2K-03　リンク構造と時間軸を利用したWebページ間の関連づけと学習の適用

岡田哲弘　後藤文太朗
北見工業大学 情報システム工学科

1　はじめに

現在、WWWには膨大な数のWebページが存在している。それらのWebページは、ユーザにとって有用な情報を含んでいるものばかりではない。WWWのような広大な情報源から、ユーザが必要な情報を含んでいるページを発見するのは困難である。このような状況では、ユーザのWWW利用を支援するシステムが重要である。

ユーザのWWW利用を支援するシステムとして、ユーザのWWWへのアクセスのデータを蓄積し、それらのデータの統計情報などをWebページのコンテンツと統合することで、WWW利用を支援するシステム[1]がある。[1]では、ブラウザにユーザのWWWへのアクセスのデータを獲得可能なプロキシを設定することで、ユーザのWWWへのアクセス毎に、アクセス先URLやコンテンツ、アクセス時間等のデータ(以降、アクセスデータ)をデータベース(以降、DB)へ蓄積することを可能としている。

さらに、アクセスデータには、ユーザがWWWを利用する上での特徴が反映されるため、それらのデータの機械学習への利用[2]も行われている。[2]では、学習アルゴリズムとしてC4.5[3]を採用し、DB中の1つのアクセスデータから1つの訓練事例を作成することで作られた、訓練事例集合による学習の実験が報告されている。学習結果の利用としてインテリジェントブックマークを提案し、その有用性が示されている。また、[2]では、複数のアクセスデータを利用した1つの訓練事例の作成についても述べられており、そのような訓練事例集合を用いることによる、学習結果の有用性の向上が提案されていた。

そこで本研究では、ユーザのアクセス履歴、および、利用したWebページ間に存在するリンク構造などを利用し、1つの訓練事例とする複数アクセスデータの決定を行い、複数アクセスデータを単位とした学習実験を行うことにより、その有用性の確認を試みた。

2　アクセスデータからの訓練事例の作成

2.1　C4.5

学習技法は[2]と同様にC4.5[3]を採用している。C4.5とは、決定木学習技法の1つであり、属性とその値の対によって特徴付けられた大量の分類データ(訓練事例集合)から、ノードが属性、枝が属性値、葉がクラスに対応する、木構造の分類モデル(決定木)を帰納的に生成する技法である。生成された決定木を用いることで、未分類のデータ(事例)をあるクラスへ分類することができる。

Association between the Web Pages using Link Structure and Time-axis, and Application of Learning.
Tetsuhiro OKADA and Fumitaro GOTO
Dept. of Computer Science Kitami Institute of Technology
165, Koen-cho, Kitami, Hokkaido 090-8507, Japan

2.2　アクセスデータからの情報の取得

C4.5での学習には、あらかじめ訓練事例集合を用意する必要がある。データベース中のデータは、訓練事例ではないため、学習に利用するためにはデータベースから訓練事例集合を作成する必要がある。そのためには、アクセスデータから、様々な情報に関する値を求めることが必要である。以下にアクセスデータから獲得可能な情報の種類と、その値を求める関数を示す。

アクセスデータから取得可能な情報

(A)　Webページそのものの情報

情報	説明
freq_word(*Word*)	コンテンツ中の語句*Word*の頻度
presence_word(*Word*)	コンテンツ中の語句*Word*の存否
url	コンテンツのURL
link	コンテンツに含まれるリンク先URLの集合

(B)　ユーザのWebページの利用情報

情報	説明
stay_time	Webページでの滞在時間
access_count	Webページへのアクセス回数
access_time	Webページへアクセスした時刻
day_of_the_week	Webページへアクセスした曜日

情報の値を求めるための関数

関数	説明
value(*d*, *info*)	データ$d \in D$の情報*info*の値。*info*は前述の(A),(B)の情報。*D*はDB中の全データの集合を示す。

2.3　訓練事例の作成

アクセスデータの情報の値を用いて、訓練事例の作成を行う。訓練事例を作成するためには、属性とクラスがあらかじめ定義されている必要がある。以降、属性および、クラスが、図1のように設定されているものとする。

属性　属性の集合を$A=\{a1,\ldots,an\}$とする。各$a \in A$は att(*info*, *type*, *op*)と表し、*info*,*type*,*op*はそれぞれ、属性とする情報、その情報の取りうる値のタイプ、複数データに対する属性値の設定で用いるオペレータを表す。また、以下の関数により属性*a*の*info*,*type*,*op*を求められる。

info(*a*)=*info*　　value_type(*a*)=*type*　　operator(*a*)=*op*

クラス　クラスの集合を$C=\{c1,\ldots,cm\}$とする。各$c \in C$は、class(*ClassName*, *Conditions*)と表され、*ClassName*,*Conditions*は、それぞれ、そのクラスのクラス名、事例をそのクラスに割り当てる条件の集合を示す。それらは、以下の関数により求められる。

class_name(*c*)=*ClassName*
class_conditions(*c*)=*Conditions*

クラスの優先度　priority(*C*)=*ClassList*
*ClassList*はクラス名のリスト。リストの先頭のクラスが一番優先度が高い。優先度の順にクラスの割当が行われる

クラスオペレータ　operator(*C*)=*op*
複数データから事例を生成する際のオペレータ*op*を求める関数

図1　属性とクラス

このとき、訓練事例の作成は、図2のexample(*Ds*,*A*)により属性値を求められ、クラスはclass(*Ds*,*C*)により求められる。

Ds={*d1*,...,*dk*,...,*dn*} 任意のデータの集合

example(*Ds*,*A*)
E←{}, *i*←1
i≦*n*の間、以下を繰り返す
　　vi←decide_value(*Ds*,*ai*)
　　*E*に*ai*/*vi*を追加, *i*←*i*++
return *E*

decide_value(*Ds*,*a*)
V←{}, *k*←1, *in*=info(*a*), *op*=operator(*a*)
全ての*dk*∈*Ds*に対し、以下を実行
　　vk←value(*dk*,*in*)　(1つのデータに対する値の計算)
　　*V*に*vk*を追加
return value_operator(*V*,*op*)

decide_class(*Ds*,*C*)
Cs←{}, *op*=operator(*C*)
全ての*dk*∈*Ds*に対し、以下を実行
　　ck←class(*dk*,*C*)　(1つのデータに対するクラスの計算)
　　*Cs*に*ck*を追加
return value_operator(*Cs*,*op*)

value_operator(*V*,*op*)
op(sum,mean,max,min,major,minor)に従い、値の集合Vから、総和,平均,最大値,最小値,最も頻度の高いもの,低いものの算出

図2　データ集合からの訓練事例の生成

2.4　複数データの集合の生成

複数データの集合の生成は、図3のtraverse(*d*,*R*)関数により行う。全ての*d*∈*D*に対し、traverse(*d*,*R*)を行うことで、訓練事例集合が生成される。

traverse(*d*,*R*)
R=r(*R1*,*N*,*R2*,*R3*,*R4*)
Ds←{}, *u*←value(*d*,url)
D1,*D2*,*D3*,*D4*,*D5*,*D6*←{}
D1←traverse_time(*u*,*R1*)
if *N*>0{
　　D2←*S*(*D1*,*R2*)
　　全ての*d*∈*D1*に対し繰り返す
　　　　D3←
　　　　D3∪traverse(*d*,r(null,*N*,*R2*,*R3*,*R4*))
　　D4←traverse_link(*d*,*R3*)
　　D5←*S*(*D4*,*R4*)
　　全ての*d*∈*D5*に対し繰り返す
　　　　D6←
　　　　D6∪traverse(*d*,r(null,*N*-1,*R2*,*R3*,*R4*))
}
return *d*∪*D1*∪*D3*∪*D4*∪*D6*

traverse_time(*URL*,*R*)=*S*(*D*,url=*URL*)∩*S*(*D*,*R*)

traverse_link(*d*,*R*)
Ds←{}, *L*←value(*d*,link)
全ての*l*∈*L*に対し
　　Ds←*Ds*∪traverse_time(*l*,*R*)
return *Ds*

S(*D*,t(*P*,*A*,*B*))={*d*∈*D*|value(*d*,access_time)>(*P*-*A*)
　　∧value(*d*,access_time)<(*P*+*B*)}
S(*D*,url=*URL*)={*d*∈*D*|value(*d*,url)=*URL*}
S(*D*,all)=*D*
S(*D*,null)={}

図3　複数データの集合の生成

3　実験

●データ

単一ユーザによる1ヶ月間のブラウジングにより蓄積された、438のデータ。

●属性とクラスの設定

実験では、属性として、全Webページにおける総出現頻度の上位1000語の各Webページ中の頻度を用い、クラスの設定には、Webページの滞在時間を用いた。滞在時間の長いページがユーザにとって有用であるものと考え、滞在時間60秒以上、30秒以上60秒未満、30秒未満の3つのクラスに分類した。なお、起点は、全Webページとした。

複数ページに対する属性値は、各Webページの属性値の平均とし、クラスは、複数Webページにおいて、最も頻度の高いクラスとした。

●実験結果

いくつかの設定による実験の結果を表1に示す。

設定	R1	N	R2	R3	R4
S1	null	1	n(near(self),1)	null	n(near(self),1)
S2	n(near(self),3)	1	n(near(self),3)	null	n(near(self),1)
S3	n(near(self),3)	2	n(near(self),3)	null	n(near(self),1)
S4	n(near(self),3)	0	null	null	null

設定	誤り	推定誤り率
S1	76(17.4%)	31.7%
S2	70(16.0%)	30.2%
S3	63(14.4%)	29.3%
S4	57(13.0%)	25.1%

表1　実験結果

●考察

複数データから1つの訓練事例を作成することで、誤りが少し減少している。しかし、複数ページ単位による学習の有効性を評価するためには、実験結果を詳しく解析する必要がある。

4　おわりに

本稿では、ユーザのWWWへのアクセス履歴、および、利用したWebページ間に存在するリンク構造などを用いた、複数データ単位の学習による学習結果の向上を試みた。

参考文献

[1]　石川雅弘,後藤文太朗, "WWWアクセス活動とWebコンテンツの情報統合", 第60回情報処理学会全国大会講演論文集(3), pp.157-158, 2000

[2]　渥美尚紘,後藤文太朗, "SPIRALへのC4.5による学習モジュールの統合", 北見工業大学大学院工学研究科修士論文, 2000

[3]　J.R.Quinlan著, 古川康一監訳, "AIによるデータ解析", トッパン, 1995