MeSHの階層構造に基づくMEDLINEアブストラクトの自動分類

飯伏勝俊　Nigel Collier　辻井潤一
東京大学大学院理学系研究科情報科学専攻
{k-ibushi, nigel, tsujii}@is.s.u-tokyo.ac.jp

1. はじめに

医学論文データベース MEDLINE[3]に収録されている各アブストラクトには、MeSH[4]と呼ばれる階層構造を持つ検索語彙から選ばれた複数の検索語(MeSH term)が付与されている。現在は毎年10万件あるMEDLINEへの新規登録論文について専門家が手動でMeSH termを付与しており、データベース拡大の大きな障害となっている。

我々はこの作業を自動化するために、学習データからの機械学習に基づく文書分類手法[1][2]を適用する。すなわち、ある1つのMeSH termが付与されている文書が1つの文書クラスを形成するとみなし、システムがその文書が属すると判断したクラスに対応するMeSH termを付与する。

当初、自動的に付与したMeSH termと専門家が付与したMeSH termが一致することを目指した[8]が、MeSHが巨大なためデータスパースネスが問題となり困難であった。そこで、我々はMeSH が持っている階層構造を利用することにした。付与されているMeSH termに対応するクラスに、階層構造上で直系の子孫にあたるMeSH term と対応するクラスに属する文書も加えて文書分類の学習データとした。これによって、専門家の付与したMeSH termのうち41%を再現でき、もとのMeSH termかその直系の先祖・子孫にあたるMeSH termが含まれていれば正解であるとすると、71%の再現率となった。

2. MeSH

この章では本研究で自動分類を行う対象となるMeSH termとその階層構造であるMeSH treeについて述べる。

2.1. MeSH term

MeSHは統制された検索語彙集であり、そこに含まれる検索語を MeSH term と呼んでいる。1999 年の時点で main heading 19,232 語と supplementary concept 105,605 語、main headingに対する同義語250,000語が含まれている。MEDLINEに付与されるMeSH termのうち95%はmain headingから選択されているが、残りの5%はsupplement conceptか同義語から選択されている。各supplement conceptと同義語は対応するmain headingを持つので、本研究では学習セット中で main heading 以外の MeSH term が付与されている場合は対応する main headingに置き換えて取り扱っている。

2.2. MeSH tree

19,232個のmain headingのうち19,100個はMeSH tree と呼ばれる木構造の各ノードに対応している。MeSH treeには33,341個のノードが存在し、45%のmain headingは複数のノードに対応している。

各ノードは、"broader-than," "narrower-than," "related"といった関係で結ばれ、シソーラスを構成している。

3. 書分類手法

我々は、確率モデルの Single random Variable with Multiple Value(SVMV)に基づく文書分類手法（以後、SVMV classification）[1]と、Boosting アルゴリズムの一種である AdaBoost に基づく文書分類手法（以後 AdaBoost classification）[2]を、学習セット中で文書クラスに属する文書の数で切り替えて用いた。この章では、我々の手法の基になった 2 つの文書分類手法について述べる。

3.1. SVMV Classification

SVMV classification では文書 d がクラス c に属する確率 $P(c|d)$ を、

$$P(c|d)=\sum_{t_i} P(c|d,T=t_i)P(T=t_i|d) \quad (2.1)$$

と表す。ただし、文書中からランダムに抽出した term T が t_i であるという事象を $T=t_i$ とする。事象 c, d, $T=t_i$ が独立であると仮定すると、$P(c|d,T=t_i)=P(c|T=t_i)$ となり、これとベイズの定理より(2.1)式は

$$P(c|d)=P(c)\sum_{t_i}\frac{P(T=t_i|c)P(T=t_i|d)}{P(T=t_i)} \quad (2.2)$$

となる。右辺の各項を学習セットからの観測値で代用してモデルを構築する。

3.2. AdaBoost Classification

Boosting アルゴリズムは、かなり正確なクラス分類のルールを複数の weak hypothesis の組み合わせによって求めるという手法である。AdaBoost classification においては、ある term t_i を含む（含まない）と文書クラス c に属する（属さない）という weak hypothesis を積み重ねて文書 d が文書クラス c に属するか否かを判定する。

図 3.1 に、AdaBoost のアルゴリズムを示す。出力として得られる $h_{fin}(d)$ を新規ドキュメントに適用し、+1 なら文書クラス c に属するものとする。アルゴリズム中で呼ばれている関数 weaklearner の動作は、図 3.2 に示すとおりである。

Input:
N documents and labels: $<(d_1,y_1),...,(d_N,y_N)>$ where $y_i \in -1,+1$;
integer T specifying number of iterations
integer u_{rel} initial classification error for the document d_i which $y_i=+1$
integer u_{nrel} initial classification error for the document d_i which $y_i=-1$
Initialize:
$D_1(i)$ (for classification error, $D_1(i)=\frac{1}{Z_0}\begin{cases}u_{rel} & \text{if } y_i=+1\\ u_{nrel} & \text{if } y_i=-1\end{cases}$
where Z_0 is a normalization factor which ensures that $\sum_i D_1(i)=1$
Do for s=1,2,....T:
1. Call WeakLearn and get a weak hypothesis h_s
2. Calculate the error of h_s : $\varepsilon_s=\sum_{i:h_s(d_i)=y_i} D_s(i)$
3. Set $\alpha_s=\frac{1}{2}ln(\frac{1-\varepsilon_s}{\varepsilon_s})$
4. Update distribution:
$D_{s+1}(i)=\frac{D_s(i)exp(-\alpha_s y_i h_s(d_i))}{Z_s}$
where Z_s is a normalization factor.

Output the final hypothesis:
$h_{fin}(d)=\text{sign}(\sum_{s=1}^{T}\alpha_s h_s(d))$

図 3.1 AdaBoost algorithm for binary text filtering

各 term t について、

$$\varepsilon(t)=\sum_{i:t\in d_i, d_i\in Rel} D_s(i)+\sum_{i:t\notin d_i, d_i\in Rel} D_s(i) \quad (2.3)$$

を求める。ここで $t\in d$ とは term t が文書 d に出現するということを意味し、$d\in Rel$ とは文書 d がクラス C に属することを意味する。$\varepsilon(t)$ あるいは $1-\varepsilon(t)$ を最小とする term t を t_s とする。そして、weak hypothesis h_s を

$$h_s(d)=\begin{cases}+1 \text{ if } t_s\in d\\ -1 \text{ if } t_s\notin d\end{cases} \quad (2.4)$$

とする。

図 3.2 Weaklearner

4. 本研究で導入した手法

本研究では、3 つの手法を導入して、文書分類の精度を向上させることを目指した。1 つは、2 つの文書分類手法の混合手法であり、残りの 2 つは MeSH の階層構造を利用することである。

4.1. 混合手法

確率モデルである SVMV は、学習セット中で文書クラス c に属する文書の数 D_c が充分でなければ、学習セットから獲得できる観測値と実際のモデルとの間に誤差が生じる。MeSH term には文書頻度に大きなばらつきがあり、低頻度な MeSH term については学習文書が不足し、充分な学習が困難となる。また、AdaBoost では D_c が大きくなると、文書クラスに属するか否かを決定づける term を探索することが困難となる。特に、本研究で対象としている MeSH term は文書中に出現していても、必ずしもその文書に付与されるわけではなく、この傾向が顕著となる。

そこで、我々はD_cの値によって両者を切り替えて使用した。具体的には、D_cが閾値以上の時には SVMV を使用し、それ以外の時には AdaBoost を使用した。閾値は我々の予備実験の結果[8]から 1,000 とした。

4.2. MeSH 階層構造の利用

本研究では 2 つの MeSH 階層構造の利用法を提案する。

4.2.1. Wide Class

“癌”について書かれた文書と“肺癌”や“胃癌”について書かれた文書には共通して出現する term が多いと考えることができる。

ある MeSH term に対応する文書クラスに、その MeSH term を付与されている文書のみが属しているとするクラスの定義をここでは simple class と呼ぶ。これに対して、ある MeSH term に対応するクラスに、その MeSH term に対応する MeSH tree 上の node 以下の sub-tree に属する MeSH term を付与されている文書も属するものとするクラスの定義を wide class と呼ぶこととする。先にあげた例では、“癌”という MeSH term に対応する wide class には“肺癌”や“胃癌”といった MeSH term が付与されている文書も含まれることになる。

Wide class の使用には 2 つの狙いがある。1 つは各クラスに属する文書の数を多くすることによってデータスパースネスを防ぐことであり、もう 1 つは、1 つの MeSH term が MeSH tree 上の複数のノードに対応している場合、各ノード以下の sub-tree はそれぞれ対応する MeSH term が異なるため、1 つの MeSH term について異なる視点からのモデルを複数構築できるということの 2 点である。

4.2.2. Common descendant

ある文書が“癌”と“肺の病気”について書かれていることが既知である場合、我々はその文書が“肺癌”について書かれているものであろうと推測できる。この推測を行うためには、医学分野の知識が必要となるが、MeSH tree の構造はこのような推測に使える。

具体的には、ある文書がどのクラスへ属するか否かの判定を終えた後、その文書が属した 2 つのクラスに対応するノードの下位に共通の MeSH term が対応するノードがそれぞれ存在するならば、そのノードに対応するクラスにも属すると判定する。

5. 実験と結果

本研究では、まず、クラスの定義として simple class・ wide class のそれぞれを用いた場合の比較を行った。次に wide class を用いた文書分類の結果に common descendant を適用し、その効果を検証した。

5.1. OHSUMED

本研究では、情報検索のテスト用につくられた OHSUMED[5]というコーパスを実験に用いた。OHSUMED は MEDLINE のサブセットになっており、そのうちの 1 つ、OHSUMED.90 には 47,415 件の文献データが存在し、そのうちアブストラクトを含んでいるものが 10,478 件存在する。我々はこのうち 9,978 件のアブストラクトを学習セットとして用い、500 件のアブストラクトをテストセットとした。学習セット中には 8477 種の main heading が存在しているが、そのうち 2,270 種は文書頻度 1 であった。また、各アブストラクトには平均 12.6 個の main heading が付与されていた。

5.2. Term

本研究で用いる文書分類手法は、文書中に出現する term の有無や頻度情報をもとに文書分類を行う。本実験では、shallow パーザ ENGCG[6]にアブストラクトを入力して出現する単語の原形と品詞の情報を取得し、名詞・形容詞・動詞であり SMART システム[7]に付属する頻出語リストに登場しない語を term として利用した。

5.3. 評価方法

本研究では以下の 4 つの指標によって提案手法の評価を行う。

$$\text{Recall} = \frac{\text{正しく付与できたMeSH termの数}}{\text{専門家が付与したMeSH termの数}}$$

$$\text{Recall}^* = \frac{\text{それ自身か親子のMeSH termが付与された場合の数}}{\text{専門家が付与したMeSH termの数}}$$

$$\text{Precision}^* = \frac{\text{それ自身か親子のMeSH termが付与された場合の数}}{\text{我々の手法で付与したMeSH termの数}}$$

$$\text{F-value}^* = \frac{2 \times \text{Recall}^* \times \text{Precision}^*}{\text{Recall}^* + \text{Precision}^*}$$

5.4. Simple Class と Wide Class の比較

Simple class、wide class のそれぞれのクラス定義に基づいて学習セットからモデルを獲得し、テストセット中の文書に MeSH term を付与した結果が表 5.1 である。Recall*については、wide class のほうが高くなるようにデザインされているので当然の結果であるといえるが、Recall も wide class を用いることによって改善した。これは、学習に用いるデータの数が増えたことによってより適切なモデルを構築できたためだと考えられる。

また、wide class の場合は 1 つの MeSH term について付与されるかどうかの判断が対応するノードの数だけ行われる。このため、誤って付与される MeSH term の数も増大することが懸念されたが、Precision*の減少は Recall*の増加に比べればはるかに小さなものであり、F-value*は 0.12 改善している。

5.5. Common descendant の結果

5.4.節での wide class によってある文書が分類されたクラスに対応するノードの間で子ノードに共通する MeSH term が存在しないかを探索した。この結果 829 個の MeSH term が新たに付与され、この内 41 個が専門家の付与した MeSH term と一致した。また、62 個は専門家が付与したものより詳細な分類の MeSH term であった。ただし、Precision*は 0.54 から 0.51 へと減少した。

6. おわりに

本研究では文書分類手法を用いて、医学論文のアブストラクトに階層構造を持つ検索語、MeSH term を付与した。MeSH tree の上下関係を用いてデータスパースネスを回避し、専門家が付与した MeSH term の 41% を再現し、30% については直系の先祖・子孫にあたる MeSH term を付与できた。

表 5.1 各クラス定義による MeSH term 付与の結果

	Recall	Recall*	Precision*	F-value*
Simple	0.38	0.43	0.58	0.49
Wide	0.41	0.71	0.54	0.61

参考文献

[1] M. Iwayama and T. Tokunaga. A probabilistic model for text categorization: Based on a single random variable with multiple values. Technical Report TR94-0008, Department of Computer Science, Graduate School of Information Science and Engineering, Tokyo institute of Technology, April 1994.

[2] R. E. Schapire, Y. Singer, and A. Singhal. Boosting and rocchio applied to text filtering. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, pages 215-223, 1998.

[3] MEDLINE. http://www.ncbi.nlm.nih.gov/pubmed/.

[4] MeSH. http://www.nlm.nih.gov/mesh/meshhome.html.

[5] W. Hersh, C. Buckley, T. J. Leone, and D. Hickam. Ohsumed: An interactive retrieval evaluation and new large test collection for research. In Proceedings of the seventeenth annual international ACM-SIGIR conference on Research and development in information retrieval, pages 61-69, Dublin Ireland, July 1994.

[6] A. Voutilainen and P. Tapanainen. Ambiguity resolution in a reductionistic parser. In Proceedings of the sixth conference of EACL, pages 394-403, 1993. 1993.

[7] G. Salton, editor. The SMART System: Experiments in Automatic Document

[8] K. Ibushi, N. Collier, and J. Tsujii. Classification of MEDLINE abstracts. In Genome Infomatics 1999, pages 290-291, 1999.