圧縮に基づく言語判別

早矢仕 裕 [1)]，田中 久美子 [2)]
東京大学工学部計数工学科 [1)]，東京大学大学院情報理工学系研究科 [2)]

1　はじめに

言語判別は,「与えられた文書が何語で書かれているか」を判定する問題である．何語で書かれているかを判別したいテストデータを入力したとき，そのテストデータの言語を判別して出力する．このとき判別器は，各言語で書かれた訓練データを利用することで判別を行う．

言語判別の従来の研究は，多くとも数十程度の言語を対象とするものがほとんどである．そこで本研究では，100 以上の言語の判別について扱う．また，言語判別は，機械学習を用いる手法など様々なものが考えられるが，本研究では圧縮に基づく手法について扱う．圧縮に基づく手法では，各言語の訓練データを用いて圧縮器を構成し，入力であるテストデータをそれぞれ圧縮し，圧縮率を最大とする言語を選ぶことで判別を行う．

本研究の目的は，まず第一に，圧縮に基づく各手法の性能評価を行うことである．判別精度は，訓練・テストデータの大きさに依存するが，これらのデータの大きさを変化させたときの判別精度の変化を調べることで，圧縮に基づく各手法における判別精度の限界を確認する．また，判別精度が限界に達したとき，テストデータの言語が，特定の類似する言語と間違えて判別されることがあるが，そのような問題に対する考察を行う．

2　言語判別の手法の紹介

2.1　zip を用いた手法

最初に zip を用いた手法 [1] を紹介する．

圧縮の手法の一つとして，gzip などで実装されている Lempel と Ziv のアルゴリズム (LZ77) がある．このアルゴリズムは，先頭から順に符号化を行い，もし現在の位置から始まる記号列が以前現れていた場合には，その始点と長さに置き換えて符号化を行う．

このアルゴリズムを用いて，言語判別を以下の値 S を計算することで行う．訓練用の文書を B，与えられた文書を b としたとき，S を

$$S = \frac{L_{B+b} - L_B}{|b|}$$

と定義する．$B+b$ は B の後ろに b を付与した文書，L_i は文書 i を zip によって圧縮したときのビット長，$|b|$ は文書 b の長さである．この値は，b に含まれる文字列が B で現れている回数が多いと小さくなる．

このことから，訓練用の各言語の文書と，判別したい文書に対して S を計算し，S を最小とする言語をその文書の言語として推定する．

2.2　PPM を用いた手法

次に，Teahan，Harper[4] により考案された手法を紹介する．この手法は PPM(Prediction by Partial Matching) を用いた言語モデルを利用して言語の判別を行う．

PPM[2] は，マルコフモデルを利用して，直前に現れたいくつかの記号から次に現れる記号の予測を行う手法である．$D = x_1x_2...x_m$ を長さ m の文書 (x_i は i 番目の文字)，p_L を PPM を用いた言語 L の言語モデル，PPM で考える文字列長を n としたとき，

$$p_L(D) = p_L(x_1x_2...x_m) = \prod_{t=1}^{m} p_L(x_t|x_{t-1}x_{t-2}...x_{t-n})$$

としてモデル化を行う．このとき p_L を単純に頻度のみによって決めると，未知の記号が現れたときに対処できない．しかし PPM では未知の記号にも確率を割り振り (escape), 次数を 1 つ減らしたモデルを繰り返し考えることでこの問題に対処する．

PPM を用いたデータ圧縮によって，シンボルあたりの平均ビット長

$$\begin{aligned} H(p_L, D) &= -\frac{1}{n}\log_2 p_L(D) \\ &= -\frac{1}{n}\log_2 \sum_{t=1}^{m} p_L(x_t|x_{t-1}x_{t-2}...x_{t-n}) \end{aligned}$$

が計算できる．この値は，D が言語 L で生成された確率が高いほど小さくなる．

以上より，訓練用の各言語の文書から PPM を用いて言語モデルを構成し，判別したい文書をそれぞれの言語モデルを用いて算術符号化を行うことで圧縮する．そして最も圧縮されたものを，その文書の言語とするという手法で推定を行う．

2.3 クロスエントロピーを推定する手法

文書のエントロピーを推定する手法として,Wyner [5] によって考案された手法がある．

Wyner は，文書 $X = x_1x_2...x_nx_{n+1}...$($x_i$ は i 番目の文字) に対して，マッチング長 $L_n(X)$ を，x_{n+1} から始まる X の部分列 $x_{n+1}x_{n+2}...$ と $x_1x_2...x_n$ との最大マッチング長として定義し，L_n の平均を $\bar{L}$ とすると，$n \to \infty$ において，$|\bar{L} - \frac{\log_2 n}{H}|$ =O(1) となることを示した (H はエントロピー)．

この証明に基づき，

$$\hat{H} = \frac{\log_2 n}{\bar{L}}$$

を計算することで，文書のエントロピーを推定する手法が考案された．

Juola[3] は，Wyner の手法を応用し，2 文書間のクロスエントロピーを推定することで言語判別を行っている．

クロスエントロピーとは，2 つの確率分布の間の尺度であり，二つの分布を p，q としたとき

$$H(p,q) = \sum_i (-\log_2 q_i) \cdot (p_i)$$

で与えられる（確率分布 p，q のもとで，事象 i が起こる確率を p_i，q_i とする）．

Juola によって考案された Wyner の手法の応用は以下の通りである．2 つの文書 $X = x_1x_2...x_m$ と $Y = y_1y_2...y_n$ に対し，マッチング長 L_i を x_i から始まる X の部分列 $x_ix_{i+1}...$ と Y の連続した部分列との最大マッチング長で定義する．このとき，2 つの文書の間のクロスエントロピーを

$$\hat{H} = \frac{\log_2 m}{\bar{L}}$$

として近似的に推定する．ただし，$\bar{L}$ は L_i の平均である．

この Juola の手法により，訓練用の各言語の文書と判別したい文書の間のクロスエントロピーを推定して，最小となった言語をその文書の言語とするという手法で推定を行う．

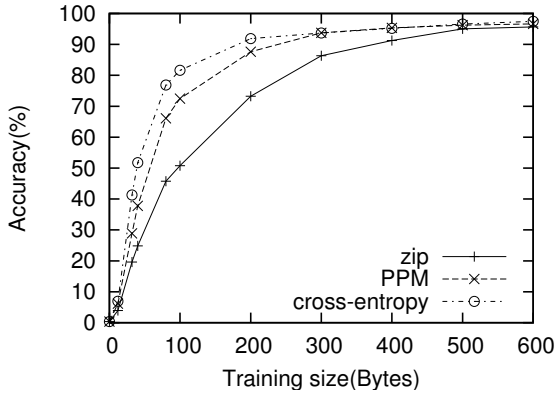

図 1: テストデータが世界人権宣言の場合の訓練データサイズと判別精度の関係

3 実験

前節で紹介した 3 つの手法を実際のデータに用いて言語判別を行い，訓練・テストデータのサイズを変化させて判別精度がどのように変化するかを確かめた．

まず，テストデータと訓練データに，275 言語の世界人権宣言のデータを用いて実験を行った．文字種の問題に対処するため，unidecode のソフトウェアを用いて，データは全て ASCII 文字に統一している．5 分割交差検定法によって，言語判別を行った．テストデータサイズは，最小のファイルの大きさである 555 バイトに統一し，訓練データサイズは 12，32，40，80，100，200，300，400，500，600 バイトとした．PPM の次数は 5，手法は PPMC を利用した．

図 1 に，訓練データサイズと判別精度のグラフを示す．横軸は訓練データサイズ (Bytes)，縦軸は判別精度 (%) である．判別精度は，275 言語中正しい言語が選ばれた場合を正解として，正しく言語判別ができた割合である．

いずれの手法も，600 バイトの訓練データを用いた場合は 95%程度という極めて高い精度での言語判別を達成していることがわかる．加えて，訓練データが 100 バイトと少ない場合であっても，PPM を用いた手法とクロスエントロピーを推定する手法は，70%以上の精度で言語判別を行っていることがわかる．

同様に，テストデータのサイズを変化させたときの判別精度の変化を確かめた．訓練データサイズは 600 バイト，テストデータサイズは 25, 50, 75, 100, 200, 300, 400, 500, 555 バイトとした．

図 2 に，テストデータサイズと判別精度のグラフを示す．横軸はテストデータサイズ (Bytes)，縦軸は判別精度 (%) である．この結果から，テストデータがある

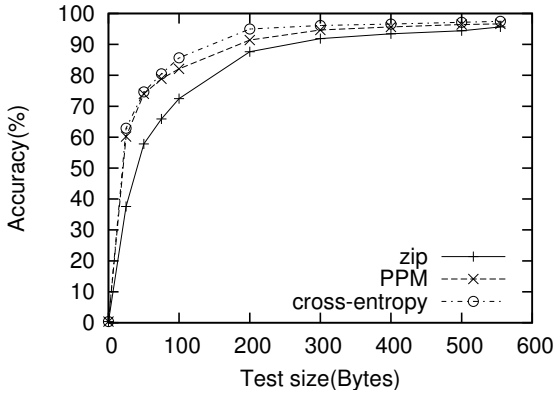

図 2: テストデータが世界人権宣言の場合のテストデータサイズと判別精度の関係

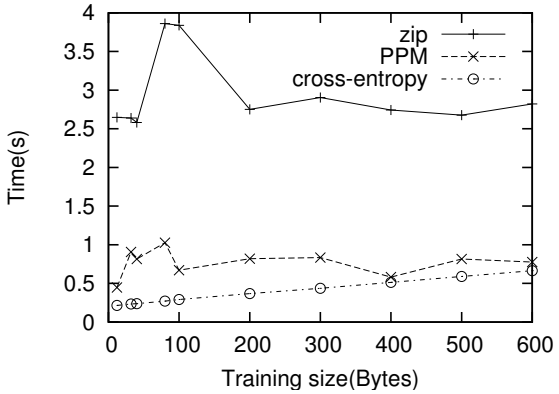

図 3: 訓練データサイズと 1 データあたりの判別時間の関係

程度小さな場合でも，高い精度で判別が行えていることがわかる．

また，訓練・テストデータを変化させたときに判別時間がどのように変化するかを調べた．訓練データと判別時間の関係を図 3 に示す．横軸は訓練データサイズ (Bytes)，縦軸は 1 データあたりの平均時間 (s) である．

クロスエントロピーを推定する手法では，各位置に対して最大マッチングを探しているため，訓練・テストデータサイズの増加に伴い，判別時間が線型に増加する．

PPM を用いた手法では，訓練データサイズを増加させた場合は，判別時間は単調に増加せず，テストデータサイズを増加させた場合は，判別時間が線型に増加する．これは，圧縮器を構成する訓練データの量が判別時間に大きく影響せず，圧縮するデータ量に比例して時間がかかるためだと考えられる．

zip を用いた手法では，訓練データサイズを増加させた場合も，テストデータサイズを増加させた場合も，

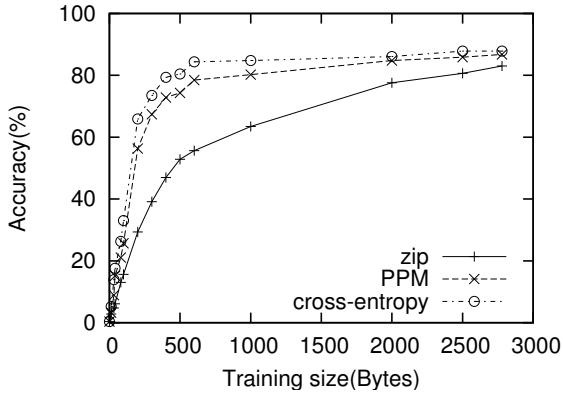

図 4: テスト・訓練データが異なる文書での訓練データサイズと判別精度の関係

判別時間に単調な増加はみられなかったが，これは，今回扱ったデータサイズが小さいためであり，増加させるデータ量を大きくした場合，判別時間は増加すると考えられる．また zip を用いた方法の判別時間が大きいのは，他の方法と異なり，テストデータに加え訓練データも圧縮していることによる．今回は，既存の zip 圧縮器を用いたが，訓練データを用いてテストデータのみを圧縮する zip 圧縮器を作ることで，速度の大幅な改善が見込める．しかし，図 1 や図 2 から，精度上は他の手法に劣る．

次に，テストデータと訓練データに異なる文書を用いた場合の実験の結果を示す．訓練データには 275 言語の世界人権宣言のデータ，テストデータには 92 言語の wikipedia のデータを用いた．wikipedia のデータも unidecode のソフトウェアを用いて，データは全て ASCII 文字に統一している．また，テストデータの大きさは 555 バイトに統一し，各言語 5 個ずつ計 460 個のデータに対して判別を行った．訓練データサイズは，世界人権宣言での実験の場合に加え，1000, 2000, 2500, 2779 バイトとした．PPM の次数は 5，手法は PPMC を利用した．

図 4 に訓練データサイズと判別精度のグラフを示す．横軸は訓練データサイズ (Bytes)，縦軸は判別精度 (%) である．訓練データを増やしたとき，判別精度の上限は，いずれの方法でも 85%程度であった．このとき zip を用いた手法は，他の手法に比べ精度が 3%程度低い．

同様にテストデータを変化させて実験を行った．訓練データサイズは 2779 バイト，テストデータサイズは，世界人権宣言での実験と同じである．図 5 にテストデータサイズと判別精度のグラフを示す．横軸はテ

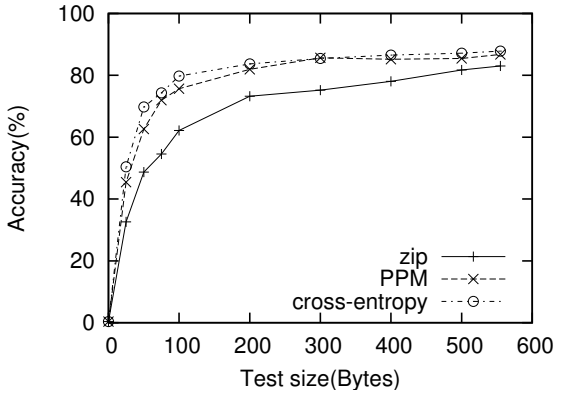

図 5: テスト・訓練データが異なる文書でテストデータサイズと判別精度の関係

ストデータサイズ (Bytes)，縦軸は判別精度 (%) である．世界人権宣言での実験同様に，ある程度小さな入力に対しても，ある程度の精度で判別が行えていることがわかる．

4 考察

まず前節の実験において，実際に誤って判別が行われた例をいくつか表に示す．

表 1: 誤って判別が行われた例

テストデータの言語	誤って判別された言語
Malay	Indonesian
Corsican	Italian
Luxembourgish	German

誤って判別される場合，ほとんどがテストデータの言語と類似した特定の言語と誤って判別されている．類似した言語と誤る場合には，そのデータの判別に複数の手法を組み合わせることなどが考えられる．

その他の場合としては，テストデータに，そのデータの言語でない固有名詞など必ずしも対象言語の語句ではないものが含まれている場合がある．このような問題に対処する方法は一概にはいえず，今後の課題として残されている．

次に，各言語のデータを圧縮したときの圧縮率に着目してみる．世界人権宣言を用いた実験（PPM を用いた手法）で，訓練データに German を用いたときのテストデータの平均圧縮率を求め，圧縮率が高いものを順に，表 2 に示す．

表から，同一の語族に含まれる言語の圧縮率が高く，上位に集まっていることがわかる．このことは，言語の系統樹の作成などに応用する可能性があることを示していよう．例えば Juola[3] は，クロスエントロピーを推定する手法を用いた言語の系統樹の作成を行っている．このような言語の系統樹の作成を，他の圧縮に基づく手法を用いても行うことができると考えられる．

表 2: German を訓練データとしたときの圧縮率

テストデータの言語	圧縮率 (%)
German	46.59458
Luxembourgish	55.42342
Dutch	57.4054
Afrikaans	57.94596
Danish	60.1081

5 おわりに

本研究では，圧縮に基づく手法を用いて多くの言語に対して言語判別を行い，データの大きさを変化させて性能評価を行った．

今回の実験では，データは全て unidecode のソフトウェアを用い ASCII 文字へ変換したが，このことによる誤りが見られた．そのため，はじめに文書の文字種を判別して，文字種毎に判別を行うことなどを考えている．

6 参考文献

[1] D. Benedetto, E. Caglioti, and V. Loreto: Language Trees and Zipping. *Physical Review Letters*, vol. 88, no. 4. (2002)

[2] J. G. Cleary, I. H. Witten: Data Compression Using Adaptive Coding and Partial String Matching. *IEEE Transactions on Communications*, vol. 32, no. 4. (1984), pp. 396–402.

[3] P. Juola: Cross-Entropy and Linguistic Typology. *Proceedings of New Methods in Language Processing 3*, Sydney, Australia, January 1998, pp. 141–149.

[4] W. J. Teahan, D. J. Harper: Using compression-based language models for text categorization. *Proceedings of the Workshop on Language Modeling and Information Retrieval*, Carnegie Mellon Univ., May 2001, pp. 83–88.

[5] A. J. Wyner: Entropy Estimation and Pattarns. *Proceedings of IEEE Information Theory Workshop*, Haifa, Israel, June 1996.