F_036

NewsMLエディタのためのSubjectCode階層に基づく記事分類システムの試作
An Article Classifier based on a SubjectCode Hierarchy for a NewsML Editor

児玉政幸† 大圃忠親† 新谷虎松†
Masayuki Kodama, Tadachika Ozono, Toramatsu Shintani

1 はじめに

近年，ニュース管理／配信フォーマットの標準化が進められており，国際新聞電気通信評議会（IPTC）が策定したXMLベースのNewsMLが普及しつつある[1]．NewsMLフォーマットによる記事(以下，NewsML記事)は様々なメタデータを包含可能であり，メタデータを利用することで一般的なキーワードベースの検索よりも詳細な検索を行うことが期待される．NewsMLに含まれるメタデータの一つに，NewsML記事をカテゴリ毎に分類するために使われるSubjectCodeがある．SubjectCodeはSubject（大分類），SubjectMatter（中分類），SubjectDetail（小分類）といった3つの階層構造によって表現される．IPTCで定義されているSubjectCodeの総数は1,371種にも及ぶ．これだけの数があれば，記事を従来より詳細に分類することができる．しかし，これだけの数の中から編集した記事に適切なSubjectCodeを付加することは記者にとって非常にコストがかかる作業である．また，記事の主題は多岐に渡ることが多く，複数のSubjectCodeを適切に付加するとなると，さらに作業負担が強いられる．上記問題を解決するためには，記事の主題に沿ったSubjectCodeを自動分類してくれるサポート機構が記者にとって必要である．そうでなくとも，SubjectCode候補を数件(ここでは3〜5件を想定している)に絞り込むだけでも，記者を支援できると我々は考えている．

本研究の目的は，NewsMLエディタ上で記者が編集した記事をSubjectCodeに基づき分類し，記者のSubjectCode付加作業を支援することである．具体的には，NewsML記事集合の学習データを用いて，SubjectCode特徴ベクトルを各々作成し，編集された記事からSubjectCode候補を記者に提示することで支援を図る．本稿では，SubjectCode階層構造や概念類義語を用いることで，SubjectCode特徴ベクトルの各次元の重みを調整する手法について述べる．

2 NewsMLエディタのための記事分類

既存の研究では，テキストを決められたクラスタに分類する研究は盛んに行われている．福本ら[3]は文書中に現れる語に対し，重み付けの学習を行った結果を用いて文書の自動分類を行う手法を提案している．Buckleyら[2]は情報間の類似度をキーワードの重みを要素とするベクトルで表現し，ベクトル間の距離を計算することにより，情報間の類似度を計算している．本研究でも同様に，重み付けを利用し特徴ベクトルを作成していくが，記事の構造やNewsML特有のSubjectCodeの階層構造を利用することで重み付けを行っている．ニュースの特徴を十分に用いることで，ニュースに特化した特徴ベクトルを作成できると考えている．

また，猪野ら[4]はWordNetから抽出した共通概念を特徴語として利用し，テキスト分類に有効であることを示している．本研究においても，共通概念による特徴語の拡張は必要であると考えており，EDR概念辞書を利用して特徴ベクトルを作成している点において関連した研究である．

本研究では，分類システムの利用としてNewsMLエディタへの組み込みを想定している．記者にとってSubjectCode付加は，記事を編集する上で余分かつ煩雑な作業である．したがって，NewsMLエディタ設計の前提として記事編集過程をできるだけ妨げない工夫が必要である．記事編集過程を妨げないために，SubjectCode分類は記事編集中に行う．具体的には，記事編集者が1文を入力し終えた時点で，その都度，編集記事が分類システムにかけられる．記事編集者はシステムによって数種に分類されたSubjectCodeを選別するだけで良いので，記事編集中においてもSubjectCode作業負担が軽くなる．

† 名古屋工業大学 大学院情報工学専攻

3 SubjectCodeの決定

3.1 NewsML記事からの語抽出

SubjectCodeを分類するために，各SubjectCode毎にSubjectCode特徴ベクトルを作成する．SubjectCode特徴ベクトルは，NewsML記事から抽出した語を用いて作成する．まず，各SubjectCode毎にそれを含むNewsML記事を全て収集し，リスト化する．日本語係り受け解析器Cabocha[1]を用いて収集した記事から語を抽出する．抽出する際のルールとして，人物名は性と名を一つに，組織名とそれに続く未知語は一つに，連続する名詞は一つにする．人物名や組織名はCabochaによって解析された語をそのまま用いる．

また，SubjectCodeの意味をSubjectCode特徴ベクトルに追加することで，SubjectCodeの意味を反映した特徴ベクトルを作成できる．しかし，SubjectCodeの意味はSubjectCode名では表現しきれない．そこで，SubjectCode説明書からSubjectCodeを表現する語を抽出する．SubjectCode説明書は日本新聞協会が配布しているSubjectCodeに対する簡潔かつ的確な説明がなされている文書である[2]．SubjectCodeを的確に表現している語が多くあるため，特徴ベクトルに付加する．また，抽出した語からEDR概念辞書[3]を用いて類義語を抽出する．類義語を用いると言い換え表現や同義語にも対応できると期待できる．また，SubjectCodeの日本語名称として使われている語は，SubjectCodeを直接表現している語だと考えられる．したがって，日本語名称に含まれる語に重み付けを行う．ここでは，語の出現頻度を2倍とする．

3.2 SubjectCode特徴ベクトルの語の重み付け

SubjectCodeの階層構造を利用した語の重み付けの例として，SubjectCode「自動車事故」の特徴ベクトルの作成例を示す．NewsML記事では例えば，「自動車事故」はSubjectCodeの小分類に属しており，大分類の「災害・事故」，中分類の「事故（交通・運輸）」と共に記述される．これらの要素は概念上階層構造になっている．つまり，上位分類に含まれる概念は下位分類にも継承される概念だといえる．さらに，上位分類の語は下位分類の語よりも一般的な語である可能性が高い．裏を返せば，下位分類の語は大分類の語より記事の主題を特徴付けやすい語だと考えられる．このことから，より詳細なSubjectCodeを分類することができるように重み付けを行う必要がある．上位分類の語をベクトル要素にそのまま加えると下位分類の語と出現頻度が同じになるため，下位分類の語の出現頻度を高くする．ここでは，下位分類における語の出現頻度を2倍とする．すなわち，小分類の語の重みを1としたとき，中分類の語の重みは$\frac{1}{2}$，大分類の語の重みは$\frac{1}{4}$となる．

[1] http://chasen.org/~taku/software/cabocha/
[2] http://www.newsml.jp/
[3] http://www.iijnet.or.jp/edr/J_index.html

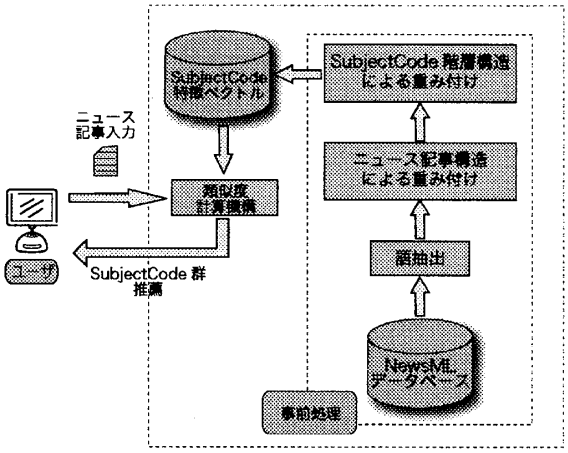

図 1: システム概要

また，記事を構造的に見てみると，一般的に記事の大意は先頭のパラグラフに記述される傾向がある．また，タイトルは記事本文全体を要約した文章だと考えられる．つまり，記事本文の先頭になればなるほど，その記事を特定するための語が含まれると考えることができる．Brandow ら [5] は，実験からこの考え方が有効であるとしている．本研究では語の重み付けに上記手法を適用する．まず，記事をタイトルと各パラグラフに分ける．タイトルから順に語の出現頻度の重みを高くする．ここでは，下位パラグラフにおける語の出現頻度を $\frac{1}{2}$ とする．すなわち，タイトルの重みを 1 としたとき，第 1 パラグラフの重みは $\frac{1}{2}$，第 2 パラグラフの重みは $\frac{1}{4}$ となる．

3.3 SubjectCode 特徴ベクトルを用いた SubjectCode 分類

3.1 で抽出した語に 3.2 で定義した重み付けを行い，特徴ベクトルを作成する．作成された記事ベクトルと SubjectCode 特徴ベクトルの類似度を計算することで，与えられた記事の SubjectCode を分類する．類似度計算は Singhal[6] の方法を利用した．Singhal の方法は，文書の長さによらず適切な類似度を求めることができる方法である．単語ごとに検索語の TF，及び検索対象の TFIDF の積を計算し足し合わせ，文書の長さにより正規化を行う．

図 1 に本手法を適用したシステムの概要を示す．

4 評価実験

本システムの評価方法として，*N-fold Cross Validation* (交差検定) を用いた．今回は N=3 として評価実験を行った．データセット E は 4,937 件の NewsML 記事を用いた．各 NewsML 記事には 1 つ，または複数の SubjectCode が付加されている．E からは Subject (大分類) 17 種 (17 種中)，SubjectMatter (中分類) 97 種 (375 種中)，SubjectDetail (小分類) 33 種 (979 種中)，計 147 種 (1,371 種中) の SubjectCode 特徴ベクトルを作成することができた．残り 1,224 種の SubjectCode は 4,937 件中の NewsML 記事には付加されていなかった．分類精度 $Prec$ は Subject，SubjectMatter，SubjectDetail 毎に式 (1)，(2) を用いて評価する．ここで，$Prec_n$ は第 n 回目の分類精度，L はテストセットの要素数，$correct_l$ は NewsML 記事 l 中の正しく分類された SubjectCode 数を示す．ここでの正しく分類されたとは，システムが提示した上位 α 番以内の SubjectCode に一致するという意味である．Subject，及び SubjectDetail に対しては $\alpha = 5$ とし，SubjectMatter に対しては $\alpha = 10$ とした．記事に含まれる SubjectCode 数は各階層において最大 5 であったため，上記の設定にした．表 1 に SubjectCode 分類の実験結果を示す．

表 1: SubjectCode 分類結果

	Subject	SubjectMatter	SubjectDetail
test0	0.84	0.51	0.57
test1	0.81	0.50	0.58
test2	0.83	0.48	0.57
平均	0.83	0.50	0.58

$$Prec = \frac{1}{N}\sum_{n=1}^{N} Prec_n \quad (1)$$

$$Prec_n = \frac{1}{L}\sum_{l=0}^{L} \frac{correct_l}{correct_l + \overline{correct_l}} \quad (2)$$

表 1 から，SubjectMatter と SubjectDetail が Subject に比べ精度が低いという結果を得た．これは SubjectCode 特徴ベクトルを作成する際の学習データの数が原因である．Subject は多くの記事に含まれていたが，SubjectMatter，及び SubjectDetail を含む記事は比較的少なかった．今後の課題として，SubjectMatter，及び SubjectDetail の特徴ベクトルを作成する際，より多くの学習データを与える必要がある．

5 おわりに

本稿では，特に SubjectCode 階層に着目し語の重み付けを行い SubjectCode 特徴ベクトルを作成し，それを用いてニュース記事を分類するシステムを作成した．本システムを NewsML エディタに組み込むことで，記事編集者は絞られた範囲内から SubjectCode を選択するだけで良く，編集時におけるコストの削減が期待できる．さらに SubjectCode が適切に付加されると，記事をより詳細なカテゴリで検索することが可能になる．今後，データ量を増やして SubjectCode 分類精度を上げると共に，本システムを実際に NewsML エディタに組み込み評価する必要がある．

参考文献

[1] 井上明，猪狩淳一，金田重郎，“ニュース配信のための国際データフォーマット NewsML:その概要と現状について”，情報処理学会論文誌，Vol.2002，No.056，2002.

[2] Buckley, C.,Allen, J., and Salton, G., “Automatic Routing and Retrieval Using SMART”, TREC-2, Infomation Processing Management, Vol.31, No.3, pp.315-326, 1995.

[3] 福本文代，鈴木良弥，“語の重み付け学習を用いた文書の自動分類”，情報処理学会論文誌，Vol.40，No.4，1999.

[4] 猪野陽子，松井藤五郎，大和田勇人，“WordNet からの共通概念抽出によるテキスト分類”，日本ソフトウェア科学会，第 22 回大会論文集，2005.

[5] R. Brandow, K. Mitze, and L.F. Rau, “Information Processing and Management“, Automatic condensation of eletronic publications by sentence selection., Vol.31, No.5, pp.675-685, 1995.

[6] Singhal A., Buckley C., and Mitra M., “Pivoted document length normalization”, In Proceedings of SIGIR'96, pp.113-126, 1997