Web 文書からの人物情報の抽出

西田 成臣† 森 辰則‡
†横浜国立大学 大学院 環境情報学府 ‡横浜国立大学 大学院 環境情報研究院
E-mail: {aki,mori}@forest.eis.ynu.ac.jp

1 はじめに

電子化文書の増加に伴い，必要な情報を効果的に抽出する技術が求められている．このような情報の一つに人物情報が存在する．人物情報を抽出することはそれ自体に価値があるだけでなく，得られた情報を利用することで，名寄せや人物の評判情報といった，更なる活用が期待できる．このような背景の下，評価型ワークショップ NTCIR-7*及び NTCIR-8†では人物情報の質問応答を含むタスクが設定されている．

我々は，質問応答の一環として，Web 文書から人物に関する情報を抽出するシステムの構築を目指している．本稿では，利用者が人物に関して求める情報の一つとして，人物に特徴的な属性と属性値の情報に焦点を当てて抽出する手法について検討する．特に，一文及び，複数文の列に対して，属性と属性値が含まれるかどうかを機械学習により分類する手法を提案する．その際に用いるいくつかの素性を，評価実験により比較検討し，さらに，それらの素性の組合せを検討する．最後に，人物の名前をクエリとし，Web 文書から本手法に基づいて，人物に関する情報を抽出し提示する実証システムの実装について検討する．

2 研究背景と関連研究

質問応答とは，利用者からの質問に対する答えを返す技術のことであり，そのタスクの一つに，定義や理由，方法といった比較的長い言い回しを回答に必要とする non-factoid 型の質問応答がある．本研究では与えられた人名に対して，対象となる人物の情報を情報源から抽出することを目的としており，その情報は比較的長い言い回しを必要とすると予想される．ゆえに，人物情報の抽出は non-factoid 型の，特に人物の内容を明らかにする定義型質問応答の一環と考えることができる．定義の対象を人物や組織等でタイプ分けした時，その用語分布には大規模な差が見られるため，タイプを重視するシステムは適切であるという報告がある [1]．型ごとに処理方法を用意するためにシステム構築のコストが高いものの，回答に特化した部分システムを用意することができるため，精度の高い回答を得られる可能性が高いという利点がある．

人物に関する情報の抽出については，人物略歴の生成を目的として，要約ベースのアプローチが取られることが一般的である．例えば，Fadi らは人物の略歴を生成するため，wikipedia を利用した教師無し手法による複数文書からの自動要約システムを構築している [2]．しかしながら，人物の略歴を生成するような手法では，利用者が必要とする情報の多様性により，必要な情報が必ずしも含まれていない場合や，人物の評判のように対応していない情報は提供できない可能性がある．

そこで，利用者が人物に関する情報を必要とする際の動機に即して，本研究では，人物の情報を属性・属性値情報，関係情報及び，評判情報の三種類に分類できると考えた．その上で，本稿では，特に人物に特徴的な属性・属性値情報が含まれる文を抽出することに焦点を当てることとした．このような，人物に関する属性の抽出に関しては，WePS2‡において人物の属性獲得のサブタスクが設定されており，この中で Han らは，抽出ルールや固有表現抽出手法などを基に Web ページから属性の候補を生成し，分類を介して人物の属性としての確からしさの検証を行っている [3]．しかしながら，これは同姓同名解決のタスクを目的としたサブタスクであり，本研究ではあくまで，利用者に実用的な人物の情報を提供するシステムの構築を目標としている．

*http://aclia.lti.cs.cmu.edu/wiki/TaskDefinition
†http://aclia.lti.cs.cmu.edu/ntcir8/
‡http://nlp.uned.es/weps/weps-2

3 本研究におけるタスク定義

本研究の目的は，知りたい人物に関する属性と属性値情報を，情報源となる Web 文書から抽出することにある．その際に，抽出の単位としては文を想定することとした．予め文単位という広い範囲で抽出しておくことで，今後，人物の属性・属性値情報の精細な抽出にも繋がると考えられる．よって，対象となる人物に関する，属性及び，属性値が含まれる文の抽出が目標となる．本研究で定義している人物の属性と属性値とは，「<人名>の<属性> は<属性値> である。」と言えるものである．ただし，実際に人物に固有な属性・属性値情報など，人物に関する全ての属性を定義して獲得することは難しいため，本研究では，関根らにより人物を説明するのに一般的とされた 16 種類の属性とそれに対応する属性値を対象することとした [4]．以下の表 1に 16 種類の属性を示す．

表 1: 本研究で扱う属性

生年月日	誕生地	別名	職業名
所属組織名	作品	別名	学歴
同僚・師匠	場所	国籍	親類
電話番号	FAX 番号	電子メイル	Web サイト

3.1 人物コーパスの作成

前述の情報に関する調査及び，抽出の処理を考えるため，Web 文書から人物に関する情報が含まれる文書を収集することとした．そのために本研究では，Yahoo! 検索ランキングを利用した．まず，2007 年 11 月 12 日から 2009 年 3 月 3 日まで，一日単位で計測された検索ランキングの上位 30 位内のキーワードを収集する．集めた全キーワードに対して Chasen を用いて形態素解析を行い，形態素解析の品詞細分類が人名 - 姓と人名 - 名の両方の表現が含まれるキーワードを人物名であると定義した．この人物名をクエリとして，Yahoo! で検索を行い，検索された文書の上位 20 文書を収集した．本研究で収集した人名のクエリ数は 592 で，集めた文書数は 11800 文書であった．なお，集めた文書中には，文書を集めた際のクエリにあたる対象人物以外の人物の情報が含まれることがあるが，本研究では，対象人物の情報のみが記述されているとして扱うこととしている．

3.2 Web 文書における人物の属性・属性値情報

人物情報の抽出に先立ち，作成したコーパスから無作為に 100 文書を選び，Web 文書における人物の属性と属性値情報の調査を行った．これによると，Web 文書において，人物の属性・属性値情報は複数の文に亘って記述されることが一般的であった．その一方で，例えばある Web ページにおいて，女優・俳優といった職業名

を基にしたカテゴリ分類を行っている場合，その職業名は対象人物を説明する意図とは無関係にページに記述される．このような，人物に関する属性や属性値になりうる情報が，まばらに記述されることもあった．しかしながら，これらの情報は，事前知識がなければ対象人物に関する情報であると判断することはできないため，本研究では，抽出の対象外と考えることとした．ただし，まばらに現れる属性・属性値情報が含まれる一文内に，人名が含まれていれば，それが属性・属性値の対象であると考えられるため，抽出と対象とすることとした．

以上の目的および抽出対象の Web 文書の調査結果を考慮すると本研究のタスクは次のようになる．

1. 利用者からの入力として人名を受け取る．その人名にまつわる属性・属性値情報が含まれる文を抽出対象とする．
2. Web 検索エンジンで人名を検索し，獲得した Web 文書の各文を対象として，対象人物に関する属性・属性値情報が含まれているか否かを判定する．
3. 属性・属性値情報が含まれていると判定された文を，獲得した全 Web 文書から集めて，利用者に提示する．

4 提案手法

Web 文書の調査に際し，人物に関する属性と属性値の記述スタイルを見たとき，「〜生まれ」や「〜卒」といった特徴的なものが多く含まれているため，これらの特徴をより効果的に利用するということを考える．このような文の記述スタイルに着目した場合，文の係り受け構造を利用して機械学習を行う手法が効果を上げている [5]．これを踏まえて，本研究においても，文の記述スタイルを活かして，係り受け構造を利用し学習・分類を行うこととした．

4.1 文分類の方法

例えば，抽出したい対象が属性であれば，ある文に，対象人物の属性が含まれているか否かという正負のクラスの分類をすれば良い．文に対して，人物の属性が含まれるか否かを注釈したコーパスを準備し，各文のクラスを分類する分類器を機械学習により構成することで，未知の文に対し，対象人物の属性が含まれるか否かの分類が可能となる．本研究では，文の正負のクラスの学習と分類に，BACT-0.13 を使用することとした．BACT はラベル付き順序木の分類器を構成する，Boosting を用いた機械学習システムである．本研究では文に対して，CaboCha を使用して文節単位の係り受け構造を導出した後，各文に対して正負の分類クラスを与える．このときの文節単位での係り受け構造においては，表層表現を素性として利用する．

4.2 文分類のための素性の検討

学習・分類時に用いられる素性については，分類対象の文の表層表現のみによる係り受け構造に基づくものをベースラインとして考えている．しかし，表層表現に限らない素性を学習・分類データに付加することは可能である．そこで，本研究では，次に説明する素性を，表層表現に基づく素性集合に追加することを検討した．

形態素解析結果：形態素解析により得られた形態素の品詞細分類や原形などのいずれかを，ベースラインの表層表現に付加するか，表層表現自体をこれらのいずれか一方で置き換える．

位置情報：各文書をその文数を基に任意の数 n で分割した上で，学習・分類対象の文が，何分割目に現れるかを素性として付加する．

人物名：文書を検索・収集する際に利用した人物名を素性として付加する．

html の path 情報：Web 文書を html の DOM 木として考えたとき，根のノードから対象文までの間のノードに現れる html タグの系列を基に，任意の深さ m までの html のタグを素性として付加する．

前後文：分類対象の文の前後文についても同様に素性表現とし，これを対象文の素性に連結する．

5 分類精度に関する評価実験と考察

分類精度の評価に先立ち，収集した人物コーパスに対して，人物に関する属性・属性値の情報に対して，注釈付けを行うこととした．本研究では，検索文書には，文書を集めた際のクエリにあたる人物の情報のみが記述されているとして扱うとしているが，実際には，対象人物以外の人物の情報が含まれている場合がある．そのため，対象人物以外の属性・属性値情報に対しても注釈付けを行うこととした．これは，以降で記述する学習・分類に際して，対象人物以外の属性・属性値情報を含めて学習・分類を行えるようにするためである．

人物情報に関する注釈付けの作業では，文単位で，その文に人物の属性が含まれているか，属性値が含まれているか，あるいは属性と属性値の両方が含まれているかという三値の判断を行い，注釈付けを行う．ここで扱う属性と属性値は 3節で述べたように，人物を特定するのに一般的とされた 16 種類のみである．さらに，注釈付けした属性・属性値が，対象人物についてのものか，対象人物以外についてのものか，あるいは，対象人物とそれ以外の両者のものかという三値の判断も行い注釈付けを行う．

また，3.2節で人物の書かれ方を調査した結果について述べたように，人物の属性・属性値情報は複数文に亘って記述されることが一般的であった．そこで，人物の情報が複数文に亘って記述されている場合には，複数文に対して，一括して人物の属性と属性値が含まれるという情報を付加する注釈付けも行うこととした．この範囲による注釈付けについては，文単位で扱った 16 種類の属性と属性値に限らず，「<人名> の<属性> は<属性値> である」と成りうる属性・属性値情報については，すべて注釈付けの対象とすることとした．なお，文の列での注釈の場合にも，文単位での注釈と同様に，属性と属性値の対象についての三値の注釈付けを行った．以下に注釈付けの例を示す．

```
<bio_pas value="tb">
山本美憂（姉）
<bio_sen value="tv">スポーツコメンテーター</bio_sen>
<bio_sen value="tv">女子レスリング（元世界チャンピオン）</bio_sen>
<bio_sen value="tb">生年月日：１９７４年８月４日</bio_sen>
血液型：A型
身長：１５６ｃｍ
</bio_pas>
-------注釈内容---------
t: クエリの人物,o: クエリ以外,c: クエリの人物とそれ以外
a: 属性,v: 属性値,b: 属性と属性値両方
bio_sen: 文に対する注釈,bio_pas: 文の列に対する注釈
```

以上の方法に従って，人物コーパスから無作為に選択した 1000 文書に対して，人手により注釈作業を行った．この注釈付けをした 1000 文書を基に，分類精度の評価を行う．以下で行う学習及び分類では，十分割交差検定を行い，各クラスごとの Precision，Recall，F 値の平均を算出して評価する．なお，本研究では，文書単位での分割により交差検定を行うこととした．これは，同一文書内の文が，分類データにも学習データにも混在してしまうことを避けるためである．

5.1 属性・属性値の分類精度

文単位，並びに，文の列に対する注釈を基に，一文単位に属性や属性値が含まれるか否かの正負のクラスを付与し，学習・分類を行う．文単位に対する注釈を基にする場合では，正のクラスを，属性を含む文のみに与える場合，属性値を含む文のみに与える場合，属性か属性値

表 2: 正のクラスの分類精度

	文単位に基づく場合									文の列に基づく場合		
正のクラス	属性			属性値			属性・属性値			属性・属性値		
評価	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
ベースライン	68.6	35.6	46.9	70.1	26.3	38.2	73.2	25.9	38.2	73.3	21.4	33.1
1 位の素性	69.0	38.6	49.5	68.6	31.0	42.7	71.4	31.8	44.0	73.4	33.7	46.1
2 位の素性	70.7	37.1	48.7	68.5	28.1	39.9	70.6	28.9	41.0	67.7	23.7	35.1
3 位の素性	69.9	36.6	48.1	70.5	26.8	38.8	73.9	27.8	40.4	72.2	22.3	34.1
1 位と 2 位	69.0	38.6	49.5	66.6	33.8	44.9	69.6	34.2	45.9			
1 位と 3 位	70.6	38.8	50.1	68.3	30.6	42.3	70.8	32.3	44.3	70.7	33.6	45.5
2 位と 3 位	71.0	38.0	49.5	69.6	28.3	40.3	69.2	30.4	42.2	66.5	25.4	36.7
1 位と 2 位と 3 位	71.1	40.9	51.9	66.5	34.1	45.1	68.6	34.9	46.3			

かその両方を含む全ての文に与える場合の三種類の分類を行う．文の列に対する注釈を基にする場合では，一括して人物の属性か属性値が含まれるとしているため，この属性か属性値が含まれる文に正のクラスを与えて分類を行う．以下の表 3に，文単位及び，文の列に対する注釈を基に正のクラスが付与された文数を示す．なお，分類対象の全文数は 152595 文であった．

表 3: 正のクラスが付与された文数

	属性	属性値	属性・属性値
文単位	2408	5552	6519
文の列			11543

5.1.1 分類精度の結果

前述の，文単位及び，文の列に対する注釈を基にした四つの分類において，ベースラインである表層表現のみによる分類精度を求めた．この表層表現に 4.2節で述べた素性を一つ追加した分類精度を求め，さらに，加えた素性のうち，F1 が良かった上位三つの素性を組み合わせて加えた場合の分類精度も求めた．以上の求めた分類精度を表 2に示す．ただし，素性を一つ追加した分類精度については，F1 が良かった上位三つの素性の分類精度のみを示す．この上位三つの追加した素性は表 4の通りである．なお，表 4の「path(n)」は html の path 情報の深さ n を，「位置 (m)」は位置情報の分割数 m を，「品詞」は形態素解析結果の品詞細分類を，それぞれベースラインの表層表現に付加することを示し，「原形 (置)」はベースラインの表層表現を形態素解析結果の原形に置換することを示している．例えば，文単位に対する注釈を基に，属性を含む文を正のクラスとした場合，表層表現に追加した素性で最も良かった素性は品詞であった．

5.1.2 考察

ベースラインとなる表層表現に対して，様々な素性を付加することで，F1 に基づく分類精度は良くなった．また，それぞれのクラスに対して，特化した素性を組み合わせて利用することにより，さらなる精度の向上が認められた．その一方で，文単位及び，文の列に基づく分類精度の両方とも，Recall がかなり低い結果となった．この大きな要因の一つとして，人物情報の多様性が考えられる．本研究で扱っている人物の属性は前述の 16 種類のみに限定はしているものの，ある特定の人物特有の属性に注目したときに，それを指し示す表層表現は多様であるし，また，属性に対する属性値も，属性によってはある特定の人物や人物のグループに固有のものが多い．例えば，属性が職業に対する属性値は，「プロ野球選手」や「歌手」などが存在するし，歌手であれば，「リリースＣＤ」が属性の「作品」と同等の表現とも考えられ，更に多種多様な作品名が属性値になるという具合である．このように，ある特定の人物に特有な属性や属性値の学習・分類が上手くいっていないという可能性が考えられる．そこで次に，一般的な属性にのみ焦点をあてた場合の評価を行うこととした．

表 4: 追加した素性

	文単位			文の列
	属性	属性値	属性・属性値	属性・属性値
1 位	品詞	品詞	品詞	前後文
2 位	path(1)	path(1)	path(5)	path(5)
3 位	位置 (5)	原形 (置)	位置 (10)	品詞

5.2 一般的な属性の分類精度

ここでは，学習データとしては一般的な属性であるかは考えず，5.1と同様のデータを用いるが，分類データに対しては，一般的な属性のみを対象とした上での分類精度の再計算をして評価を行う．これにより，全ての属性を対象とした分類器でも，一般的な属性に関してのみはどの程度分類できているかを評価することができると考えた．具体的には，人手で注釈付けをしたコーパスを用いて，以下の処理を行う．

まず，人物の属性情報における一般的な属性を考えるため，注釈を基に，属性か属性と属性値の正のクラスが付与される文を収集する．収集した全ての文に対して形態素解析を行い，収集した文群における形態素の頻度を計算する．この中で，形態素の品詞細分類が名詞 - 一般と解析された形態素の上位 10 個を取り出す．以上の処理を，文及び文の列の注釈を基にそれぞれ行う．そして，分類対象の文を分類する際に，属性，もしくは，属性と属性値の両者が含まれる場合を正のクラスとして，これらが付与される文において，上位 10 個の形態素が含まれる場合は残し，逆に含まれない場合は，分類対象から除外した上で，分類精度を再計算することで評価を行うこととした．以下の表 5に，残したもののうち正のクラスが付与される文数を示す．なお，文単位，及び，文の列の注釈を基に取り出された形態素の上位 10 個はそれぞれ表 6の通りである．

表 5: 一般的な属性を含み正のクラスが付与された文数

	文	文の列
文数	1120	1547

表 6: 形態素の上位 10 個

文単位		文の列	
出身	ドラマ	出身	女子
作品	ＣＭ	テレビ	ドラマ
テレビ	番組	作品	投手
映画	生まれ	映画	ベスト
生年月日	写真	タレント	血液

5.2.1 分類精度の再計算

ベースラインの表層表現のみを用いて学習・分類した分類結果を基に，分類精度の再計算を行う．具体的には，文単位の注釈に基づいて属性が含まれる文に正のクラスを付与する場合と，文の列の注釈に基づいて属性と属性値が含まれる文に正のクラスを付与する場合において，正のクラスが付与される文について，上記の一般的な素性が含まれないものを除外した上で再計算する．文を除外する前の分類精度と，除外した上で再計算した分類精度を表 7に示す．

表 7: 分類精度の再計算結果

	文単位の属性			文の列の属性		
評価	Pre	Rec	F1	Pre	Rec	F1
除外前	68.6	35.6	46.9	73.3	21.4	33.1
除外後	49.7	58.7	53.8	53.3	47.9	50.4

5.2.2 考察

再計算の結果，文に正のクラスを付与する割合は多くなり，これにより，Precision と Recall の優劣が逆転し，F1 は大きくなった．しかし，一般的な属性に限定したとしても，精度が極端に良くなったわけではない．その大きな要因の一つとして，属性の出現の不偏性の高さが考えられる．例えば，「生まれ」や「出身」といった表記は，人物の属性情報とは関係のない文にも出現する．また，3.2節で述べたように，人物の属性になりうる表記が含まれていても，必ずしも人物の属性とは判断できずに抽出の対象外である文も存在する．このような場合，人物の属性情報とは無関係の文に誤って正のクラスを付与していた．この表記の不偏性の高さにより，逆に属性が含まれる文に正のクラスが付与されないこともあった．例えば，東京という表記は人物の属性情報とは関係のない文にも頻出しやすい．このため，東京という表記が含まれる属性の文に対し，関係の無い文として，負のクラスを付与してしまっていた．

以上の不具合の原因としては，本来学習すべき，人物の属性情報に特徴的な係り受け構造をうまく学習できておらず，あるひとつの形態素の表層表現に依存しやすいということが考えられる．ただし，5.1.1の結果を見ると，表層表現に別の情報を付加することで，精度の向上が認められた．例えば，html の path 情報や，文の位置情報などの大域的な情報が精度の向上に効果があることは，前述の，単一の表層表現への過度の依存を解消することに繋がると思われる．局所的な情報と言える形態素解析の品詞細分類を利用した際の精度向上からも，同様のことが考えられる．

これらのことは，属性だけでなく，属性値に関しても言えることであろう．しかしそれでも，5.1.1の結果を見れば，上記の素性を加えたとしても極端な精度の向上につながったとは言えない．これに対しては，例えば，ある人物に固有な情報に対しては，固有表現抽出器などの出力する固有表現クラスを表層表現の代わりに利用することで，表層表現に依らない素性として有効に扱えると考えられる．また，人物の属性・属性値情報は複数文に亘って記述されやすいことも，有効な情報につながると思われる．実際に，文の列に基づく分類では，分類対象の文の前後文を用いることで，大きな精度の向上が見られた．分類対象の文の前後の情報などをうまく素性として扱うことができればさらなる精度の向上につながるのではないかと考えている．今後としては，これらに限らず，様々な局所的，大局的な素性の検討を行っていきたい．

6 実証システムの実装

最後に，これまで説明してきた，人物の属性と属性値情報を抽出する実証システムの実装について説明する．3.2節で述べたタスク定義に従い，実証システムは次の流れにより実現される．

まず，利用者からのクエリを入力とする．ここの入力は，知りたい人物の名前を想定している．次に，知りたい人物の名前で Web 検索エンジンを検索し，上位 20 文書を獲得する．獲得した文書を上記の抽出手法に基づいて，文単位で分類する．分類において，属性や属性値が含まれると正のクラスが付与された文を抽出して提示する．このとき，抽出された文は，分類器である BACT-0.13 により与えられるスコアに基づきランキングした上で提示することとした．本研究で実装した実証システムの一部分を図 1に示す．

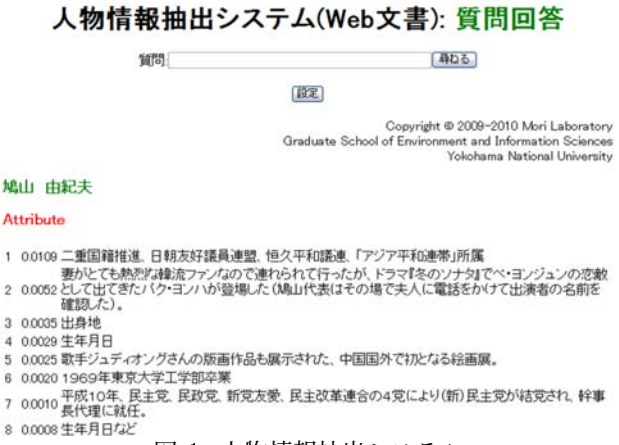

図 1: 人物情報抽出システム

7 おわりに

本研究では，質問応答の一環として，利用者から与えられた人名をクエリとし，クエリの対象となる人物を特徴付ける属性と属性値情報を，Web 文書から抽出する手法を提案した．さらに，提案した手法に基づき，人物の属性・属性値情報が含まれる文を Web 文書から抽出する，実証システムの実装も行った．

Web 文書から人物の属性と属性値情報を抽出する手法では，表層表現に加えて，様々な素性を加えることで精度を向上させることができることが確認できた．さらに，これらの素性を複数組み合わせることで，さらなる精度の向上を確認することができた．

今後の課題としては，属性・属性値に関しては，学習及び分類に用いる様々な局所的，大域的な素性の検討を行っていきたい．さらに，人物の属性・属性値に限らず，人物に関する別の情報を抽出する手法を検討し，利用者が必要とする情報の多様性に対応することで，より価値のある情報を提供していきたいと考えている．また，利用者への情報提示についても，より効果的な方法などを考えていきたい．

参考文献

[1] Kyoung-Soo Han, Young-In Song, Hae-Chang Rim. Probabilistic model for definitional question answering. In Proceedings of *SIGIR* 2006, pp. 212–219, 2006.

[2] Fadi Biadsy, Julia Hirschberg, Elena Filatova. An Unsupervised Approach to Biography Production using Wikipedia. Proceedings of ACL-08: HLT, pp. 807–815, 2008.

[3] Xianpei Han, Jun Zhao. CASINED:People Attribute Extraction based on Information Extraction. *WWW 2009*, April 20-24, 2009, Madrid, Spain.

[4] 関根聡, Javier Artiles, Julio Gonzalo. 人名の曖昧性解消評価型プロジェクト WePS. 言語処理学会 14 回年次大会発表論文集 pp.741–744, 言語処理学会, 2007.

[5] Taku Kudo, Yuji Matsumoto. A Boosting Algorithm for Classification of Semi-Structured Text. In Proceedings of EMNLP 2004,pp301-308,2004.