拡張文字ベースのHMMを利用した固有名詞抽出

新納浩幸
茨城大学 工学部 システム工学科
shinnou@dse.ibaraki.ac.jp

abstract

Extraction of proper nouns through extended character based HMM

In this paper, we report our method for NE task of IREX, and its result.
As the approach for NE task, we use the character based HMM basically. NE task can be regarded as the problem to assign an appropriate tag to each character in the given text. In this task, we prepare 36 kinds of tag. We set these tags as states of HMM, and use character tri-gram as the output symbol. In HMM, we can estimate the sequence of states in which observed output symbols pursue by the Viterbi algorithm. That is, we can estimate which tag is assigned to each character and consequently achieve NE task.
Our approach is almost same to Sekine's approach[1], but the character based method, the way to get output symbol costs and types of tag for characters unrelated to proper nouns are different points. Moreover, we give the part of speech to each character, and use this information in getting output symbol costs.
The character based approach has at least following two advantages: (1) unknown proper nouns can be handled, (2) a character in the word can be extracted as a proper noun. For example, the "日" can be extracted as a proper noun from the one word "来日".
Our result of NE task was about F-measure 60. It was bad. Moreover, we didn't succeed in above two advantages having an effect. We guess it was caused by small training data and no heuristics. We believe that our approach is appropriate for NE task.
Our method doesn't use heuristics depended on this task and the used language. Therefore our result may present the base line score.

1 はじめに

筆者は IREX の NE タスク部門に参加した．ここではそこで用いた手法と試験成績について報告する．

NE タスクは手作業で細かな規則と大規模の辞書を構築してゆけば，相当高い成績を納めることは予想できる．このような方向の研究も重要であるが，一方で移植の容易性のために機械学習手法を利用して抽出規則を自動獲得する研究も重要であろう．筆者は後者の方向でこのタスクに対するシステムを作成した．

従来より NE タスクに対する学習戦略としては幾つか報告があるが[1]，これらは主に英語が対象であり，直接日本語に適用できるかどうかは明らかではない．日本語に関しては関根の研究 [1] がある．その手法を概説すると以下のようになる．

[1]筆者による簡単なサーベイが [3] にある

固有名詞，例えば人名，の抽出は入力文の各単語に以下のようなタグをつける問題に一般化できる．

OP-CL ： その単語自身が人名
OP-CN ： 人名が複合語でその最初の単語
CN-CN ： 人名が複合語でその中間の単語
CN-CL ： 人名が複合語でその最後の単語

関根の研究では8タイプの固有名詞を扱っているので，合計32種類のタグと固有名詞とは無関係というタグ NONE の計33種類のタグを用意している．そして決定木を利用して，各単語が持つタグの確率を求める．次に Viterbi アルゴリズムによって，尤もらしいタグの列を生成する．この手法は，状態をタグとして，出力シンボルを単語としたときの HMM とほぼ同等である．状態を遷移するコストを決定木から得ていると捉えられる．

本手法は基本的にはこの研究と同様のアプローチをとる．ただし，以下の点で異なる．

- HMM の枠組みで行う．

  HMM の枠組みでも実行時の処理の違いはない．違いは状態を遷移する際のコストの与え方である．決定木などを利用した方が，様々な文脈情報を利用できるために精度的には良い結果が得られると考えられる（例えばタグを付ける単語のはるか後方の単語なども枠組み的には参照できる）．手法の簡潔性という観点から HMM の形をとった．ただし，本手法は純粋なマルコフモデルではない．シンボルが状態 i から状態 j に遷移する際の出力確率（コスト，得点）の定義に，始点の状態 i が依存しないからである．ただし出力シンボルを複雑にとっているので，実質は同等のものが利用されている形にはなっている．また状態遷移確率（コスト，得点）に状態 i が依存しているので，結果的には HMM と見なせる．

- 文字ベースである．

  HMM を単語ベースではなく，文字ベースとした．文字ベースの場合，未知語に対応できる可能性がある．例えば筆者は日本人名に関しては未知語であってもその文字の構成から，人名であることを推定できることを示している [2]．これは文字ベースの HMM により未知の人名が抽出可能であることを示している．また固有名詞抽出では「来日」という通常一単語として扱う文字列から「日」を「地名」として抽出する必要がある．この問題に対して，単語ベースでは統一的な処理が困難であるが，文字ベースではこの問題は生じない．

- NONE のタグを細分類する．

  NONE のタグが与えられる単語には，固有名詞（あるいはそれを同定するのに有効な単語）と接しない単語や，逆に頻繁に接する単語がある．このために NONE のタグを分割することは精度の向上に貢献する．

以上が主な相違点である．以下，本手法，試験の結果と考察，結論を述べる．

2 拡張文字ベースの HMM による固有名詞抽出

2.1 36種類のタグの付与

固有名詞抽出はその固有名詞を構成している句の始点の文字と終点の文字を見つければ良い．例えば，人名について考えれば，入力文の各文字に対して以下の5つのタグをつけることで人名を認定できる（図1参照）．

NS (name-start) 人名を構成している句の最初の文字

NE (name-end) 人名を構成している句の最後の文字

NM (name-middle) 人名を構成している句の最初と最後の文字の間の文字

NI (name-itself) その文字自身が人名

NONE (none) 人名とは無関係

図 1: 文字へのタグ付け

上記は人名に対してであるが，IREX の NE では，人名の他，組織名，地名，商品名，日付，時間，お金，割合のタグが設定されているので，それぞれに対して上記と同様のタグを用意する．また NONE は共通であることを考慮すると，合計 33 種類のタグが必要となる．この設定は関根の設定と基本的に同じである．そこでの研究では文字ではなく単語に上記のようなタグを付加している．

本研究では NONE の扱いが関根の研究とは異なる．ここでは固有名詞とは無関係という文字列に対しても，その始点と終点を与えることにした．つまり以下のようなタグを用意した．

NNS (none-start) 固有名詞とは無関係の文字列を構成している文字列の最初の文字

NNE (none-end) 固有名詞とは無関係の文字列を構成している文字列の最後の文字

NNM (none-middle) 固有名詞とは無関係の文字列を構成している文字列の最初と最後の文字の間の文字

NNI (none-itself) その文字自身が固有名詞とは無関係の文字列を構成している文字列

このようにタグの種類を増やす方が，一般にタグ付けの精度は増す．ただし十分な学習データが必要ではある．この場合，トレーニングデータ中でも NONE というタグは非常に多いので，効果はあると考えられる．結果的に，本研究では表 1 で示される 36 種類のタグを用意した．

NS	人名の最初の文字	DS	日付の最初の文字
NE	人名の最後の文字	DE	日付の最後の文字
NM	人名の間の文字	DM	日付の間の文字
NI	その文字自身が人名	DI	その文字自身が日付
OS	組織名の最初の文字	TS	時間の最初の文字
OE	組織名の最後の文字	TE	時間の最後の文字
OM	組織名の間の文字	TM	時間の間の文字
OI	その文字自身が組織名	TI	その文字自身が時間
LS	地名の最初の文字	MS	お金の最初の文字
LE	地名の最後の文字	ME	お金の最後の文字
LM	地名の間の文字	MM	お金の間の文字
LI	その文字自身が地名	MI	その文字自身がお金
AS	商品名の最初の文字	PS	割合の最初の文字
AE	商品名の最後の文字	PE	割合の最後の文字
AM	商品名の間の文字	PM	割合の間の文字
AI	その文字自身が商品名	PI	その文字自身が割合
NNS	無関係の最初の文字		
NNE	無関係の最後の文字		
NNM	無関係の間の文字		
NNI	その文字自身が無関係		

表 1: 付与するタグ

以上の結果，NEタスクの問題は,「与えられた文書の各文字に対して36種類のいずれかのタグを与える問題」に置き換えることができる．

2.2 文字ベースのHMMの利用

NEタスクが「与えられた文書の各文字に対して36種類のいずれかのタグを与える問題」であれば，自然に文字ベースのHMMが利用できる．

HMM M は以下の6つの組で定義される．

$$M = (S, Y, A, B, \pi, F)$$

S ： 状態の集合
Y ： 出力シンボルの集合
A ： 状態遷移確率（コスト，得点）の集合
B ： 出力確率（コスト，得点）の集合
π ： 初期状態確率（コスト，得点）の集合
F ： 最終状態の集合

上記したように，Sには36種タグを対応させる．また

$$F = \{NE, OE, LE, AE, DE, TE, ME, PE, NNE\}$$

とおく．また π に対しては，

$$\{NS, OS, LS, AS, DS, TS, MS, PS, NNS\}$$

のそれぞれの状態に1を与え，それ以外の状態には0を与えた．同様に，Aに対しては，有り得ない遷移（例えばNSからME）に対して0を与え，それ以外には1を与えた．あとはYとBを設定すれば，NEタスクを実行する文字ベースのHMM M が構築できる．

HMMでは出力シンボル系列がどの状態をたどってきたかをViterbiアルゴリズムより推定することができる．すなわち各文字にどのタグが付与できるかを推定でき，NEタスクが行える．

文字ベースのHMMを利用することで，未知語に対応できる可能性があること，また「来日」という単語から「日」が「地名」であるといったように，通常は1単語と考える文字列の部分文字列から固有名詞を取り出すことも自然に対応できる．

2.3 拡張文字列

YとBを設定すればHMMは構成され，固有名詞の抽出が可能である．ただし精度は期待できない．なぜなら，この枠組みでは，タグ付けに利用する情報が，問題の文字の前後数文字だけであるからだ．これは利用する情報としては非常に小さい．ここでは利用する情報を増やすために，文字に品詞の情報を付与する．

例えば,「今日学校へ行く」という文は以下のように形態素解析される[2]．

```
今日    今日    時相名詞
学校    学校    普通名詞
へ      へ      格助詞
行く    行く    動詞  子音動詞カ行促 基本形
```

本システムでは，時相名詞は“J”に，普通名詞は“D”に，格助詞は“9”に，動詞は“7”という記号を与えているので，結果として，“今”と“日”には品詞情報“J”が付与され，“学”と“校”には品詞情報“D”が付与され，“へ”には品詞情報“9”が付与され，“行”と“く”には品詞情報“7”が付与される．

今日学校へ行く
ＪＪＤＤ９７７

“J”や“D”などの品詞の分類としては，JUMAN 3.5で利用されている品詞を基にして43種類を用意した．

2.4 出力シンボルの設定と出力得点の算出

状態から状態へ遷移する際に出力されるシンボルをここでは文字tri-gramに設定した．例えば入力文が「今日学校へ行く」であれば，最初の状態から次の状態へ移る際に,「今日学」が出力され，次の状態へ移る際に「日学校」が出力されるという具合いに，1文字づつづらしながら，文字tri-gramが出力されてゆく設定にした（図2参照）．

状態 i から 状態 j へ遷移する際に，シンボル $C_1C_2C_3$ が出力される得点 $B_{ij}(C_1C_2C_3)$ を以下のように与えた．

[2]本システムではJUMAN 3.5を利用した

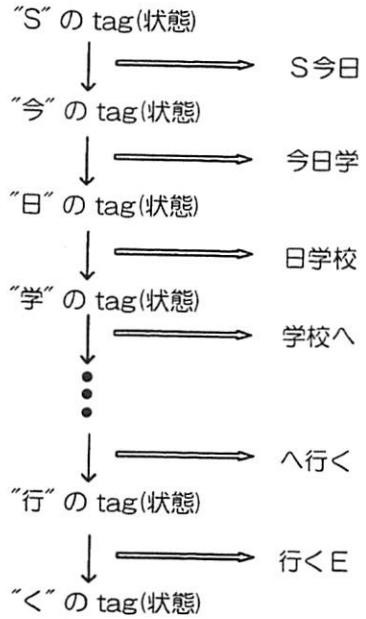

図 2: 出力シンボル

$$B_{ij}(C_1C_2C_3) = 8P_0 + \frac{4}{3}(P_1+P_2+P_3) + \frac{2}{3}(P_4+P_5+P_6) + P_7$$

ここで 8 や $\frac{4}{3}$ などの各 P_n につけられた重みには理論的な根拠はない．筆者の経験的な値である．また P_n の定義は以下の通りである．

$$P_0 = \frac{f_i(C_1\cdot C_2\cdot C_3)}{\sum_k f_k(C_1\cdot C_2\cdot C_3)}$$

$$P_1 = \frac{f_i(h(C_1)\cdot C_2\cdot C_3)}{\sum_k f_k(h(C_1)\cdot C_2\cdot C_3)}$$

$$P_2 = \frac{f_i(C_1\cdot h(C_2)\cdot C_3)}{\sum_k f_k(C_1\cdot h(C_2)\cdot C_3)}$$

$$P_3 = \frac{f_i(C_1\cdot C_2\cdot h(C_3))}{\sum_k f_k(C_1\cdot C_2\cdot h(C_3))}$$

$$P_4 = \frac{f_i(h(C_1)\cdot h(C_2)\cdot C_3)}{\sum_k f_k(h(C_1)\cdot h(C_2)\cdot C_3)}$$

$$P_5 = \frac{f_i(h(C_1)\cdot C_2\cdot h(C_3))}{\sum_k f_k(h(C_1)\cdot C_2\cdot h(C_3))}$$

$$P_6 = \frac{f_i(C_1\cdot h(C_2)\cdot h(C_3))}{\sum_k f_k(C_1\cdot h(C_2)\cdot h(C_3))}$$

$$P_7 = \frac{f_i(h(C_1)\cdot h(C_2)\cdot h(C_3))}{\sum_k f_k(h(C_1)\cdot h(C_2)\cdot h(C_3))}$$

ここで $h(C)$ は文字 C が持っている品詞の記号とする．$f_j(C_1\cdot C_2\cdot C_3)$ は，文字列 $C_1C_2C_3$ のトレーニングコーパス中の頻度である．ただし文字 C_2 につけられたタグが j であるという条件が付いている．また $f_j(h(C_1)\cdot C_2\cdot C_3)$ の意味は，文字列 HC_2C_3 のトレーニングコーパス中の頻度である．ただし H は品詞記号であり，この品詞記号を持つ全ての文字を表す．以下同様に，$f_j(C_1\cdot h(C_2)\cdot C_3)$ なども定義できる．

3 つの点を注意したい．

1 点目は，$f_j(C_1\cdot C_2\cdot C_3)$ や $f_j(C_1\cdot h(C_2)\cdot C_3)$ などには真ん中の文字（C_2 や $h(C_2)$）につけられたタグが j であるという条件がある，ということである．

2 点目は，各 P_n は状態 i に依存していない，ということである．つまり状態 i から状態 j に遷移するときの遷移の得点が，始点の状態 i に依存していない．これは，一見，マルコフモデルとは異なるモデルに感じられるが，状態遷移得点 $\boldsymbol{A}$ が始点の状態 i に依存するので，全体的にはマルコフモデルになっている．また遷移先の状態 j が C_2 につけられているので，C_1 の部分で実質的には始点の状態の情報を利用していると考えられる．

3 点目は，ここで用いたトレーニングデータは CRL コーパスだけである点である．自らもトレーニングデータを作成しても良かったが，学習手法を使って本試験に挑む他手法との比較を明確にするためにも，あえて CRL コーパスだけで行った．

3 成績と考察

作成したシステムの本試験及び予備試験[3]の結果は以下の通りである．NEScore の結果は一般課題のみ示す．また限定課題に対して特別な対策は行っていないことを注記しておく．

[3]本システムは予備試験には参加しなかった．本試験の後に実行してみた

	GLD	SYS	COR	MIS	OVG	REC	PRE
ORGANIZATION	361	192	126	235	66	34.90	65.62
PERSON	338	266	178	160	88	52.66	66.92
LOCATION	413	322	219	194	103	53.03	68.01
ARTIFACT	48	44	2	46	42	4.17	4.55
DATE	260	249	216	44	33	83.08	86.75
TIME	54	46	44	10	2	81.48	95.65
MONEY	15	11	10	5	1	66.67	90.91
PERCENT	21	14	14	7	0	66.67	100.00
OPTIONAL	86	-	-	-	-	-	-
?	0	0	0	0	0	-	-
ALL SLOTS	1510	1144	809	701	335	53.58	70.72
F-MEASURES							60.96

	F-値
本試験（一般課題）	60.96
本試験（限定課題）	58.46
予備試験	60.36

結果としては決して好成績とは言えない．特に目についた問題について述べる．

1. 例えば「～酒井一男さん（３６）～」という表現で括弧内の“３６”を商品名として誤って抽出してしまう．
   この人名の後に括弧をつけて年齢を入れる表現は，試験の問題では頻出し，一般課題では合計35種類の誤りをおかした．これは２つの原因から生じていると考える．１つは，本システムでは形態素解析システムとしてjuman 3.5を用いたが，その解析では“（）”と“「」”の 品詞づけが全く同じであり，ここでの品詞の設定でも分けて考えなかったためである．もう１つは本システムでトレーニングデータとして利用したCRL コーパスには，不思議なことに，上記した人名の後に括弧をつけて年齢を入れる表現がひとつもないことである．そのかわり，英数字だけで構成される商品名が“「」”で囲まれる表現はいくつかある．
   この点は結果に大きなマイナスを生じさせたが，学習という観点からは適切だったと考えている．
2. 住所の表現，例えば，「豊田市平戸橋町石平６１」などは部分的な表現を抽出し，全体を住所として抽出できていない．
   この誤りも目立った．試験前にこの問題には気が付いてはいたが，ヒューリスティクスを入れないという方針のために結果的に抽出できなかった．
3. 「同年△月」という表現自体を日付ととらえてしまう．
   「○年△月」という表現を日付とするように学習されてしまった結果である．この誤りも目立った．
4. 「欧米」という表現自体を地名ととらえてしまう．
   「欧米」から「欧」と「米」を取り出せることが，本手法の長所であったはずだが，これは達成できなかった．
5. 未知語へはほとんど対応できなかった．
   例えば，「グベルビョイ村」や「ドミニク・ボワネ」などが抽出できていない．カタカナが連続して一つの句を構成することは学習できているが，そのタグ付けまではできていなかった．
6. 英数字の略記，例えば「ＵＮＴＡＣ」などを抽出できていない．

(4),(5),(6) が問題として残るのは，本手法の狙いが十分には達成できていないことを示している．ただし学習の枠組み的には妥当だったと思う．上記の問題の多くは，トレーニングデータを増やしたり，パラメータを調整したりすることで多くは対応できると考える．

本手法が固有名詞を抽出するために利用している情報は前後の数文字と品詞だけである．これらの情報だけでは文脈が必要な場面において無力であることは明らかである．またタグをつける単位を単語にしていない点で，単語の区切り情報が失われている．これらの点から本来，この手法では高精度は望めない．ただし以下の長所もある．

- 未知語に対応できる可能性がある．
- 単語内の文字列から固有名詞を抽出できる．

  上記2点について，既に説明した（ただしここでは，これら点に関して，良い結果は得られなかった）．

- 開発が容易．

  この点を特に強調したい．本システムは実際精度を上げるためのヒューリスティクスを含めなかった．またこのタスクを目的とした辞書も用意しなかった．本モデルの設定には何の工夫もいらない．つまり対象の言語を知らずとも作成できる．このため本システムの成績はこのタスクの一種のベースラインと位置づけることもできる．

4 おわりに

IREX の NE タスクに参加した．そこで利用したシステム，手法について述べた．概略，文字ベースの HMM を利用した．品詞情報を付与した拡張文字を用いた点，それに基づいて出力シンボルのコスト（得点）を求めた点，NONE のタグを細分類した点などが工夫点である．精度的には高くはなかったが，対象言語のヒューリスティクスを入れていない点が長所と考える．当面の課題としては，学習データを増やし，本来本手法で抽出できるはずであるが，抽出できなかった表現を抽出できるようにすることである．

参考文献

[1] S. Sekine, R. Grishman, and H. Shinnou. A Decision Tree Method for Finding and Classifying Names in Japanese Texts. In *WVLC-6*, 1998.

[2] H. Shinnou. Revision of Morphological Analysis Errors Through the Person Name Construction Model. In *Machine Translation and the information Soup (AMTA-98)*, pp. 398–407, 1998.

[3] JEIDA 自然言語処理技術委員会.『自然言語処理システムに関する調査報告書』.（社）日本電子工業振興協会, 1999.