ったとき，当たる確率は 1/10 と 1 等より大きいので，その情報が持つ平均情報量，つまりエントロピーは 1 等の場合の当選結果の情報より大きいということは，直観的にあっているといえる。

このエントロピーについて情報理論として学生に教えたあと，どのような場面で実際適用可能なのかについて，教える必要がある。身近でわかりやすい文章のエントロピーの計算方法を具体例として示すことが効果的であると考える。たとえば，つぎの 3 つの文章を例として示す。

文章 A：今日の天気は雨だった。
文章 B：朝から雪となり，雪の影響で朝から交通が渋滞となった。
文章 C：今日は，今日であり，今日は明日ではない。

文章を解析するときは，形態素解析をおこなって，文章を品詞毎に区切る方法をまず学生に教える。形態素解析の無料サイトなどで実際に試してみて，どのように結果が出てくるのかを学生に実際に確認してもらうことも効果的である。文章 A から文章 C の形態素解析の結果，各品詞の間をカンマ（,）で区切った結果が下記のようになることを示す。この形態素解析の結果から，名詞のみを選び出し，それぞれの文章のそれぞれの名詞について，出現回数，出現する確率（p），情報量（$-\log_2(p)$），確率×情報量（$-p\times\log_2(p)$）の項目について，Excel やプログラムで値を表 1 から表 3 のように計算することができることを学生に教え，学生に実際に計算してもらうというかたちで教えると効果的であると思われる。

文章 A：今日 , の , 天気 , は , 雨 , だっ , た ,。
文章 B：朝 , から , 雪 , と , なり ,, , 雪 , の , 影響 , で , 朝 , から , 交通 , が , 渋滞 , と , なっ , た ,。
文章 C：今日 , は ,, , 今日 , で , あり ,, , 明日 , で , は , ない ,。

また，表 1 から表 3 より，出現する確率（p）が小さい単語ほど情報量が多いが，確率×情報量（$-p\times\log_2(p)$）は，この例では確率（p）が小さい単語ほど値が小さいことを学生

表1　文章Aのエントロピーの計算

No.	形態素	出現回数	確率（p）	情報量（$-\log_2(p)$）	確率×情報量（$-p \times \log_2(p)$）
1	今日	1	1/3	1.58	0.53
2	天気	1	1/3	1.58	0.53
3	雨	1	1/3	1.58	0.53
合計	−	3	1	4.74	1.59 ←エントロピー

表2　文章Bのエントロピーの計算

No.	形態素	出現回数	確率（p）	情報量（$-\log_2(p)$）	確率×情報量（$-p \times \log_2(p)$）
1	朝	2	2/7	1.81	0.52
2	雪	2	2/7	1.81	0.52
3	影響	1	1/7	2.81	0.40
4	交通	1	1/7	2.81	0.40
5	渋滞	1	1/7	2.81	0.40
合計	−	7	1	12.05	2.24 ←エントロピー

表3　文章Cのエントロピーの計算

No.	形態素	出現回数	確率（p）	情報量（$-\log_2(p)$）	確率×情報量（$-p \times \log_2(p)$）
1	今日	2	2/3	0.78	0.39
2	明日	1	1/3	1.58	0.53
合計	−	3	1	2.36	0.92 ←エントロピー

には確認してもらう。情報量については，文章B> 文章A. 文章Cとなっていることを学生に確認してもらう。

4.　マルコフ情報源とオートマトン

情報理論で考える情報源では，情報源から記号が順次発生されるが，この記号の発生を確率事象であると考える。そして，記号の発生は毎回独立の事象として発生される場合と，その発生の前に発生された記号に影響を受けて発生される場合がある。前者は独立生起情報源と呼ばれ，後者はマルコフ情報源と呼ばれる。また，次の記号の発生事象の確率が現在の状態によって決まる場合，この記号の発生の過程はマルコフ過程と呼ばれる。このマルコフ過程は，状態遷移図により表現することが可能である。図3は，マルコフ過程を表現する状態遷移図の1つの例である。この例では，状態S1にいる場合，0.3の確率で記号0を発生し

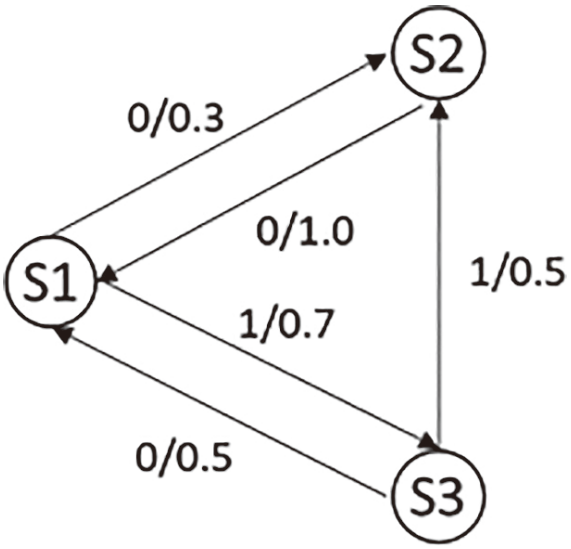

図3　マルコフ過程を表現する状態遷移図の例

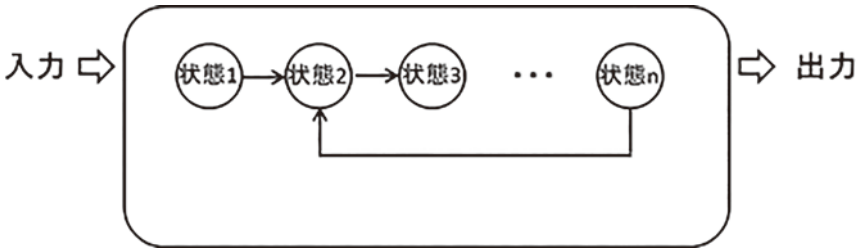

図4　オートマトンのモデル

て状態 S2 に遷移するか，または 0.7 の確率で記号 1 を発生して状態 S3 に遷移する。状態 S2 にいる場合，1.0 の確率で記号 0 を発生して状態 S1 に遷移する。状態 S3 にいる場合，0.5 の確率で記号 0 を発生して状態 S1 に遷移するか，または 0.5 の確率で記号 1 を発生して状態 S2 に遷移する。このマルコフ過程の考え方は，あるテーマについて討議に参加した人の発言すべてを記録した発言録データベースに対してテキストマイニング手法を適用した討議過程の可視化，同じく発言録データベースから意見の推移などの文脈マイニングに適用できることを学生に示して，マルコフ過程を適用できる場面についての具体的な理解を図ることが重要と考える。

このマルコフ過程の考え方を拡張して，内部に複数の状態があり，外部からの入力に応じて状態を変化させるシステムへ適用することも可能である。内部に有限個の状態を持ち，入力により状態が遷移し，ある特定の状態になると出力するという図 4 に示すようなモデルで動作するシステムはオートマトン（自動装置やロボットのようなものを意味する）と呼ばれ

る。マルコフ過程では, 現在の状態から記号を生成しながら別の状態に遷移するのに対して, オートマトンでは，現在の状態において記号の入力があったときに別の状態に遷移する。このオートマトンは一般的に論理回路や言語理論のなかで学ぶものであるが，情報理論を学習した学生が論理回路や言語理論を学ぶとは限らない。そのため，情報理論のマルコフ過程のなかで，これと関連付けてオートマトンを学生が学び，文字列解析やその他一般的なプログラミングにも応用できるようにすることが重要と考える。

たとえば，図 5 に示すようなたし算のための状態遷移図を例にとり，学生にはまず，このオートマトンの意味を説明する。太い矢印のある状態は初期状態で，状態 S1 および S2 はそれぞれたし算のための 1 つ目および 2 つ目の数を受け付けている状態であり，二重丸の状態は 2 つの数字のたし算の答えを出力する状態であることを説明する。その上で，実際に “1582+265=” という文字列をプログラムに入力したら，文字列を 1 文字ずつ解析して状態遷移しながら 1847 という答えを出力するプログラムの作成方法を学生に説明する。図 5 の状態遷移図とは無関係にプログラムを作成することも可能であるが，状態遷移図と対応付けてプログラミングをしたほうがわかりやすいことを実例で示すと効果的であると考える。

図 6 が Java プログラムの例であり，変数 state に現在どの状態にいるかを記録するようにして，状態 S1,S2,S3 はそれぞれ 1,2,3 としている。状態の遷移は switch 文を使い，case で現在の状態を切り分け，それぞれの case のなかでそれぞれの状態で処理すべき内容を記述する。このプログラムを実行した結果は図 7 に示される。学生には，最初に状態遷移図に対応するプログラムの作成方法を学んでもらったあと，図 8 のような状態遷移図を使わないプログラムについても説明し，このプログラムの場合，入力された文字列の先頭から＋記号までの文字列を切り出して 1 つ目の数とするため，そのなかに数字でない文字が入っている可能性があることを学生に説明する必要がある。

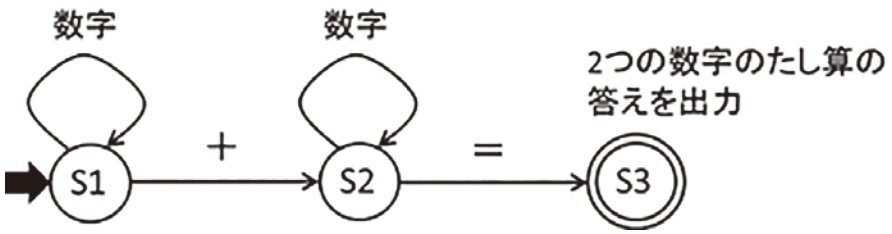

図 5　たし算のための状態遷移図

```
public static void main(String[] args) {
    // TODO 自動生成されたメソッド・スタブ
    Scanner stdIn;
    String shiki, t, t1, t2;
    int state, i1, i2, sum, i;
    String digit="0123456789";

    //「式を入力してください：」を表示し、式の入力を促す
    stdIn = new Scanner(System.in);
    System.out.print("式を入力してください：");
    shiki = stdIn.next();

    // 各種変数の初期化
    state = 1; // 初期状態としてS1を1として設定
    t1 = ""; //1つめの数の文字列を入れる変数を初期化
    t2 = ""; //1つめの数の文字列を入れる変数を初期化
    i = 0; // 入力された文字列を先頭から順番に1文字ずつ取り出すためのカウンター

    while(true) {
      switch (state){
        case 1: // S1の状態
          while(true) {
            t = shiki.substring(i, i+1);
            if (digit.indexOf(t) != -1) {
              t1 = t1 + t;
              i++;
            } else {
              state = 2; // 状態S2へ遷移
              i++;
              break;
            }
          }
        case 2: // S2の状態
          while(true) {
            t = shiki.substring(i, i+1);
            if (digit.indexOf(t) != -1) {
              t2 = t2 + t;
              i++;
            } else {
              state = 3; // 状態S2へ遷移
              i++;
              break;
            }
          }
        case 3: // S3の状態
          i1 = Integer.parseInt(t1);
          i2 = Integer.parseInt(t2);
          sum = i1 + i2;
          System.out.println("答えは"+sum);
          break;
      }
      if (state == 3) {
        break;
      }
    }
}
```

図 6　状態遷移図を使ったたし算の Java プログラム

```
式を入力してください：1582+265=
答えは1847
```

図 7　Java プログラムの実行結果

```
public static void main(String[] args) {
    // TODO 自動生成されたメソッド・スタブ
    Scanner stdIn;
    String shiki, t1, t2;
    int i1, i2, sum;

    //「式を入力してください: 」を表示し、式の入力を促す
    stdIn = new Scanner(System.in);
    System.out.print("式を入力してください: ");
    shiki = stdIn.next();

    // 各種変数の初期化
    t1 = ""; //1つめの数の文字列を入れる変数を初期化
    t2 = ""; //1つめの数の文字列を入れる変数を初期化

    // 1つ目の数字と2つ目の数字を＋記号と＝記号を目印にsubstringメソッドで取り出す
    t1 = shiki.substring(0,shiki.indexOf("+"));
    t2 = shiki.substring(shiki.indexOf("+")+1,shiki.indexOf("="));

    i1 = Integer.parseInt(t1);
    i2 = Integer.parseInt(t2);
    sum = i1 * i2;
    System.out.println("答えは"+sum);
    System.out.println("");
}
```

図 8　状態遷移図を使わないたし算の Java プログラム

5.　符号化と復号化

情報理論のなかでは，符号化と復号化は情報を通信路で送信する際に重要となる内容となる。また，データの圧縮，エントロピーとの関連など，データサイエンスを学ぶ学生にとっても重要な内容となっている。情報を記号列としてデジタルな通信路に送る場合や，記憶媒体に保存する場合には，0 と 1 だけの符号に変換する必要がある。発生確率が高い事象の情報は情報量が少なく，発生確率が低い事象の情報量は多いということが情報理論では説明さるので，この考え方は情報を符号化する際にも適用される。一般的に，各アルファベット文字に対する 2 進数表記を規定する ASCII 表などの文字コード表では，各文字の 2 進数表記の際，2 進数の桁数は同じになっている。しかし，情報理論では，発生確率が高い事象の情報は情報量が少なく，発生確率が低い事象の情報量は多いと考える。この考えに基づけば，情報の符号化においても，発生確率が高い記号の 2 進数表記は短く，発生確率が高い記号の 2 進数表記は長くということが当てはまることを学生に理解してもらうことが重要である。

こうした背景を学生に理解してもらったあと，ハフマン符号を学んでもらうことが効果的であると思われる。また，各アルファベット文字に対する 2 進数表記を規定する ASCII 表などの文字コード表などと，ハフマン符号化の関係も学生には正しく理解してもらうために，最初に ASCII 表などの文字コード表により 2 進数化したのちに，ハフマン符号に変換する

ことを説明することも重要と考える。アルファベットの各文字の出現頻度に応じたハフマン符号への変換をすぐに教えるのではなく，一度アルファベットの各文字 2 進数表記を規定する ASCII 表などの文字コード表により変換したあと，ハフマン符号に変換することを教えることは，学ぶ学生が実際に使うパソコン環境と整合性があって理解しやすいと思われる。

具体的には，6 個の記号 A, B, C, D, E, F のみを発生する情報源を考える。これらの記号を使って記述された文章を解析したところ，それぞれの記号の出現確率がそれぞれ 0.02, 0.18, 0,10, 0.31, 0,16, 0.23 となった場合の記号 A, B, C, D, E, F をハフマン符号へ変換してみる。ここで，記号 A, B, C, D, E, F は，コード表によりそれぞれ 000, 001, 010, 011, 100, 101 と表記されることとすると，これらの 2 進数表記を新たなハフマン符号へと変換することになる。情報源から発生される記号列が ABCDEF の場合，符号化される前は 000001010011100101 となる。これらの記号のハフマン符号への変換は以下のような手順によって行われることを学生に理解してもらう。

（1） 各記号をその出現確率が高いものから並べる
（2） 確率の最も低い記号 2 個をまとめ，これを 1 つまとめて確率を合計する
（3） この段階で確率の高い順に並べ変える
（4） 上記の（2），（3）の手順を繰り返し，最後に合計確率が 1 になるまで繰り返す

上記の手順は。図 9 のように，出現確率が高い順に並べ，確率の最も低い記号 2 個をまとめながら確率が 1 になるまで繰り返す。このハフマン符号化により，情報源から発生される記号列が ABCDEF の場合，ハフマン符号化される前は 000001010011100101 であるが，ハフマン符号化により，10000010011110101 となる，ハフマン符号では，出現確率が一番高い記号 D は 11 と符号化され，出現確率が一番低い記号 A は 1000 と符号化されている。記号 A, B, C, D, E, F の 6 個を 0 と 1 で記号化するには，一般的には 2 進数 3 桁で表記できる。これは，2 進数 3 桁であれば，0 から 7 までの 8 種類を表現できるからである。しかし，図 9 を見ると，記号 C および A は，2 進数 4 桁になっている。ここで，記号の発生確率を考慮すると，つまり，確率と符号長の積を 6 個の記号について合計すると，その値は 2.50 となる。これは，すべての記号を 2 進数 3 桁で表現するより，効率の良い符号化ということがわかる。表 4 では 6 個の記号 A, B, C, D, E, F のエントロピーの計算を行い，その合計が 2.32 となっている。このエントロピーは，その情報源から得られる平均的な情報量を示しているが，同時に符号化する際の 2 進数の桁数の下限値を示している。エントロピーが 2.32 の情報源であれば，その情報源から発生される各記号は，2.32 を切り上げた 2 進数 3 桁で表現できることがわか

る。しかし，ハフマン符号では，符号長の平均は 2.50 であり，3 桁より小さい。このことは，情報理論を学ぶ学生に理解してもらうことが重要である。また，このハフマン符号は，符号長が短い符号が符号長の長い符号の最初の部分と同じになっていないため，復号化する際には，先頭から 1 桁ずつ解析しながらもとの記号に戻すことができる。先頭から 1 桁ずつ解析しながら復号化する処理では，前述した状態遷移図をベースにするオートマトンが適用できることを学生に理解してもらうことは重要である。図 9 に示されるハフマン符号に変換された 0 と 1 の記号列を復号化するには，図 10 に示すような状態遷移図を作成して，これをベースにプログラム化することを学生に教えることにより，情報理論は字句解析やプログラミングと有機的に連携していることを理解してもらうことが可能となる。さらに，ハフマン符号は，情報を通信路に送信するためだけでなく，データの圧縮に適用されていることを理解してもらうことが重要と考える。つまり，情報源の符号化とは，情報を記号列であるデータに変換することであるということもできる。

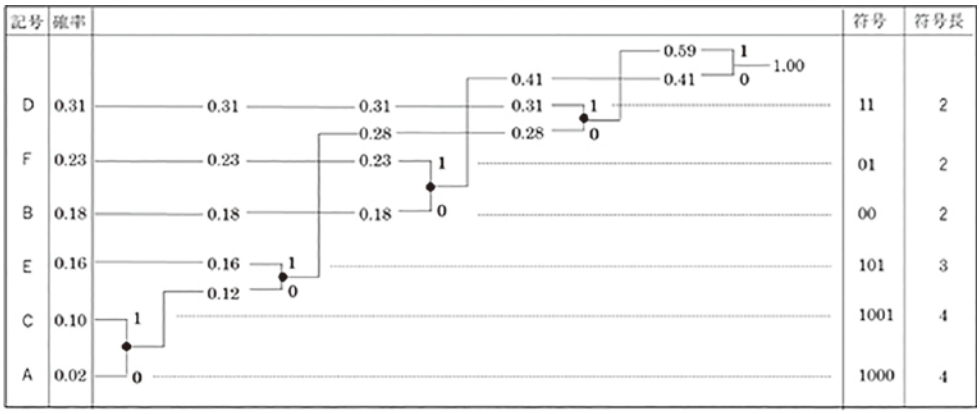

図 9　ハフマン符号化の手順

表 4　6 個の記号 A, B, C, D, E, F のエントロピーの計算

No.	記号	確率（P）	情報量（$-\log_2(P)$）	確率×情報量（$-P\log_2(P)$）
1	A	0.02	5,64	0.11
2	B	0.18	2.47	0.45
3	C	0.10	3.32	0.33
4	D	0.31	1.69	0.52
5	E	0.16	2,64	0.42
6	F	0,23	2.12	0.49
合計	−	1.00	17.88	2.32 ←エントロピー

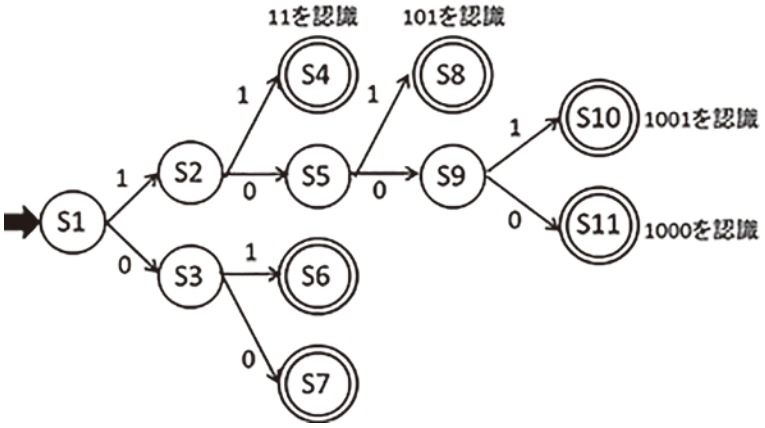

図 10　ハフマン符号を復号化するオートマトン

6.　おわりに

情報理論は，情報について学ぶものである。すべての情報がデジタル化される現在では，情報は通信路に送り出されるとき 0 と 1 の記号列のデータに変換されて送信され，受信側ではそのデータを復号化して情報に戻される。情報理論は, シャノンが通信路をモデル化して，情報源から生まれた情報を符号化して通信路に送り，受信側では受け取った記号列を復号化して情報に戻すことを学ぶことが主な目的ではある。情報理論は，この名前にあるように情報に関する理論ではあるが，理論として理解するだけに終わることなく，それをどのように応用できるかを学ぶ学生に理解してもらうことが教える側の責務であると感じる。

シャノンの通信路のモデルでは，通信の最中にノイズが入る可能を前提としている。情報をハフマン符号に変換し 0 と 1 の記号列で通信路に送り出したとき，通信路になかで発生するノイズのために 0 が 1 に，また 1 が 0 に反転してしまう可能性がある。このような反転が発生しているかを検出するために，0 と 1 に記号列を先頭から 8 個ずつ区切り，その 8 個のなかにある 1 の記号の個数を数え，奇数個なら 8 個の記号列の後ろに 1 を加えて 9 個の記号列で 1 の個数が偶数個になるようにし，8 個のなかに 1 の記号の個数が偶数個なら 8 個の記号列の後ろに 0 を加えて 9 個の記号列で 1 の個数は偶数個になるようにすることにより，記号列が 8 個から 9 個に 1 つ増えるが，ノイズによりどこか一か所の 0 または 1 が反転しているかどうかを判定することができる。データに信頼性を考えるとき，そのデータから得られる情報の信頼性は非常に重要であり，データそのものの信頼性を確保することいついて情報理論で学べることを学生に伝えることも重要と考える。

参考文献

［1］Shannon, C. E.（1948）. A mathematical theory of communication. *The Bell system technical journal*, 27(3), 379-423.
［2］瀧保夫（1980）.「情報論Ⅰ　―情報伝送の理論―」岩波全書.
［3］Zvi Kohavi（1979）. Switching and finite automata theory. TATA McGRAW-HILL PUBLISHING CO. LTD.

（いとう　のりゆき　東北学院大学教養学部　教授）

Education of Information Theory Intended to Apply to Data Science

Noriyuki ITO