2006－ICS－145（4）
2006／10／25

次世代記事編集システムにおけるNewsMLを用いたメタデータ配信について

児玉　政幸[†a)]　伊藤　正都[†b)]　大圃　忠親[†c)]　新谷　虎松[†d)]

On Metadata Transmission of NewsML for a News Management System

Masayuki KODAMA[†a)], Masato ITO[†b)], Tadachika OZONO[†c)], and Toramatsu SHINTANI[†d)]

Abstract. 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）によって策定されたXMLベースのNewsML(News Markup Language)が登場した．本研究では，ニュース管理／配信を行う環境とニュース編集を行う環境を土台とした次世代記事編集システムを構築する．ニュース管理／配信に関しては，NewsMLとは独立させた記事配信メタデータ記述言語NVML(News Value Markup Language)を提案する．NVMLの利用目的は，1)NewsML文書とのリンク管理，及びNewsML文書配信，2)ニュース価値情報付加，の2点である．リンク管理とコンテンツ管理を別々にすることで，ニュース配信者，及び受信者間で保持する各文書を整合性を保ちリンク管理することができる．また，ニュースに価値を付けることで，価値に基づいたニュース検索や一覧表示を行えたり，将来的なニュースの再利用を行うことができる．ニュース編集については，ニュース予測入力候補支援システム，及びニュースカテゴリの自動判定システムを試作した．

Keywords. NewsML，メタデータ，NVML，継続性，分類

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）によって策定されたXMLベースのNewsML(News Markup Language)が登場した[1]．NewsMLはXMLや他の適切な標準，仕様を元に，ニュースにコンパクト，かつ拡張性が高く，柔軟な構造化の枠組みを提供する．また，電子的なニュース，ニュースの集合，それらの間の関係，及び関連のメタデータの表現をサポートする．

本稿では，ニュース管理／配信システム，及びニュース編集システムについて説明する．ニュース管理／配信においては，NewsMLから配信，及び管理フェーズを分離し，さらにニュースの価値付けを行っていくことで，ニュースの再利用を目指したニュース管理システムについて説明する．ニュース編集においては，ニュースの継続性を利用したニュース作成支援システム，及びニュースカテゴリの自動付加を目的としたSubjectCode割当システムの2点について説明する．我々は，ニュース管理／配信を行う環境とニュース編集を行う環境を土台としたプラットフォームを次世代記事編集システムと呼ぶ．

NewsMLはニュース管理／配信フォーマットとして提案されているが，我々はNewsMLを用いたニュース管理／配信について問題があると考えている．従来，NewsMLのみを用いてニュースコンテンツの作成／管理／配信を行っているが，独立した各操作に関する情報を混合しているため，ユーザはNewsML文書を作成，管理しづらくなっている．また，ニュースコンテンツとは別にそのニュースの価値を受信者側に配信することは非常に重要である．

そこで我々は記事配信メタデータ記述言語NVML(News Value Markup Language)を提案する．NVMLの目的はニュース配信者と受信者との円滑な

† 名古屋工業大学情報工学専攻，〒 466-8555 名古屋市昭和区御器所町
a) E-mail: kodama@ics.nitech.ac.jp
b) E-mail: itomasa@ics.nitech.ac.jp
c) E-mail: ozono@nitech.ac.jp
d) E-mail: tora@nitech.ac.jp
(注1)：著作権は（社）情報処理学会にある．

ニュース配信，管理，及びニュースの価値付けである．リンク管理とコンテンツ管理を NewsML から切り離すことで，ニュース配信者，及び受信者間で保持する各文書を整合性を保ちリンク管理することができる．また，ニュースの価値付けにより，ニュース配信者は受信者に NVML 文書を配信することで，どのニュースを大きく取り扱うか，どのニュースが今後重要になってくるかを受信者側に指示することができる．同様に，受信者は NVML 文書を参照することで，大量のニュース集合からどのニュースが重要かを一覧することもできる．

また，現状の NewsML を用いたニュース作成環境には 2 つの問題があると考えている．

[問題 1] メタデータ付加支援環境が未整備

[問題 2] メタデータが活用されていない

NewsML のメタデータは人手による入力が必要なデータがあるが，ユーザは最低限ニュース記事のみを書けばよいので，メタデータ付加作業はユーザにとって非常に煩雑な作業になる．また，NewsML のメタデータは大量のニュースアーカイブを管理／再利用するのに必要不可欠になってくるが，問題 1 が存在するためニュースアーカイブには十分にメタデータが付加された NewsML 文書が蓄積されていない．

本稿で提案する次世代記事編集システムでは，ニュース編集においてメタデータ付加支援は必要不可欠であるため，本研究ではカテゴリメタデータ付加を支援するシステムを試作した．また，ユーザの入力補助を行うことを目的としたニュース予測入力候補支援システムを試作した．

2. NVML と NewsML

2.1 記事配信メタデータ記述言語 NVML(News Value Markup Language) 概要

NVML とは，従来 NewsML で行っていた配信／管理部を独立して制御するための記述言語である．NVML 文書の利用目的は，1)NewsML 文書メタデータ配信／リンク管理，2) ニュース価値情報付加，の 2 点である．

配信／リンク管理部は，従来行っていた NewsML での管理／配信部分を切り離した部分である．また，ニュースコンテンツのメタデータを価値情報と定義する．価値情報を用いることで，ニュースの重要度に基づいたニュース再利用やニュース配信を行うことができる．

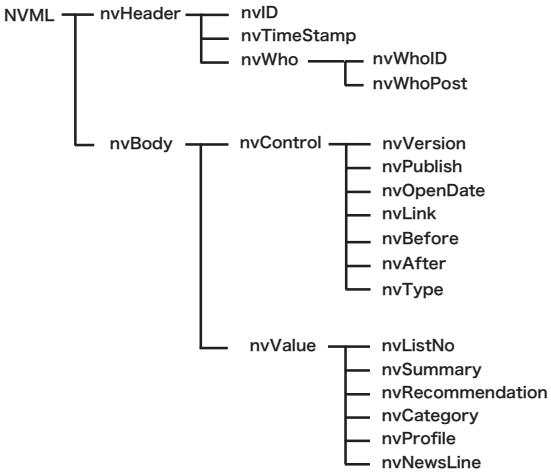


図 1　NVML のメタデータ階層構造

表 1　NVML 仕様

タグ	用途	データタイプ
⟨nvID⟩	NVML 文書 ID	数値型
⟨nvTimeStamp⟩	NVML 文書作成日時	数値型
⟨nvWhoID⟩	NVML 文書作成者 ID	数値型
⟨nvWhoPost⟩	NVML 文書作成者の出稿部署	数値型
⟨nvControl⟩	NVML 文書管理情報	
⟨nvVersion⟩	NVML 文書バージョン	数値型
⟨nvPublish⟩	NVML 文書配信日付	数値型
⟨nvOpenDate⟩	ニュース解禁日時	数値型
⟨nvLink⟩	NVML 同士のリンク情報，NVML-NewsML 間のリンク情報	文字列型
⟨nvBefore⟩	修正前情報	文字列型
⟨nvAfter⟩	修正後情報	文字列型
⟨nvType⟩	NVML 文書種類	文字列型
⟨nvValue⟩	ニュースコンテンツの価値情報	
⟨nvListNo⟩	NVML 文書内でのインデクス	数値型
⟨nvSummary⟩	ニュースサマリ	文字列型
⟨nvRecommendation⟩	ニュースの重要度	数値型
⟨nvCategory⟩	ニュースカテゴリ	数値型
⟨nvProfile⟩	ニュースプロファイル	文字列型
⟨nvNewsLine⟩	ニュース本文の行数	数値型

図 1 に NVML のメタデータ階層構造を，表 1 に NVML の各メタデータの仕様を，図 2 に NVML 文書の例を示す．

NVML 文書はヘッダ (nvHeader) と内容部 (nvBody) に分けることができる．

nvHeader には NVML 文書の ID(nvID)，作成日時 (nvStamp)，作成者 ID(nvWhoID)，作成者部署

```
<NVML>
 <nvHeader>
  <nvID>0011343</nvID>
  <nvTimeStamp>200607211535</nvTimeStamp>
  <nvWho>
   <nvWhoID>012</nvWhoID>
   <nvWhoPost>001</nvWhoPost>
  </nvWho>
  <nvType>新規</nvType>
 </nvHeader>
 <nvBody>
  <nvControl>
   <nvVersion>0</nvVersion>
   <nvPublish>20060722</nvPublish>
   <nvLink>
    <nvNewsMLLink nvListNo="0">
     20060722DHD31__________01
    </nvNewsMLLink>
     ......
    <nvNewsMLLink nvListNo="40">
     20060722DHD79__________01
    </nvNewsMLLink>
   </nvLink>
  </nvControl>
  <nvValue nvListNo="0">
   <nvSummary>葬儀会社８億所得隠し</nvSummary>
   <nvRecommendation>2</nvRecommendation>
   <nvCategory>11013000</nvCategory>
   <nvProfile></nvProfile>
   <nvNewsLine>40</nvNewsLine>
  </nvValue>
    ......
  <nvValue nvListNo="40">
   <nvSummary>政策決定会合控え続落</nvSummary>
   <nvRecommendation>0</nvRecommendation>
   <nvCategory>04008001</nvCategory>
   <nvProfile></nvProfile>
   <nvNewsLine>15</nvNewsLine>
  </nvValue>
 </nvBody>
</NVML>
```

図 2　NVML 文書の例

(nvWhoPost)，及び種別 (nvType) などの情報が記述される．

内容部はさらに，NVML 文書を管理する部分 (nvControl) とニュースコンテンツの価値を記述する部分 (nvValue) とに分けることができる．

nvControl には NVML 文書のバージョン (nvVersion)，NVML 文書をいつ配信するか (nvPublish)，どのニュースに解禁日時を指定するか (nvOpenDate)，リンク管理情報 (nvLink)，NVML 文書が新規か更新か (nvType)，修正前情報 (nvBefore)，修正後情報 (nvAfter) などの情報が含まれる．

nvValue にはニュースのインデクス (nvListNo)，サマリ (nvSummary)，重要度 (nvRecommendation)，カテゴリ (nvCategory)，プロファイル (nvProfile)，行数 (nvNewsLine) などのニュースコンテンツに関わる情報が含まれる．特に nvRecommendation は NVML の中で価値が高いメタデータであり，ニュース配信者が各ニュースの価値付けを行うために利用する．具体的な価値付けは，最重要，重要，一般の 3 分類に重要度を振り分ける．ニュース配信者は何本もあるニュースの中から最重要となるニュースを選出し，nvRecommendation を付加することで受信者側に配信者側の意思を伝えることができる．また，大量のニュースアーカイブから過去に重要になったニュースの再利用を行うこともできる．nvProfile にはニュースをベクトル空間モデルやブーリアンモデルなどの検索モデルに変換したデータを格納する．nvProfile のデータは関連ニュースを調べたいときなどに利用できる．

ニュース配信者は NVML 文書を作成すると，受信者に NewsML 文書と共に配信する．受信者は NVML 文書を見ることで，今日の主要なニュースは何か，いつニュースが作成され送られてくるか，などの情報を獲得できる．

2.2　NewsML に基づく記事作成

NewsML には様々なメタデータが定義してあり，全てのメタデータは管理情報，分類情報，記事情報，記事内容情報の 4 つに分類することができる．表 2 に NewsML のメタデータの一部を示す．

人手による入力が必要なメタデータには，記事内容分類，地域情報，企業情報，ルビなどがあり，ユーザはこれらのデータを入力することに負担を感じる．したがって，ニュース編集システムには上記メタデータを自動付加する機能が必要になってくる．上記メタデータ紹介で述べた記事内容分類は NewsML では

表 2　NewsML のメタデータ

管理情報	記事 ID，運用日時，修正区分，関連リンク情報
分類情報	記事内容分類，見出し，記事種別，地域情報，企業情報
記事情報	記事種目，主ヘッダ情報，副ヘッダ情報，編注
記事内容情報	段落，ルビ，URL，修正／訂正部情報，年齢

SubjectCode と定義されており，カテゴリの総数が 1,371 種と非常に多い．本研究では，SubjectCode を自動付加するシステムを試作した．

ニュース配信者はニュースドキュメントを日々大量に作成している．このような背景のもと，より質の高いニュース記事の作成，及びより多くのニュース記事の作成には，ニュース記事自体を作成することをより効率的にする必要がある．

そこで，本研究ではニュース記事作成時の入力，および校正に対する作業時間を減らすというアプローチをとる．具体的には，ニュース記事を作成する際に，過去のニュース記事から入力中のニュース記事に対し，関連性の高い語を列挙する．そして，ユーザに対し列挙された語からの入力を促すことで，ユーザがニュース記事を作成する際の入力コストを下げるニュース作成支援システムを試作した．

2.3　NVML と NewsML の関係

NVML と NewsML で行えることは完全に分離しており，それぞれの利用目的も異なる．NewsML はニュースコンテンツのみを扱う目的で利用する．一方，NVML は NewsML 文書を管理／配信すること，及びニュースの価値付けを行うことの 2 つを目的とする．したがって，NVML にはニュースのメタ的な情報のみを含むようにしている．

また，NVML と NewsML との関係は NVML の nvLink で全て管理する．NewsML 配信は NewsML，及び NVML と共にリンクテーブルを配信することによって，リンクの整合性をとっている．詳細については第 3 章で述べる．

3.　NVML を用いた NewsML 管理／配信

3.1　NVML 文書と NewsML 文書のリンク管理

ニュース記事は頻繁に作成されるため，ニュース配信者は NewsML 文書が出来上がれば，ニュースの新鮮度を保つために，その都度受信者に NewsML 文書をプッシュ配信する必要がある．プッシュ配信にすることで，受信者側は特別な行動なしに NewsML 文書を受け取ることができる．

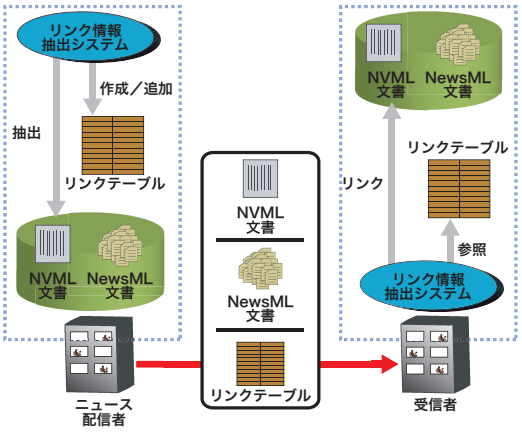


図 3　NVML 文書と NewsML 文書のリンク管理

しかし従来，リンク情報は個々の NewsML 文書に含まれるので，リンク管理コストが非常に大きい．また，ニュースは随時更新されていく特徴があるため，NewsML 文書の更新の度に更新前のリンク情報，及び関連記事へのリンク情報の整合性を保つことは複雑になる．

我々はリンク情報を一元管理するためにリンクテーブルを用いる．リンクテーブルには NVML 同士のリンク，NVML-NewsML 間のリンク，及び NewsML 同士のリンク，の 3 種類のリンク情報が含まれる．

リンクテーブルのみでリンク情報を管理することで，ニュース配信者-受信者間のリンク情報の整合性をとることが容易になる．リンクテーブルはニュース配信者側のリンク情報抽出システムによって作られる．リンク情報抽出システムは NVML の nvLink からリンク情報を抽出し，整形するシステムである．受信者側は NVML 文書，NewsML 文書，リンクテーブルの 3 つを随時受信することになる．受信者はリンク情報管理システムを用いて，リンクテーブルからリンク情報を閲覧することができる．図 3 に NVML 文書と NewsML 文書のリンク管理のイメージを示す．

受信者はユーザインタフェースを通して，NVML 文書と NewsML 文書のリンク関係を簡単に把握できる．また，NewsML 文書が更新された場合，元となる NewsML 文書とリンクしてあるので，すぐに修正作業を行うことができるといった利点もある．

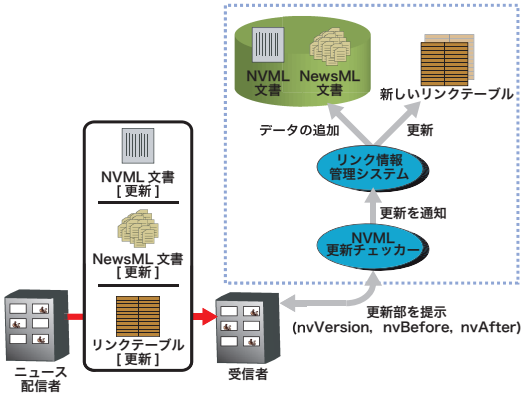


図 4　NVML 文書，及びリンクテーブルの更新過程

3.2　NVML 文書，リンクテーブルの更新

NewsML 文書は頻繁に作成，編集，及び修正が行われるため，それに伴い NVML 文書の更新も頻繁に行う必要がある．NVML 文書の更新は第 2 章で説明した nvControl の情報を用いて行う．

NVML 文書が更新される場合，NVML 文書のどこがどのように更新されたのかを一目で分かるようにする必要がある．そのために，NVML で定義した nvBefore，及び nvAfter を用いる．nvBefore，及び nvAfter にはそれぞれ修正前の情報と修正後の情報が格納されている．また，更新前の NVML 文書は nvVersion を参照することで取得できる．これらの情報を NVML 更新チェッカーが自動収集し，ユーザに NVML 文書の更新を伝える．NVML 更新チェッカーはどの部分が更新されているかをユーザの意思に対してインタラクティブに応える．

また NVML 文書が更新されると，以前受信した NewsML 文書も更新される場合もあるため，受信者側にあるリンクテーブルも更新する必要がある．その場合，NVML 更新チェッカーはリンク情報管理システムに対してリンクテーブル更新の情報を伝える．リンク情報管理システムは以前配信されたリンクテーブルを新しく配信されたリンクテーブルに置き換える．リンク情報管理システムと連携をとることで，更新時のリンクの不具合をなくす．図 4 に NVML 文書，及びリンクテーブルの更新過程を示す．

4.　ニュース編集システム

4.1　NVML 文書編集システム

従来，ニュース管理／配信は NewsML 文書を用いていたが，本研究では NVML 文書を用いるため，NewsML 編集システムとは別のシステムを用いる．

ユーザは NVML 文書に配信する NewsML 文書，及び今後配信する予定の NewsML 文書のメタデータを記述する．基本的には NewsML 文書からメタデータを抽出して NVML のメタデータに自動付加させる．ユーザの入力が必要なメタデータは，ニュースの重要度 (nvRecommendation)，ニュースの解禁日時 (nvOpenDate)，及びリンク情報 (nvLink) である．更新用の NVML 文書の場合，更新前情報 (nvBefore)，及び更新後情報 (nvAfter) の入力も必要になる．

特に，リンク情報付加は NewsML 文書が NVML 文書よりも先に作成されるか，後に作成されるかで操作が異なってくる．以下，それぞれの場合に分けて説明する．

（1）　NewsML 文書が先，NVML 文書が後

NewsML 文書が先に作成される場合，自動的にその日の NVML 文書の nvLink にリンク情報が付加される．受信者に NVML 文書を配信する際，出来上がった NewsML 文書も同時に配信される．

（2）　NVML 文書が先，NewsML 文書が後

NewsML 文書がまだ出来上がっていない場合，NVML の nvValue には NewsML 文書がいつ作成されるか，いつ配信されるか，コンテンツの行数はどのくらいかなどの情報を記述する．このとき，リンクテーブルには配信予定リンクデータが付加される．受信者には NVML 文書，及びリンクテーブルが配信される．後に NewsML 文書が作成されたら，リンクテーブルの配信予定リンクデータを実際のリンクデータに書き換える．このとき，受信者には NVML 文書，リンクテーブル，及び NewsML 文書が配信される．

4.2　継続性に基づく予測入力支援

4.2.1　ニュースの継続性

ニュースの継続性とは，ニュース記事 A があるニュース記事 B に対して話題の流れが続いている性質を示す．つまり，あるイベントに対して，ニュース記事として配信されるイベントの経緯を継続性と定義する．あるニュース記事に対して継続性が存在する場合，ニュース記事内においては，元となるニュース記事内に使用されている語が出現する確率が高いと考えられる．以上の仮説を元に，ニュース記事作成時，配信時間の新しいニュース記事において，同一の語が出現した文章から出現する名詞，及び未知語を抽出し，それらの語をユーザに提示することでニュース記事作成を支援

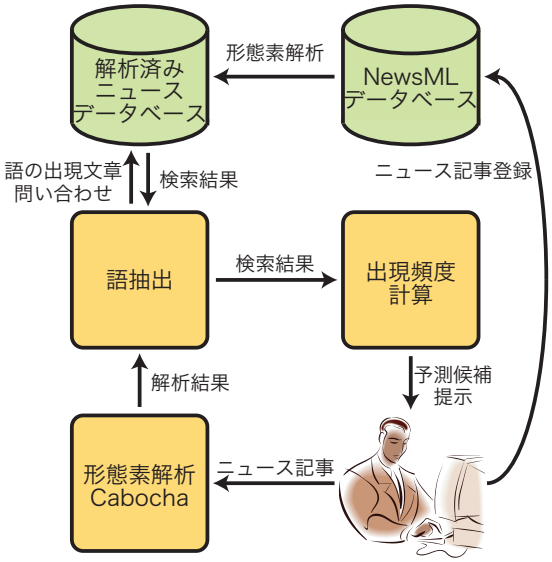


図 5　システム構成図

する．

4.2.2　システム詳細

本節では，ニュースの継続性に基づき効率的に記事を作成できるシステムについて述べる．本システムのシステム構成図は図 5 である．ユーザは本システムにおいてニュース記事を入力する．入力されたニュース記事を Cabocha(注1) によって形態素解析を行う．形態素解析の結果から語を抽出し，関連性の高いニュース記事を解析済みニュースデータベースから検索する．検索結果から，語の出現頻度を計算し，ユーザに対して関連性の高い語である予測入力候補語の提示を行う．語の抽出，及び出現頻度計算に関しては後述する．作成されたニュース記事は，NewsML フォーマットに変換されて NewsML データベースに登録される．また，作成されたニュース記事に対して形態素解析を行い，解析済みニュースデータベースに登録をする．

4.2.3　語　抽　出

抽出された固有表現，一般名詞，及び未知語をクエリワードとし，形態素解析済みニュースデータベースから抽出された語を含む NewsML 文書を抽出する．抽出された形態素解析済みの文章から出現する名詞，および未知語を集計する．この際に未知語を検索対象にするのは，最新の出来事を基に作成されるニュース記事の性質上，Cabocha 側で用いられる辞書に登録されていない単語が出現することがあり，それらが未知語として出力されるためである．また，出力された未知語が，ニュース記事において重要な語となる可能性があるからである．

4.2.4　日付指向性抽出

本システムでは，配信日時，及びニュース記事本文からニュース記事の示している日付を抽出するものとする．これは，ニュース記事において，行事の予定，及び結果を配信する場合があり，関連性の高いニュース記事は語出現頻度計算時に考慮に入れる必要があると考えられるからである．事前に日付指向性を抽出しておくことで，語出現頻度計算時に大きな影響を与えると考えられる．

4.2.5　語出現頻度計算

ニュース記事の配信時間を参考にし，語抽出の過程において集計された名詞，および未知語を新しい記事での出現頻度が高いほど影響力が高くなるように順位付けする．これは，ニュース記事が展開，および継続の性質を持っており，同一の語が出現するニュース記事に関して，配信日時がより最近のニュース記事ほどユーザが編集中のニュース記事に対して関連性が高いと考えられるためである．次に語出現頻度計算の計算式を示す．

$$F_W = \frac{\sum_{t=0}^{30} \left\{ N_{Wt}(d)^t \right\}}{M} + C \qquad (1)$$

式 (1) において，t はニュース記事が配信されてからの時間 (日)，N_{Wt} は t 日前のニュース記事に出現したある語 W の回数，d は過去のニュース記事からの影響減衰率，M は 30 日以内に配信されたニュース記事総数，C は日付指向性における補正値，F_W はある語 W が 30 日以内に出現した頻度を表す．F_W を基に，語出現頻度を順位付けする．

4.3　SubjectCode 割当支援システム

4.3.1　SubjectCode 概要

SubjectCode はニュース記事をカテゴリ毎に分類するための NewsML のメタデータである．SubjectCode は Subject（大分類，17 種），SubjectMatter（中分類，375 種），SubjectDetail（小分類，979 種）といった 3 つの階層構造から構成される．ユーザはこの中から分類コードを選ぶが，ニュース記事は 1 日に数千本配信されており，またニュースの主題は複数の SubjectCode に及ぶ可能性もあるので，一つ一つの記事に SubjectCode を付加する作業はユーザにとっ

(注1)：http://www.chasen.org/~taku/software/cabocha/

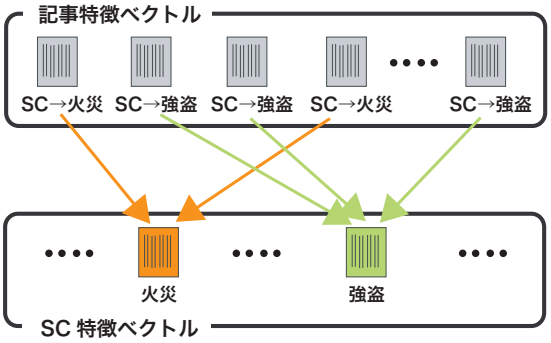


図 6　SubjectCode 特徴ベクトル作成方法

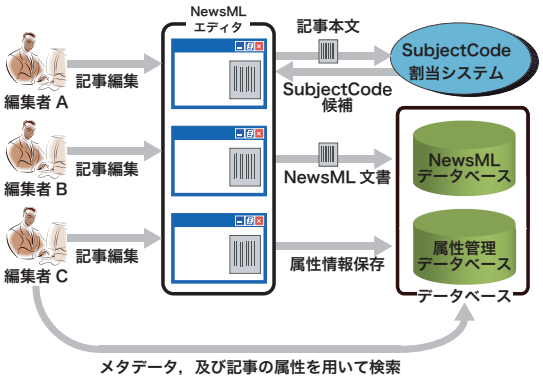


図 7　システム概要

て非常に煩雑な作業になる．また，別の観点から見てみると，SubjectCode は記事作成の面では付加しなくても問題はないが，管理上の面では付加すべきである．したがって，システム支援により NewsML 文書に SubjectCode を付加することは重要である．本節では，ユーザによる SubjectCode 付加の負担を減らすための支援システムについて述べる．

4.3.2　SubjectCode 特徴ベクトル作成方法

SubjectCode を割り当てるために，各 SubjectCode 毎に SubjectCode 特徴ベクトルを作成する．図 6 に SubjectCode 特徴ベクトルの作成方法を示す．SubjectCode 特徴ベクトルは，NewsML 文書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 文書を全て収集し，語抽出を行う．ここで，各 SubjectCode 毎の語集合を SubjectCode 特徴ベクトルと定義する．

さらに，SubjectCode 名を 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(注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．

4.3.3　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．

たとえば NewsML 文書では，「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．

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．

また，ニュースを構造的に見てみると，一般的にニュース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，見出しはニュース本文全体を要約した文章だと考えられる．つまり，ニュース本文の先頭に近ければ近いほど，そのニュース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 [4] は，実験からこの考え方が有効であるとしている．上記背景を基に本研究では，ニュースを見出しと各パラグラフに分け，見出しから順に語の出現頻度の重みを高くする．

次に SubjectCode 割当処理の流れを説明する．SubjectCode ユーザによる入力記事から抽出した語に本節で定義した重み付けを行い，入力特徴ベクトルを作成する．入力記事特徴ベクトルと SubjectCode 特徴ベクトルの類似度を計算することで，入力記事の SubjectCode を割り当てる．図 7 にシステムの概要を示す．

(注2)：http://www.newsml.jp/

5. 関連研究

文書自体にメタデータを付加して文書そのものに意味付けを行うドキュメントアノテーションの研究が盛んに行われている [6] [7]．我々は，NVML で記述された文書の中にニュースコンテンツの価値を付加している．ニュースコンテンツの価値とは，文書のカテゴリや検索モデル表現の他に，そのニュースがどれくらい重要なのかという人間の価値観をも含めたデータ集合である．

Web の世界では，Web コンテンツに意味情報を付与することにより Web の有用性を飛躍的に高めようとするセマンティック Web が注目されるようになってきている [8]．本研究をセマンティック Web の観点から眺めると，ニュースコンテンツに価値付けを行い，価値に基づいてニュース間の関係を作れる点が類似している．具体的には，NVML 文書に含まれるニュースコンテンツ価値情報とリンク情報により，意味ネットワークを構成できると考えている．ニュースコンテンツ価値情報にはニュースサマリ，ニュースカテゴリ，及びニュースプロファイル情報が含まれており，それらの情報を用いてニュースコンテンツの意味を考慮してリンクを構成することができる．

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．

6. おわりに

本稿では，NewsML の管理／配信部分を継承した記事配信メタデータ記述言語 NVML を提案した．また，ニュース編集としてニュース予測入力支援システム，及び SubjectCode 割当システムを試作した．

NVML の役割は，1)NewsML 文書とのリンク管理，及び NewsML 文書配信，2) ニュース価値情報付加，の 2 点である．

1) に関しての利点は，個々の NVML 文書，及び NewsML 文書を参照するのではなく，リンクテーブルを配信，及び参照することで，NVML 文書同士のリンク関係，NVML-NewsML 文書間のリンク関係，及び NewsML 文書同士のリンク関係を管理しやすくなる．

2) に関しての利点は 2 点ある．1 点目は，受信者は配信されてきたニュースアーカイブからその日の重要なニュースを簡単に発見できること，2 点目は，データベースに蓄積されている大量のニュースアーカイブから過去に重要になったニュースの再利用を行えることである．

今後の課題として，関連ニュースを発見しやすくするために NVML のメタデータである nvProfile の自動付加を行う必要がある．ニュース配信時に，nvProfile にデータを自動付加することで，関連したニュースを探したいと思ったときに，効率的にニュースを発見することができる．

また，nvProfile が付加されるようになれば，ニュースコンテンツの意味を考慮した意味ネットワークの構築を行う可能性がでてくる．意味ネットワークが構築されれば，大規模ニュースアーカイブから意味的階層構造をもったリンクのみを抽出し，ユーザに部分的なニュース集合を提示することもできる．

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] 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.

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

[6] 斉藤孝広，“文書記述内容のメタデータ化について ”，情報処理学会研究報告，Vol.2003，No.37，pp.51-58，2003.

[7] 伊藤一成，斎藤博昭，“汎用アノテーション記述言語 MAML”，情報処理学会研究報告，Vol.2004，No.36，pp.63-69，2004.

[8] 今村誠，伊藤山彦，増塩智宏，“Semantic Web の技術と応用の動向”，情報処理学会研究報告，Vol.2003，No.11，pp.17-24，2002.