社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－NL－179 (6)
2007－SLP－ 66 (6)
2007／5／24

情報検索と情報集約による情報取得システム

金田晃征　　野村浩郷
九州工業大学 情報工学部
〒820-8502 飯塚市川津 680-4
nomura@ai.kyutech.ac.jp
http://www.dumbo.ai.kyutech.ac.jp/nomura/

あらまし 現在の情報検索は、Web 検索の場合には URL のリストを検索結果として出力し、文書セットの場合には文書の題目を検索結果として出力するものが多い。しかし、いわゆる「情報検索」の本来の目的は、必ずしもこのような出力を得ることではなく、例えば、知りたいことを簡潔にまとめたレポートを得ることである。そこで、本研究では、与えられた要求項目の内容に関連する文書を検索し、それらを一つの文書にまとめて出力する情報取得のシステムについて述べる。このような目的のシステムとしてはいろいろなものが考えられるが、本研究では、手元にあるデータの都合により、新聞記事を対象として、関連記事を検索しそれらを一つの記事に集約するシステムについて述べる。
キーワード 情報検索、情報取得、文章要約、文書集約

Information Acquisition by Information Retrieval and Information Integration

Akiyuki Kaneda　　Hirosato Nomura
Kyushu Institute of Technology
Iizuka 820-8502 Japan
nomura@ai.kyutech.ac.jp
http://www.dumbo.ai.kyutech.ac.jp/nomura/

Abstract This article proposes a new Information Retrieval system. The system gathers articles related to the given inquiry from internet and/or storage. The system is not intended to display a list of URLs or a list of titles of articles found, but it is designed to perform a further task that integrates the important contents of them and then generates a summarized document as the result of the task. Thus, we do not use the popular and traditional technical term "Information Retrieval", instead we use a new technical term "Information Acquisition". The processing includes search of articles, evaluation of importance of each article, discovery of relationships among articles, summarization of each article, integration of articles, so on. Although the processing is carried out automatically, the system provides some manual control facilities in order to get better result.
Keyword Information retrieval, Information acquisition, Text summarization, Text integration

1. はじめに

現在の情報検索システムは、その多くが、検索結果として、例えば、UTL のリストを返してくれる。しかし、情報検索の本来の目的はこのような URL のリストを得ることではなく、例えば、知りたい情報をまとめたレポートを得ることである。本稿では、そのような「情報取得」システムについて述べる。したがって、伝統的な用語である「情報検索」ということばは使わない。その代わりに、新しい用語として「情報取得」ということばを使う。

我々は、このような考えの下に、2～3年の予備検討の後に、2002 年に研究・開発を開始した。このとき、我々の情報取得システムに CODIFY *(COrrelative Documents Intentionally Finding sYstem)* という名前をつけた。CODIFY の第一バージョンは 2003 年 3 月に作成した。その後、いくつかの改良バージョンを経て、現在の第 1.2 バージョンを 2007 年 3 月に開発した。本稿は、その第 1.2 バージョンについて述べるものである。

CODIFY は、検索して取得すべき情報に関する「要求」が与えられると、その処理を開始する。その処理は幾つかの処理からなり、最終的に一つの文書を生成する。幾つかの処理には、「要求」に関連する文書を収集すること、各文書の重要度を評価すること、文書間の関連を見つけ出すこと、各文書の要約を作成すること、文書集合の集約を作成すること、などである。

「要求」を与えるときの形としては、伝統的なキーワードによる論理式、自然言語で書かれた文章、興味を持った文書などがある。ここでは、一つの事件なり一つの出来事なりを対象として、それに関連する文書を収集し、それらの中から「要求」に対して関連度が高くかつ重要な内容を持つ文書を選別し、選別された文書の集合を集約して一つの文書としてまとめることを考える。したがって、本稿では、興味を持った一つの文書が「要求」として与えられた場合についてのべる。結果としての文書のサイズも「要求」に付随するものとして、応じる。他の形の「要求」のものについては、あらためて別の機会に述べることとする。

CODIFY は要求が与えられると自動的に処理を行うが、よりよい結果を得るために、システムにはいくつかの対話型制御機能が用意されている。これらの制御の指示は、可視化されたレーダーチャートの操作などにより、スクリーン上で行われる。

2. 関連文書の検索

2.1 ベクトル空間と類似度

文書を特徴付ける要素をベクトル空間の軸とする。各文書は、各要素の特徴量により、ベクトル空間内の一つのベクトルとして表現する。これを特徴ベクトルと呼び、j番目の特徴ベクトルを $\vec{D}j$ で表す。「要求」も同様にベクトル表現する。これを要求ベクトルと呼び、$\vec{Q}$ で表す。

文書における単語の重み付けは、よく知られた TF-IDF を用いる。これは、単語の重要性と文書の重要性を求めることを目的としている。

単語 t が文書 d に出現する頻度を tf(d,t) で表す。単語 t が出現する文書数を df(t) で表す。Df(t) については、その逆数を log と文書総数 N を使って正規化した idf を用いる。これらの値を用いて、文書 d における単語 t の重み w(t,d) を w(t,d) = tf(d,t) ・ idf(t) を使用する。

文書間の類似度は、各文書に対応するベクトル間の余弦を用いる。したがて、「要求」ベクトルと文書ベクトルの間の類似度 sim(D,Q) は、コサイン相関値を用いた類似度評価値として、 $\mathrm{sim}(D,Q) = \vec{D}\vec{Q} / |\vec{D}||\vec{Q}|$ を用いる。sim(D,Q)の値は 0 以上 1 以下であり、1に近づくほど類似度が高くなる。

検索は一度で完了させるのではなく、検索結果をフィードバックして検索を繰り返す。これにより、検索を「要求」にできるだけ近づける。すなわち、 sim(D,Q) の値をできるだけ 1 に近づける。この操作は、関連性フィードバック (Relevance Feedback) としてよく知られている。

フィードバックとして最も一般的なものは、結果の正例 (positive example) および負例 (negative example) の提示である。その他に、正例のみをフィードバックするもの、それぞれの妥当性の度合いをランクづけてフィードバックするものなど、様々なものがある。

フィードバックを検索に反映させる方法としては、大きく分けて、次の二つの方法がある: 検索質問の特徴ベクトルを修正・変換して、正例の特徴に近づけ、負例から遠ざける質問ベクトル修正法(Query Vector Movement)、および正例を検索するのに好都合な軸を強調し、負な例のものの影響を減らすように重みづけを動的に変更する再重みづけ法(Feature Re-weighting)がある。ここでは、これらの両者を採用する。

検索結果を「要求」に近づける手法として、「要求」ベクトルを、適合していると判断される記事に近づけ、不適合であると判断される記事から遠ざけるように漸進的に更新する方法を活用する。この操作を反復適用することにより、徐々によい検索結果が得られるようになる。このためによく利用されるのは次の式で示される Rocchioのフィードバック法である。すなわち、要求ベクトルQ_iをQ_{i+1}に漸進的に改善していく。

$$Q_{i+1} = Q_i + \alpha \frac{1}{R_n} \sum_{j \in R} \vec{D}_j - \beta \frac{1}{N_n} \sum_{j \in N} \vec{D}_j$$

ここで、Rは、適合であると判断された文書D_jに対する特徴ベクトルを表す。N は、不適合であると判断された

文書に対する特徴ベクトルを表す。R_nおよびN_nは、それぞれ、適合文書数および不適合文書数である。αおよびβは、それぞれ、適合文書、不適合文書に対する変数であり、αの値が高いと適合文書による変更が重要視され、βの値が高いと不適合文書による変更が重要視される。本稿では、広く応用されているこのRocchioフィードバック検索システムを活用する。

2.2 状態遷移確率を考慮に入れた重要度評価

PageRank は、ハイパーリンクされたWebページに関して、「良い多くのWebページからリンクされているWebページは、良いWebページである」、という仮説に基づき、Webページの重要度を定量的に求めるものである。リンクを持つ二つのWebページは、PageRank 値を折半する。

PageRankがページ間のリンクの重みを平等に扱っているのに対し、本稿では、各文書との類似度で重み付けを行い、文書間の類似度による文書の重要度評価とする。これによって新聞文書群を関連度の強さに応じたリンクによって結ばれた構造と考えることができるようになる。そのグラフから、PageRank の場合と同様に、遷移確率の最大固有値における固有ベクトルを計算することにより、より関連性が高いものとしてリンクされている文書を見つけ出す。

計算は、よく知られているように、次のようにして行う。すなわち、記事数を N とするとき、N×N のN次正方行列を作り、その要素に各文書間の類似度の値をそれぞれ割当てる。これを類似度行列と呼ぶ。次に、各文書、すなわち各列について合計が1になるように正規化する。これを状態遷移確率行列と呼ぶ。

実際に計算してみると、高い重みでより多くの文書からリンクされている文書ほど高いスコアを得ていることが確認できる。

このスコアは、現在見ている記事から、その記事と類似度が高い記事を優先的に選択し、類似度に応じた確率で読み進めていくという仮想的なユーザを考えるとき、無限時間後に定常状態になった時点で、どの記事に行きつくかという確率に相当する。すなわち、その記事が類似性があるとしてユーザが興味を持ち、辿り着きやすいかというスコアであり、また、記事群の中でどの記事が多くの記事から類似性を持っているとして高い重みでリンクされているか、ということを示すスコアであるともいえる。

以上によって求められたスコアを、ベクトル空間モデル上の類似度を計算したスコアに併用することにより、質問ベクトルとの類似度でユーザの興味を考慮に入れつつ、そのなかで代表らしい記事を結果として示すことができる。

これによりユーザが検索結果の判断に用いるのに適している記事を得て、効率良くフィードバック検索を行おうとするものである。

3. 集約

3.1 カテゴリ

複数の文書の集約では、文書を文カテゴリに分類し、それらの組み合わせにより、「要求」に沿った形式の文書を生成する。その際の各カテゴリは、ユーザの情報取得の選択肢を広げ、複数新聞記事を比較する際の利点に沿ったものでなければならない。

そこで、カテゴリとして、各記事に共通の箇所である重複箇所、各記事に固有の箇所である固有箇所、および記事中における補足的な内容である補足説明の合計 3 つのカテゴリを設定する。ここで、重複箇所中の文の対応の定義として、一方の文に比較対象の文の話題が完全にまたは部分的に含まれていることとする。

カテゴリ分類のための判定基準として、文単位の類似度および文タイプの 2 つを用いる。

文単位の類似度では、各文中の動詞をキーとした名詞集合中の名詞単体の概念間の距離と表記を利用して求めた値と、それらの結果を利用した名詞単語中の表記が同じ名詞の割合の合計を、結果の値としている。

文タイプでは、各文に対して文のタイプ付けを行う。そして、類似度と文タイプの結果を総合的に利用し、カテゴリ分けを行う。

3.3 複数新聞記事間における文単位の類似度

重複箇所および固有箇所の選定の一基準として、本システムでは、文単位の類似度を採用する。以下に類似度の算出方法について述べる。

一般に、文の類似度の指標には、構文的な類似度と意味的な類似度が考えられる。本システムの類似文検索では、構文構造の類似度を求めるために「動詞への係り受け」を使用する。また、意味的な類似度を求めるために「動詞に直接係る文節中の名詞の意味属性」および「名詞表記の一致の割合」を利用する。

本システムにおける類似文の検索は、次の 4 つのステップで行う。

1) 動詞を含む文節に係る文節中の名詞の検出
2) 1 で抽出した動詞をキーとする名詞集合毎の類似度の比較
3) 2 の結果を利用した名詞表記の一致の割合
4) 2 と 3 の結果を利用した類似度の算出

動詞を含む文節に係る文節中の名詞の抽出は、要約処理で使ったものを援用する [1]。

動詞を含む文節に係る文節中の名詞の概念関係を利用した比較を次のように行う。

まず、部分的な重複を文間の類似度の情報に入れるため、日本語形態素解析を参照して、各文中の動詞を含む文節に係る文節中の名詞を抽出し、各動

詞に対する名詞集合を作成する。その際の集合中の各名詞間の類似度は、表記が異なるものは EDR 電子化辞書により概念間の距離からその値を求め、表記が同一のものはその値を最大値にする。そして、各名詞単体同士の類似度から名詞集合同士の類似度を算出し、その中で最も類似度が高い値を採る。

名詞の類似度を測る方法としては、意味属性体系上での共通な親属性の位置や、両意味属性間のパスの長さから類似度を求める方法が考えられる。しかし、一般に名詞には複数の意味属性を割り当てることができるため、名詞の類似度を求めるために、その名詞がどの意味属性の名詞として使われているのかを、文脈情報などから一意に決定しなければならない。本手法においては、この多義性の問題には立ち入らずに、崔[3]らの研究で提案されている手法である名詞に割り当てられた複数の意味属性から総合的に名詞の類似度を求めるという手法を採用する。

また、動詞に係る文節中の名詞を類似度の指標として特に取り上げているのは、一文に含まれる話題の数の違いを考慮したことによる。

名詞集合間の類似度を算出するための処理は、次のようにしておこなう。

名詞同士の比較を行い、表記が同じものは類似度を最大値の 1 として算出する。それ以外の表記が異なるものがある場合には、概念辞書を利用した比較を次のように行う。

まず、名詞の概念を表す概念識別子を日本語単語辞書からとりだし、それを利用して概念辞書から意味属性のリストを得る。次に両名詞の持つ意味属性から名詞間の関係を篠原ら[4]の提案による同義関係と類似関係とに分類する。

この2つの関係に基づき、概念関係を利用した表記が異なる名詞間の類似度を求める。同義関係の類似度 a と類似関係の類似度 b はそれぞれ次式により求める。

各式については、篠原らの名詞間の概念関係の式を採用した。また篠原らは、他に同一関係という概念識別子が同一であるという関係を定義しているが、EDR 電子化辞書においてはかなり詳細に概念が定義されているので、同一関係というものは採用していない。

同義関係の類似度 a

$$a = \frac{2D_{\partial}}{A_1 + A_2}$$

A_n: 名詞nの意味属性数 (n = 1, 2)
D_a: 重複する意味属性数

類似関係の類似度 b

$$b = \frac{1}{N_1 N_2} \sum_{i,j=0}^{N_1,N_2} \frac{2D_{ij}}{N_{1i} + N_{2j}}$$

N_n: 名詞nの意味属性数
N_{ni}: 名詞nの意味属性iの上位概念数
D_{ij}: 意味属性i, jの上位概念の重複数

名詞同士の類似度S_1

$$S_1 = 1 - e^{-(a+b)}$$

以下にそれらを使用した動詞をキーとする名詞集合間の類似度の算出方法を述べる。

係り受け解析により得られた係り受け情報から、動詞が含まれる文節に係る文節の中の名詞句を動詞をキーとした組として取り出す。

名詞集合1と名詞集合3との類似度を算出する際には、記事 1 を主体と考えた場合に、名詞Aと名詞FおよびG間で類似度が高い方を名詞Aに対する類似した名詞とし、ここでは便宜上それを名詞Fと仮定する。同様に名詞Bも名詞FおよびG間で類似度が高い方を名詞Bに対する類似した名詞とし、ここでは便宜上それを名詞Gと仮定する。そして、主体側の名詞の数をn、名詞Aと名詞Fの類似度をS_{AF}、名詞Bと名詞Gの類似度をS_{BG}とした場合に、名詞集合間の類似度をS_2とすると、以下のようになる。

$$S_2 = \frac{S_{AF} + S_{BG}}{n}$$

同様に、名詞集合 2 と名詞集合 3 を比較し、集合間の類似度を求める。そこで名詞集合 1 と名詞集合 3、名詞集合 2 と名詞集合 3 の類似度をそれぞれ比較し、値が高い方を動詞をキーとする名詞集合間の類似度として採用する。ここでは便宜上名詞集合1と名詞集合3の類似度S_2を採用するものと仮定する。

前工程では名詞の概念間の距離を利用して最も類似度が高い動詞をキーとした名詞集合を各文で選んだ。ここでは、そこで選んだ名詞集合以外の文中の名詞単語中の表記が同じ名詞の割合を算出する。

表記の一致の割合によるS_3

$$S_3 = \frac{2D_{ij}}{A_i + A_j}$$

D_{ij}: 文iと文jの動詞に係る文節以外の部分の名詞の内の一致した数
A_i: 文i中の動詞に係る文節以外の部分の名詞の数
A_j: 文j中の動詞に係る文節以外の部分の名詞の数

文の類似度Sは、前述の便宜的な仮定の下で、S_2とS_3により、以下のようになる。

文の類似度 S

$$S = S_2 + S_3$$

4. 文タイプによる選定

本稿では、より文書の特色を利用した重複箇所

の選定方法として、各文に文書の特徴を考慮した文タイプを設定し、それに基づいた重複文、固有文、および補足説明の選定を行っている。この手法により、文書特有の言い回しがある場合、表現に関する制約を選定の指標として新たに採り入れることが可能となる。

4.1 文タイプの種類

従来の要約システムの研究[1][2]では、多くの文タイプを定義していた。本稿ではそれほど精密な文タイプを必要としないため、あらためて朝日新聞及び毎日新聞のそれぞれ 100 記事づつ合計 200 記事を再度分析して、少数の文タイプに定義し直した。

本稿での文タイプは、文書の特徴に基づき、従来の文タイプの中から、要旨、予定、理由、分析、および補足説明の5つを採用し、また、寺村による概言のムード[5]を取り入れ、さらに、上記データ分析から、様態・伝聞と比況・推量の 2 つを採用した。詳細な説明は省略する。

4.2 文タイプの適用優先順位

文によっては複数の文タイプを兼ねるものも多数存在する。その際に文タイプを設定する優先順位というものを考慮にいれる必要が出てくる。

そこで、文タイプの優先順位が未設定であるシステムを試作し、前述の実験データ 200 記事を使って各文に対し文タイプの出現頻度を調査した。その結果を次表に示す。

文タイプ名	出現数	出現率 (%)
要旨	199	14.26
予定	109	7.81
理由	11	0.79
分析	69	4.94
補足説明	105	7.52
様態・伝聞	116	8.31
比況・推量	3	0.21
その他	784	56.16

この内、要旨は記事全体の要約であるという性質上、各文に一文程度設定されていると考えられるので、優先順位は最上位とする。また補足説明も補足的な説明を表すという性質上優先順位を最上位とする。それ以外の文タイプ(予定、理由、分析、様態・伝聞、比況・推量)を出現数が少ないものから優先する。その結果、優先順位は以下のように設定される。

要旨 = 補足説明 > 比況・推量 >
理由 > 分析 > 予定 > 様態・伝聞

4.3 文タイプによるカテゴリ分け

重複文、固有文、補足説明のカテゴリ分けは次の処理の流れに沿っておこなう。

1) 類似度が一定の閾値以上の文を重複文候補とする。
2) 文Aに対して、文B、文C、および文Dが類似文であると認定された場合:
   2.1) 文Cと文Aが同じ文タイプの場合は、文Cは文Aの重複文とする。
   2.2) 文B、文C、文Dとも文Aと異なる文タイプの場合は、文B、文C、文Dの中で最も類似度が高い文を文Aの重複文とする。

5. 複数記事の集約

本研究では、以前作成した内容統合システム[6]を利用して、カテゴリ分類した情報を元に、複数文書の集約を作成する。

5.1 2記事間の集約

2記事間の集約は、1つめの記事の重複箇所、固有箇所の文章に、2つめの記事の固有箇所を合わせた文章を2記事の集約とする。

これにより、2記事間で重複しているものや、補足的なものを削除した集約ができる。

5.2 3記事の集約

3つの記事がある場合、2記事同士の集約を3つ作成する。3つの記事を記事A、記事B、および記事Cとする。

本稿では続報記事を分類するので、時系列順でみた初めの2記事AおよびBを集約したものを基本とする。

この集約と、記事BおよびCを集約したものを1文づつ比較し、含まれていない文を集約として追加する。

これにより、記事A、B、およびCについての内容に関する集約ができる。

記事AおよびCの集約において、Cの重複箇所と判断されたのもが、もし集約に含まれていた場合、それを削除することによって、新しい集約とする。

もちろん、この手法を応用することで、3つ以上の記事を一度に集約することもできる。

5.3 記事間の関連度を考慮した集約

上記の方法に、各記事間の関連度を考慮にいれて集約を行う。

記事間の関連度は各記事を1つのベクトルで表し、そのベクトルを比較し求める。

検索を行う際に記事間の関連度も計算されるが、続報記事であるため、1つめの記事と2つめの記事、2つめの記事と3つめの記事は繋がりがあっても、1つめの記事と3つめの記事の繋がりが弱い場合がある。

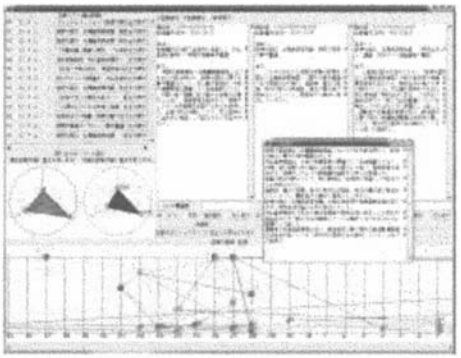
記事同士の関連度が低い場合は、記事 A および

C の比較を行わないようにすることで、無駄な処理を省くことができるようになる。

6. 情報取得システム及びその評価

6.1 システム概要

情報取得システム CODIFY は、Linux 上で Ruby を用いて作成している。CODIFY のスクリーンショットを下に示す。



6.2 スクリーン構成

左上のウインドは、最上部でしめされた「要求」文書の題名を示し、それを元に検索された関連文書が「要求」に対するスコア順に下方に順番に表示されている。各文書は、「要求」に対して適合・非適合のタグが付けられるようになっている。

その右側上部の三つのウインドは、指定された三つの文書の要約の内容を表示している。表示文書の選択および表示の指示は、上記のウインド内で行われる。

左側中央の二つのレーダーチャートは、文書検索結果からのフィードバックによる文書検索式調整のためのものである。レーダーチャートの数は二つには限られていなくて、類似度の計算式調整などのものを小さいサイズで表示したり、サイズはそのままで下方のグラフ領域に侵略して表示することもできる。

底部のグラフは、文書間の関連度・類似度を示しており、それらの強弱の程度を線の色および太さで示している。ノードは文書を示している。ノードをクリックすることにより、その文書の内容が文書内容表示ウインドに表示される。

グラフの縦軸は、「要求」に対する文書の重要度を表している。現在のシステムは、文書として新聞記事を使っているので、横軸は、日付すなわち時間軸を表している。したがって、集約は、日付にしたがって事象の移り変わりを反映することができる。

これらの色と連結線と重要度と日付により、「要求」を満たす形で続報性を持つ文書すなわち記事を読み進めていくこともできる。

右より側の中央部にオーバライトされた独立ウインドは、情報取得の出力である集約された文書を提示している。ウインドはいずれの場所にも移動させることができる。集約文書は要約処理によりサイズの調整ができる。

集約文書の背部には、検索の実行、文書表示の指示、要約量の指定、要約の実行、集約の実行、集約文書内容の文書表示ウインドへの移行指示、などの各種指示ボタンがある。要約量の指定は、元の文書のサイズに対する割合を%で指定することもでき、また、要約文の文字数で指定することもできる。

6.3 集約の評価

集約を行った結果と元の記事との比較、元記事を要約した文章との比較を行い、集約として適切であるかどうかの評価を行った。その結果、完璧とまではいかないまでも、情報取得の目的を十分達成できていることが確認できた。

7. むすび

現在の情報検索の概念を変える情報取得を提案し、情報取得システム CODIFY を作成した。このシステムは、情報検索として URL を提示するものではなく、キュアリあるいは「要求」に深い関連がありかつ重要な内容を持つ情報を検索・収集し、それらをまとめて一つのレポートに集約して提示するシステムである。

参考文献

[1] H. Nomura, H. Koga, H. Nagai, & T. Nakamura: "Text Summarization Based on Linguistic Function, Conceptual Relationships and Partial Contextual Constraints", Proc. of IPSI-2004 VENICE, CD-ROM, 6 pages (2004)

[2] S. Egami, H. Nagai, T. Nakamura, H. Nomura: "Text Summarization by Text Structure and Semantic Network", Proc. of SIGNLP, IPSJ, Vol.2004, No.108, 2004-NL-164, pp.83-88 (2004) (in Japanese)

[3] 崔 進, 小松 英二, 安原 宏: "EDR 電子化辞書を用いた単語類似度計算法", 情報処理学会研究報告 NL-93-1, pp1-6

[4] 篠原 直嗣, 増山 繁, 山本 和英: "類似文の比較による省略可能な格要素の認定", 情報処理学会研究報告, NL-139-14, pp101-108

[5] 寺村秀夫 著:''日本語のシンクタンスと意味 2", くろしお出版

[6] H. Nomura: "Information Retrieval and Integration of Relevant News Articles", Proc. of International Symposium on Machine Translation, NLP and Translation Support Systems, pp.4-9 (2004)