原著論文 Original Paper

テキスト含意認識に有効な意味類似度変換及びその獲得法

Jointly Learning Similarity Transformations for Textual Entailment

横手 健一 Ken-ichi Yokote
東京大学大学院 情報理工学系研究科
School of Information Science and Technology, The University of Tokyo
velleykt@iis.u-tokyo.ac.jp

ボレガラ ダヌシカ Danushka Bollegala
（同　上）
danushka@iba.t.u-tokyo.ac.jp, http://www.iba.t.u-tokyo.ac.jp/~danushka/

石塚 満 Mitsuru Ishizuka
（同　上）
ishizuka@miv.t.u-tokyo.ac.jp, http://www.miv.t.u-tokyo.ac.jp/

keywords: recognizing textual entailment, semantic computing, natural language processing

Summary

Predicting entailment between two given texts is an important task on which the performance of numerous NLP tasks such as question answering, text summarization, and information extraction depend.The degree to which two texts are similar has been used extensively as a key feature in much previous work in predicting entailment. However, using similarity scores directly, without proper transformations, results in suboptimal performance. Given a set of lexical similarity measures, we propose a method that jointly learns both (a) a set of non-linear transformation functions for those similarity measures and, (b) the optimal non-linear combination of those transformation functions to predict textual entailment. Our method consistently outperforms numerous baselines, reporting a micro-averaged F-score of 46.48 on the RTE-7 benchmark dataset. The proposed method is ranked 2-nd among 33 entailment systems participated in RTE-7, demonstrating its competitiveness over numerous other entailment approaches. Although our method is statistically comparable to the current state-of-the-art, we require less external knowledge resources.

1. は じ め に

自然言語においてテキスト間の含意関係の推測は，難解重要な課題である [Dagan 04]．テキスト含意とは，より具体的には，ある 2 つのテキスト対T(Text) と H(Hypothesis) が与えられた時の，T から H への推論可能性のことである．T から H を推論可能な時,「T が H を含意している」という．

例えば，以下の 2 文を考える．

(1)**T:** *All animals must eat to live.*

(2)**H:** *All wild animals must eat to live.*

この時，全ての動物が生きるために食べなくてはならないなら，その部分集合である野生動物も当然生きるために食べなくてはならないため，1 番目のテキストから 2 番目のテキストを推論可能である．つまり，(1) は (2) を含意している．

このようなテキスト含意が認識できるようになることは，質問応答や文書要約, 情報抽出を初めとした様々な自然言語処理系のタスクで潜在的に求められている．例えば，あるシステムが「alcohol reduces blood pressure」から「alcohol affects blood pressure」を推測できたとしよう．この時，その結果を用いてシステムは「What affects blood pressure?」の質問に答えることができるようになる．また，テキスト含意を判定することは，ある機械が知的かどうかを判定する有名なチューリングテストとも関係が深いと考えられており，異なる知識源を扱い推論を生成できることは知能システムを実現するための主要な要件として挙げられている [Bos 05].

2 つのテキスト間の含意関係を検知するためには，しばしばそのテキスト中には明示的に示されていない知識を必要とする [LoBue 11]．例えば (1)-(2) の場合,「wild animals」が「animals」の部分集合であるという事実は，システムが正確に含意関係を推測するためには必須の知識である．この様な知識は外部の言語資源から取得することが一般的であり，実際 NIST（National Institute of Standards and Technology）が主催するテキスト含意認識技術の評価タスク[*1]においても，多くのシステムが外部の言語資源をシステムに組み込んでいた [Bentivogil 11].

*1 http://www.nist.gov/tac/2011/RTE/

テキスト含意認識に関する既存研究が利用する知識の形態は様々であるが，その中に WordNet [Tatu 05] や FrameNet[Aharon 10]，Web[Glickman 05] の様な言語資源を用いて，T の H に対する意味類似度を計測し知識として活用しようというものがある．T と H の間に高い意味類似度が見られれば，それは T と H の間に含意関係が生じている根拠になりえるかもしれない．例えば，(1) と (2) は多くの単語を共有している．さらにオントロジーや複合語処理技術等を用いれば，「wild animals」と「animals」の類似度も計測可能である．結果として (1) と (2) の間には高い意味類似度を評価でき，この事実は (1) が (2) を含意していることを示唆しているようにも見える．

しかしながら，2 つのテキスト間の高い意味類似度は必ずしも含意の根拠とはならないことに注意しなくてはならない．

例えば，以下の 2 文を考える．

(3)**T:** *Fannie Mae's accounting has been under investigation by the Justice Department and the SEC, and it has become the subject of investor lawsuits.*

(4)**H:** *Fannie Mae is a big company.*

この場合，(3) の内容は会社の規模を知るには不十分である．したがって，この 2 文に含意関係はない．しかし，「Fannie Mae」という単語は (3) と (4) 両方に含まれており，また (4) 中の「company」は (3) 中の「department」「investor」「lawsuit」「accounting」といった単語と一定の意味類似性が見られる．その結果，例えば WordNet 内のエッジの数え上げ [Leacock 98] (類似度スコア = 0.39) や，WordNet 内の lowest common subsumer の深さ比較 [Wu 94] (類似度スコア = 0.33) といった有名な意味類似度計測手法を用いると，(3) と (4) の間には高い類似度が評価されてしまう．この例はテキスト間の高い意味類似度が必ずしも含意の根拠とはならない事を示すものであり，従って類似度スコアはそのまま含意判定のための情報として用いてはならないことが分かる．

そこで本研究では，上記のようなエラーを改善する方法として，意味類似度に基づいて含意認識のためにより適切な情報を出力するような非線形変換関数を提案する．変換関数で得たい情報は，どの程度の意味類似度が，どの程度含意に影響を与えるかである．本研究ではシグモイド関数のステップの位置と高さでこの情報を表現することを目指し，従って 2 つのパラメータを持つシグモイド関数を学習モデルとして設定する．そして，この変換関数を用いてテキスト対 (T, H) の間にある含意関係性を，ある特徴ベクトルで表現することを目指す．このベクトルは，各々の要素ごとに異なる意味類似度計測手法と変換関数を用いて計算された特徴量を持つベクトルである．本提案手法の汎用性を損なわないために，適用可能な意味類似度計測手法について特別な制限は極力設けないことを目指した．また，意味類似度だけではなく，テキスト対 (T, H) の抽出元のコーパスも背景知識として利用した．さらに既存の機械学習技術を適切に適用できるように，特徴ベクトルの次元数はテキスト T,H のサイズに関係なく，用いる意味類似度計測手法の総数のみに依存するよう工夫した．

本研究では，加えて上記の変換関数の獲得法，及び特徴ベクトル中の特徴量の最適な組み合わせ法を学習する手法も提案する．

NIST が主催するテキスト含意認識タスクである RTE-7 のデータセットを用いた実験では，変換関数を用いることで，類似度スコアを直に利用した場合と比べて性能が一貫して向上した．また，変換関数を実装した RTE システムの性能は，最終的に micro-averaged F-measure で 46.48 に達し，NIST のタスクに参加した全 33 システム中第 2 位を記録した．既存の研究と比べ本研究で用いた言語資源の数は少なかったにも関わらず，高い成績を収める結果となり，本提案手法の高い有効性を示している．

2. 提案手法の概要

2·1 システムの概観

本研究が提案するシステムの概観を図 1 に示す．このシステムは入力された T,H ペアからまずスコア行列 $\mathbf{A}$ と，重要度ベクトル $\boldsymbol{v}$ を生成し，それらを組み合わせて特徴ベクトル $\boldsymbol{\phi}(T, H)$ を作る．そして最後に $\boldsymbol{\phi}(T, H)$ を含意の有無の 2 値に分類する．各々の詳しい生成手順は次章で説明する．

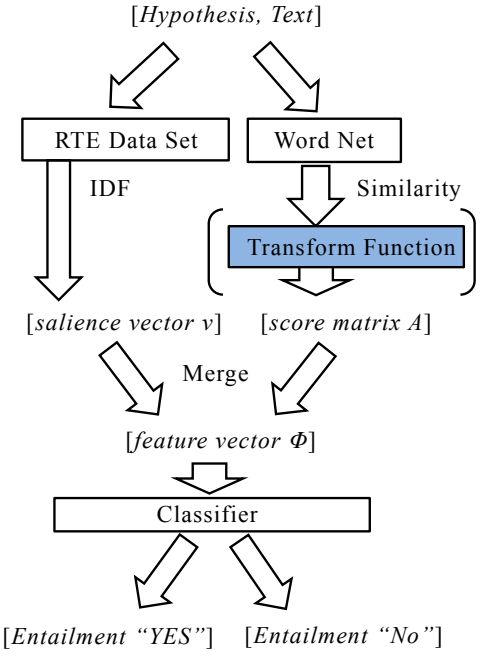

図 1 システムの概観

2·2 標準的なデータセットに基づくシステムの学習

先行研究に従い，本研究でもテキスト含意認識タスクを2つのテキスト対T,Hに対して含意の有無を識別する2値分類問題とみなし，訓練データは$\mathcal{D} = \{((T_q, H_q), y_q)\}_{q=1}^{Q}$の形で与えられることを想定する．
Qは訓練データの総数である．$y_q \in \{-1, 1\}$であり，T_qがH_qを含意していれば，$y_q = 1$，そうでなければ$y_q = -1$とする．
本提案手法では2つの学習を行う．1つは序章で説明した変換関数（図1の「Transform Function」）の内容を決定するための学習，もうひとつは，特徴ベクトル$\boldsymbol{\phi}(T, H)$を入力として，含意の有無を出力する分類器（図1の「Classifier」）の内部パラメータを決定するための学習である．後者は，分類器が出力するべき正解が訓練データ内に直接記述されているため，これを利用して教師あり学習ができる．
一方で，変換関数の学習については，正解が訓練データ内に直接記述されていない．従って，「目指すべき具体的な目標値は不明だがシステム全体の含意判定能力を高めるような最適な振る舞いを見つける問題」とみなし，強化学習の一種として扱う．

3. 特徴ベクトルの生成手順

3·1 スコア行列A

序章で述べたように，本提案手法ではTとHの含意関係性を特徴ベクトルで表現する．
そのために，まず意味類似度計測手法の集合$\mathcal{S} = \{s_1, \ldots, s_L\}$を用意する．Lは利用する意味類似度計測手法の総数を示す．
$\mathcal{S}$の要素にできる意味類似度計測手法s_iについて特別な制限はないが，値域が$[0, 1]$の範囲に収まるよう縮尺が施されていることが必要である．
次に，HとTの単語をbag of wordsモデルでそれぞれ以下のように表す．
$\mathcal{H} = \{h_1, \ldots, h_N\}$
$\mathcal{T} = \{t_1, \ldots, t_M\}$
NはHが含む単語の総数であり，MはTが含む単語の総数である．
これらを用いて，図2の様なスコア行列**A**を作る．
Aは，行数が集合$\mathcal{S}$のサイズに依存し，列数がHのサイズに依存するL行N列の行列である．それぞれの行には用いるs_iを，またそれぞれの列には，$\mathcal{T}$あるいは$\mathcal{H}$の単語が対応づけられており，i行j列目の要素を$\mathbf{A}_{(i,j)}$とすると，$\mathbf{A}_{(i,j)}$は以下のように求める．

$$\mathbf{A}_{(i,j)} = \max_{k=1,\ldots,M} s_i(h_j, t_k) \ \ \forall j = 1, \ldots, N \qquad (1)$$

この数式の意味は，以下の通りである．
まず，ある$\mathcal{H}$の単語に対して，$\mathcal{T}$の全ての単語とs_iを用いて意味類似度を計測し，その最大値を単語のスコアとする．この計算を$\mathcal{H}$の全ての単語について行う．それらの結果を，「s_iを用いて計測したHとTの類似度スコア群」として，i行目に格納する．一方の単語は，必ずしも他方の全ての単語と高く類似する必要はない．例えば含意関係にある(1)-(2)の場合，「animal」は「wild animal」と類似していることのみが分かれば，含意関係を知るための情報として十分である．従って，類似度の最大値をスコアとした．また，「Hの全ての単語がTの単語のうち少なくとも一つと類似していること」は含意関係の有無を決定する重要な要因である．従って，あるT,Hペアにおいてスコアが0となる$\mathcal{H}$の単語が存在した場合，その事実は含意関係が無いことを示す根拠とみなせそうである．しかし，s_iはオントロジーの情報の不足に基づくデータスパースネス問題や，アルゴリズムそのものの不完全性が原因で，類似性が存在するのに0と出力する場合がある．s_iのエラーによる影響を小さくするため，本研究では**A**の生成過程でスコアが0となる$\mathcal{H}$の単語が発生した場合も，処理を継続する．

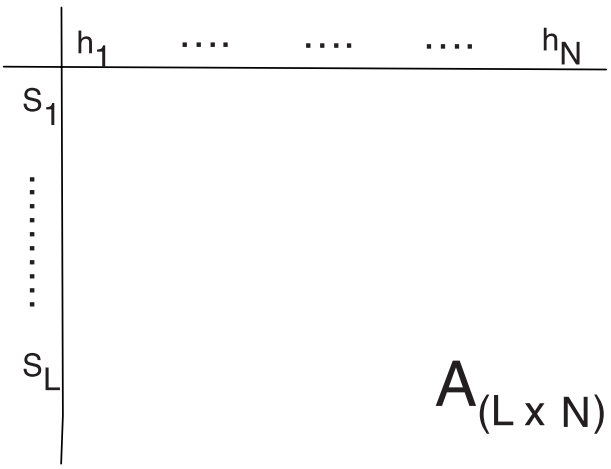

図2 意味類似度計測手法群$\mathcal{S}$を使って構築したTとHのスコア行列**A**

3·2 重要度ベクトル$\boldsymbol{v}$

前述したスコア行列**A**が持っていない情報として，テキスト中の単語間で生じる相対的な重要度の差異がある．例えば以下の例文で考える．
(5)*He teaches children about computers.*
ここで，「children」と「computers」はどちらが情報として重要であろうか．
もし「He」が教師であることが既知であれば，「teaches children」は教師の仕事なので特に重要ではなく，「computer」を教えたという事実の方が情報として重要だと言える．他方，「He」が情報技術者であった場合，「computer」について教える事は重要ではなく，「children」を対象に教えたことが重要な情報ということになるだろう．この様な単語の重要度を表現したものが，重要度ベクトル$\boldsymbol{v}$である．$\boldsymbol{v}$は対象のT,Hペアに対してN次元の大きさ

を持ち，i 番目の要素が単語 h_i $(i = 1,\ldots,N)$ の重要度を示す．テキスト間の含意関係に対して，テキスト中の各単語がどの程度影響を与えているかを判断することは背景知識を要する難しい問題とされている [LoBue 11] が，本研究では単純に逆文書頻度 (IDF)[Salton 83] を用いた．

$$\boldsymbol{v}_{(j)} = \mathrm{IDF}(h_j) \ \ \forall j = 1,\ldots,N \tag{2}$$

例えば，Steve Jobs に関する情報が文書群 D という形で与えられ，その下で以下の 2 文の含意関係を判定する問題を考える．

(6)**T:** *Steve Jobs, the CEO of Apple, teaches kids about PCs.*

(7)**H:** *Steve teaches children about computers.*

D 中に「CEO」「teach」や「PC」「computer」等を含む文書が多く存在すれば，(6) が最も主張したいことは「子供」に指導をしたという事実であり，(7) も同様に「子供」という要素に重きを置いている可能性が高い．テキスト含意認識はテキスト間の推論可能性を判定するタスクであるので，各テキストが最も主張したい内容の間の関係を知ることは重要である．従ってこの問題においては「kids」と「children」の 2 単語がどれほど類似しているかが，(6) と (7) の含意関係を決定する重要な要因になる．

そして，D 中に「CEO」「teach」や「PC」「computer」等を含む文書が多く存在すれば，IDF を計算することによってこれらの単語に低い値が割り振られ，一方で「kids」や「children」には高い値が割り振られる．従って，IDF が含意認識における単語の重要度を反映する．

例えば，RTE-7 のデータセットにおいては，テキストと共にそれらの抽出元となったコーパス及び，同様の話題を扱った文書群を topic という属性で纏められている．従って，容易に IDF が計算できる．

3·3 特徴ベクトルの生成

$\boldsymbol{v}$ を用いて，行列 **A** に重要度を反映させることができる．重要度を反映させた行列を **A'** とすると，**A'** の各要素 $\mathbf{A'}_{(i,j)}$ は以下のようになる．

$$\mathbf{A'}_{(i,j)} = \mathbf{A}_{(i,j)} v_{(j)} \tag{3}$$

$v_{(j)}$ は $\boldsymbol{v}$ の j 番目の要素を示す．ただし，**A'** は L 行 N 列の行列であり，行数は最初に決めた $\mathcal{S}$ の要素数 L で固定だが，列数は評価する T,H ペアによって変動することに注意する．**A'** が固定長の列数になり，さらに固定後の値域が一定の範囲内に収まれば，機械学習の問題はより簡単になる．また，本研究が知りたいのはテキストの間の関係性であり，個々の単語に関する情報まで把握することは必ずしも必要ない．そこで，以下の特徴ベクトル $\boldsymbol{\phi}(T,H)$ を考える．

$$[\boldsymbol{\phi}(T,H)]_{(i)} = cos(\boldsymbol{a'}_{(i)}, \boldsymbol{v}) \tag{4}$$

ここで，$\boldsymbol{a'}_{(i)}$ は **A'** の i 行目を取り出したベクトルであり，$cos(\boldsymbol{x},\boldsymbol{y}) = \frac{\boldsymbol{x}\cdot\boldsymbol{y}}{|\boldsymbol{x}|\cdot|\boldsymbol{y}|}$ である．s_i は最大値が 1 であるので，H の全ての単語が T 内に存在する時，$[\boldsymbol{\phi}(T,H)]_{(i)}$ は 1 になり，H の全ての単語が T の全ての単語と全く類似していない時，$[\boldsymbol{\phi}(T,H)]_{(i)}$ は 0 になる．H の全ての単語が T 内に存在すれば，文法構造の相違などが原因で必ずしも含意関係があるわけではないが，T が H を含意している可能性は高い．また，H の全ての単語が T の全ての単語と全く類似していない場合，類似性の判定の間違いなどが原因で必ずしも含意関係が無いわけではないが，T が H を含意している可能性は低い．従って，$\boldsymbol{\phi}(T,H)$ を生成することによって，ある T,H ペアが「含意している可能性が高い状態 (値 1)」と「含意している可能性が低い状態 (値 0)」のどちらにどれほど近いかを s_i 毎に評価でき，含意関係を判定するための根拠として利用できる．さらに，$\boldsymbol{\phi}(T,H)$ は $\mathcal{S}$ と同じ数の要素を持ち，$\mathcal{T}$ と $\mathcal{H}$ のサイズに依存しない L 次元のベクトルとなっている．本研究では，この $\boldsymbol{\phi}(T,H)$ で T と H の含意関係性を表現する．

4. 意味類似度変換と学習

4·1 類似度変換関数

序章の例文 (3)-(4) で説明したように，2 文間の高い意味類似度は必ずしも含意の根拠とはならない．そこで，類似度スコアを含意判定する上でより適切な情報（この情報を，含意スコアと定義する）へと変換するために本研究が提案するのが，意味類似度変換である．これは，類似度スコアを入力にとり含意スコアを 0 から 1 の範囲で出力する関数の形で実装し，システム内では 図 1 の「Transform Function」と記した位置で動作させるものとする．また，変換関数の内容は入力を与える s_i ごとに異なると考えられるため，$\mathcal{S}$ と同じサイズの変換関数の集合 $\mathcal{U} = \{u_1,\ldots,u_L\}$ を用意する．この変換関数の導入によって，スコア行列 **A** の要素の計算式 (1) は以下の様に修正される．

$$\mathbf{A}_{(i,j)} = \max_{k=1,\ldots,M} u_i(s_i(h_j,t_k)) \ \ \forall j = 1,\ldots,N \tag{5}$$

4·2 類似度変換の獲得法

本研究が提案する変換関数の学習手法について説明する．例えば，s_i に対応する変換関数 u_i を得たいとする．この時，まず以下の様な探査を行う．

1) 標準的なデータセットを用いて **A** を生成し，**A** の i 行目のベクトル $\boldsymbol{a}_{(i)}$ の中で特定の類似度スコアに注目し，その値を $Input_i$ とする．

2)$Input_i$ を 1 に置き換えた上でシステムを動作させる．次に，$Input_i$ を 0 に置き換えた上でシステムを動作させる．

3)2 の動作結果に基づいて，システムの性能が改善され

表 1 クラス名と，システムの判定の変化の対応表

変換値	POSITIVE	NEGATIVE	ANY
0	誤った判定	正しい判定	同一の判定
1	正しい判定	誤った判定	

たかどうかを基準に，$Input_i$ を 表 1 の 3 クラスに分類する．

表 1 における各用語の定義は以下の通りである．

【正しい判定】:

「システムの出力が含意関係あり」かつ「正解が含意関係あり」の場合か，あるいは「システムの出力が含意関係なし」かつ「正解が含意関係なし」の場合．

【誤った判定】:

「システムの出力が含意関係あり」かつ「正解が含意関係なし」の場合か，あるいは「システムの出力が含意関係なし」かつ「正解が含意関係あり」の場合．

【同一の判定】:

正しい判定か誤った判定のいずれか．

従って，クラス POSITIVE は値を 0 に置き換えた時に誤った判定，1 に置き換えた時に正しい判定が行われた $Input_i$ の集合を示し，クラス NEGATIVE は値を 1 に置き換えた時に誤った判定，0 に置き換えた時に正しい判定が行われた $Input_i$ の集合を示し，クラス ANY は $Input_i$ を 1 と 0 のどちらに置き換えてもシステムの判定が変化しなかった $Input_i$ の集合を示す．

システムの性能が改善されたという事実は，その時の類似度スコアと置き換えた値の対応関係が，変換として適切であることを示唆する根拠とみなせそうである．このヒューリスティックスに基づき，クラス POSITIVE の $Input_i$ に $Output_i = 1$ をペアとして与え，クラス NEGATIVE には $Output_i = 0$ を与え，クラス ANY は破棄する． この様にして，変換関数の入力と出力に関する訓練データ $\mathcal{D}' = \{(Input_q, Output_q)\}_{q=1}^{Q'}$ を生成し，類似度変換の獲得を教師有り学習問題に帰着させる．Q' は訓練データの総数である．ただし，$\mathcal{D}'$ は, 本来 0 から 1 の連続値をとってよい変換関数の値域 (output) を 0 と 1 に限定してしまっていることに注意しなくてはならない．本当なら 0.8 であるはずの値が 1 に設定されていたり，あるいは 0.3 であるはずの値が 0 に設定されているかもしれない．$\mathcal{D}'$ を扱う際には，この様なノイズの存在に十分配慮しなくてはならない．

まず，学習モデルは以下を用いた．

$$u_i(s_i(h,t), \boldsymbol{\alpha}_i) = \frac{\alpha_{i1}}{1 + \exp(-\lambda(s_i(h,t) - \alpha_{i2}))}. \qquad (6)$$

$\boldsymbol{\alpha}_i = (\alpha_{i1}, \alpha_{i2})^\top$

これは，ロジスティックシグモイド関数にステップの位置 α_{i1} と強度 α_{i2} を学習パラメータとして設定したものである．変数 λ はステップの鋭さを決める数値だが，最も良い学習結果が得られる定数 (=10000) とした．今回の様な離散的な訓練データに対して，鋭いステップを設定したシグモイド関数は学習モデルとして相性が良い．また，$\mathcal{D}'$ に含まれるノイズに伴う過学習の危険性を回避するため，パラメータ α_{i1} と α_{i2} は同時ではなく個別に学習することで，モデルの自由度を制限した．それぞれのパラメータは，最尤推定によって決定した．

4·3 分類器の学習法

2 章で示したように，変換関数を組み込んで生成された特徴ベクトルは，最後に含意の有無に 2 値分類される．本研究では，マージン最大化に基づく 2 値分類を行うサポートベクトルマシン [Vapnik 98] を分類器として導入した．内部パラメータの設定や，その学習手法もこれに従う．

5. 実験用データセット

本研究では,NIST が主催する Text Analysis Conference の一部門である，Seventh Recognizing Textual Entailment (RTE-7)[Bentivogil 11] タスクで用いられたデータセットで実験を行った．RTE-7 は RTE に関するワークショップの中でも最新で最大のものである．従って，このデータセットで評価すれば，RTE に関する他の最新の研究成果と比較が可能である．RTE-7 のデータセット内には，システムが各種パラメータを学習するための開発セットと，最終的にシステムの性能を評価するための評価セットがある．開発セットと評価セットは，内容は異なるが構成は同一である．開発セット全体では，284 の H があり，それぞれに最大 100 の T が割り当てられペアを作る．総計 21,420 のペアがあり，その中の 1136 が正例 (含意しているペア) である．評価セット全体では，269 の H があり，同様にそれぞれに最大 100 の T が割り当てられペアを作る．総計 22,426 のペアがあり，その中の 1308 が正例 (含意しているペア) である．また，RTE-7 のガイドラインに従うと，本データセットに基づく評価の指標は micro-averaged F-measure が指定されている．

6. 実 験 準 備

前処理として，まずストップワードの除去と NLTK[*2]のツールによる原形抽出 (word lemmatization) を，前述した RTE-7 データセット内の全てのテキストに対して行う．その上で，提案手法でシステム内の分類器及び変換関数の学習を行う．提案手法が想定している訓練データ $\mathcal{D}$ は，RTE-7 のデータセットをそのまま適用できる．また，意味類似度計測手法の集合 $\mathcal{S} = \{s_1, \ldots, s_L\}$ は，既存研究でも使用頻度の高い以下の 6 つを用いて構成する．

Path: Path では，与えられた二つの単語について，まず WordNet の知識構造内における最短パスを計算する．その後，その経路上にあるエッジの総数の逆数をもって，類

*2 http://www.nltk.org/

似度を計測する．

Wup: 2 つの単語の類似度を計測するために，[Wu 94] らは単語の知識構造上の深さ及び共有する最上の上位概念を利用する手法を考案した．Wup ではこの手法に従って類似度を計測する．

Res: 多くの情報を共有する 2 つの概念があったとき，その事実は両者が似ていることを示す根拠になるかもしれない．この理屈に基づき，[Resnik 95] は 2 つの単語間の類似度を，それらを知識構造上で下位概念に持つすべての単語の情報量の中の最大値とみなした．Res ではこの手法に従って類似度を計測する．単語の情報量は，巨大な文書内におけるそれの出現頻度から計算した．

Lin: [Lin 98] が提案する類似度定理 (Similarity Theorem) に基づいて 2 単語間の類似度を計測する．より具体的には，最も下位の共通上位概念 (lowest common subsumer) の情報量に 2 をかけたものを，2 単語の情報量の和で割った値である．

Jcn: [Jiang 97] は，WordNet の'is-a' 階層を使って情報量とエッジの数え上げの両方を組み合わせることで，2 単語間の類似度を計算する手法を提案した．Jcn ではこの手法に従って類似度を出力する．

Lch: [Leacock 98] が提案した手法を用いる．WordNet 内の'is-a' 階層上における 2 単語間のエッジ数を数え上げ，'is-a' 階層の深さの最大値で正規化する．

これらの実装には，[Pedersen 04] が提供したものを用いた．

以上の準備を行った上で，システムの性能を RTE-7 の評価セットで評価する．評価の指標は，micro-averaged F-measure を用いる．

7. 実験結果と考察

まず，本提案手法で得られた変換関数について，対応する意味類似度計測手法ごとに 図 3 に示す． ステップの位置 α_{i1} と強度 α_{i2} は変換関数ごとに異なる値になり，適切な意味類似度変換は類似度計測手法ごとに異なる事が判明した．そして，この変換関数群 $\mathcal{U}$ を使い，前述した環境の下で以下の様な 4 つの設定を設け実験を行った．

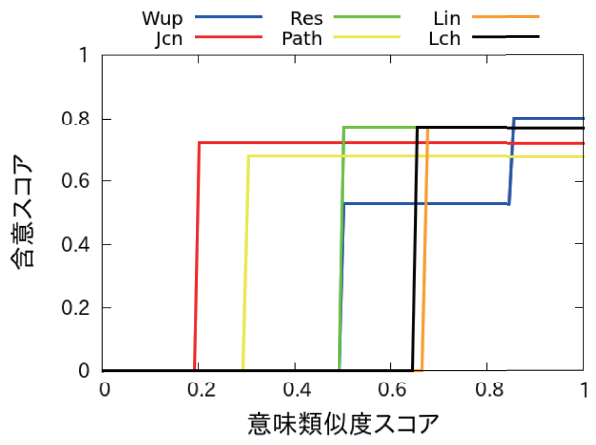

図 3 それぞれの類似度計測手法に対して提案手法を用いて学習した変換関数

§1 単一の類似度計測手法，変換関数無し

類似度計測手法の集合 $\mathcal{S}$ の内，単一の要素のみを用いてシステムを動作させる．特徴ベクトル $\phi(T,H)$ は 1 次元の値となる．この時，高度な分類器は必要なく閾値 θ を定義し含意判定は $\phi(T,H) \geq \theta$ を満たすかどうかで行う．θ の値は，開発セットで最も高い micro-averaged F-measure を出す数値とする．その θ で評価セットを評価する．また，変換関数を導入しない．

この設定はベースラインとしての役割があり，それぞれの類似度計測手法ごとの結果が．表 2 の上位 6 行である．

§2 単一の類似度計測手法，変換関数有り

設定 1 に，変換関数を導入したものが設定 2 である．表 2 の 7 行目から 12 行目がその結果である．

§3 全ての類似度計測手法，変換関数無し

集合 S の要素全てを使う． $\phi(T,H)$ は 6 次元のベクトルとなる．多次元のデータを分類するため，分類器にサポートベクトルマシンを導入する．サポートベクトルマシンが併用するカーネル関数 [Zanzotto 09a] は以下の 5 種類とする．

Linear:線形
Quadratic:二次元多項式
Cubic:三次元多項式
RBF:動径基底関数
Sigmoid:シグモイド基底関数

それぞれの結果を，表 2 の 13 行目から 17 行目に示す．

§4 全ての類似度計測手法，変換関数有り

設定 3 に，変換関数を導入したものが設定 4 である．表 2 の 18 行目から 22 行目がその結果である．

また，特徴ベクトル生成の章では $\mathcal{H}$ の単語を用いてスコア行列 $\mathbf{A}$ を構成する場合のみを示したがテキスト含意の性質に関する考察をするため，以下の 3 通りを設けた．

「$\mathcal{H}$ only」:$\mathcal{H}$ のみを用いて構成（表 2 の「$\mathcal{H}$ only」）
「$\mathcal{T}$ only」:$\mathcal{T}$ のみを用いて構成（表 2 の「$\mathcal{T}$ only」）
「$\mathcal{H}+\mathcal{T}$」:$\mathcal{H}$ と $\mathcal{T}$ 両方を用いて構成（表 2 の「$\mathcal{H}+\mathcal{T}$」）

「$\mathcal{H}$ only」が計算式 (1) に対応する．「$\mathcal{T}$ only」の計算式は，式 (1) について $\mathcal{H}$ と $\mathcal{T}$ の役割を入れ替えたものである計算式 (7) になり，「$\mathcal{H}+\mathcal{T}$」では式 (1) と式 (7) を結合した計算式 (8) になる．

$$\mathbf{A}_{(i,j)} = \max_{k=1,\ldots,N} s_i(h_k, t_j) \quad \forall j = 1,\ldots,M \tag{7}$$

$$\mathbf{A}_{(i,j)} = \tag{8}$$
$$\begin{cases} \max_{k=1,\ldots,M} s_i(h_j, t_k) & \forall j = 1,\ldots,N \\ \max_{k=1,\ldots,N} s_i(h_k, t_{j-N}) & \forall j = N+1,\ldots,N+M \end{cases}$$

重要度ベクトル $\boldsymbol{v}$ も，この修正に対応する様に要素を変更，追加する．

表 2 様々な設定の下で スコア行列 **A** のタイプ毎の性能の比較.(Micro-averaged F-scores)

設定	$\mathcal{H}$ only	$\mathcal{T}$ only	$\mathcal{H}+\mathcal{T}$
Path	32.36	12.78	16.21
Wup	21.67	11.41	12.65
Res	36.76	14.66	20.46
Lin	20.75	14.69	20.70
Jcn	**36.91**	13.85	20.82
Lch	20.75	11.28	12.18
Path+変換	44.72	26.92	35.23
Wup+変換	36.85	17.79	24.96
Res+変換	44.86	27.83	36.10
Lin+変換	**45.28**	28.18	36.41
Jcn+変換	45.06	28.17	36.41
Lch+変換	44.88	35.83	28.00
SVM (Liner)	44.96	28.79	36.91
SVM (Quadratic)	46.06	28.15	34.34
SVM (Cubic)	45.93	24.81	32.94
SVM (RBF)	**46.19**	29.16	33.76
SVM (Sigmoid)	45.42	20.84	36.15
変換 + SVM (Liner)	45.68	28.72	36.54
変換 + SVM (Quadratic)	46.19	28.23	35.27
変換 + SVM (Cubic)	45.94	25.76	33.74
変換 + SVM (RBF)	46.35	28.53	34.73
変換 + SVM (Sigmoid)	**46.48**	27.40	37.28

それぞれの設定における結果を比較すると,「$\mathcal{H}$ only」の場合が「$\mathcal{T}$ only」の場合に比べて大きく精度が上回っている.「$\mathcal{H}+\mathcal{T}$」は,「$\mathcal{T}$ only」よりやや性能が出ているが,「$\mathcal{H}$ only」ほどではない．この様な結果になった理由はいくつか考えられる. まず，2 テキスト間の含意関係性を考える上で T のみに雑音が存在するためである．例えば含意関係にあるテキスト対 (1)-(2) を考える．ここで，T にあるテキストを付加する．

(8)**T:** *All animals must eat to live. The earth has always been round.*

(9)**H:** *All wild animals must eat to live.*

この場合も，T は H を含意しており，元々の含意関係性が損なわれていない.（全ての動物が生きるために食べなくてはならず，地球は丸いなら，全ての野生動物も生きるために食べなくてはならない.）一方で，同じテキストを H に付加する．

(10)**T:** *All animals must eat to live.*

(11)**H:** *All wild animals must eat to live. The earth has always been round.*

この場合，T は H を含意していない.（地球は丸いという事実は，全ての動物が生きるために食べなくてはならないという事実からは推論できない.）従って，テキストを付与したことがテキスト間の含意関係性に影響を与えたと言える．この様に，T は H と異なり含意関係性に影響を持たないテキストが含まれることがあり，これは含意判定をする際の情報としては雑音である．本提案手法では，T の全ての単語に対して雑音かどうかを判別すること無しにスコアを与えてしまっている．このことが，T を全く用いない「$\mathcal{H}$ only」が最も性能が良かった理由である．

また,「$\mathcal{H}$ only」が最善の結果になった他の理由としては，含意認識が一方向性の特徴を持つためである．T は，H の内容全てを包含している必要があるが，逆は必要ではない．この様な性質を，双方向的にスコアを算出する「$\mathcal{H}+\mathcal{T}$」や，H に基づいて T の各単語のスコアを算出する「$\mathcal{T}$ only」では十分に反映することができなかった．一方で,「$\mathcal{H}$ only」は含意認識の一方向性を正しく捉えた方式であり，このことが「$\mathcal{H}$ only」が最も性能が良かった理由である．

以降では「$\mathcal{H}$ only」のみに注目して各設定の結果を考察する．

表の設定 1 と設定 2 を比較すると，変換関数を導入した設定 2 で大きく性能が向上している．この結果は，本研究が提案する変換関数の有効性を示している．変換関数のグラフの形から分かることは「十分に類似度があればたしかに含意に効果的といえるが，多少の類似度では含意に全く効果が無い」という知見であるが，以下にその具体例を示す.

(12)**T:** *“We heard three blasts,” he told AFP outside the church which was set ablaze.*

(13)**H:** *A Christian general in the Iraqi police force was shot dead as he drove home.*

これは，含意していないペアである．即ち，スコアとしては低い値が出なくてはならない．ところが，例えば Jcn を直に使った場合

police ：0.114（相手単語：church）
force ： 0.066（相手単語：church）
shot ： 0.076（相手単語：set）
drove ： 0.079（相手単語：set）
home ： 0.063（相手単語：church）

このように少量の数値が出てしまう．しかしながら変換関数を通すことで，全ての値が 0 に抑えられる．変換関数が有効に作用する一例である．あるいは,

(14)**T:** *The prosecutor in the case asked Miller in recent days to explain how Valerie Plame _ misspelled in those notes as “Valerie Flame” _ appeared in the same notebook the reporter used in interviewing her confidential source, Vice President Dick Cheney’s chief of staff, according to the Times.*

(15)**H:** *Dick Cheney holds the position of Vice President.*

これは含意しているペアであるが，同じく Jcn を直に使った場合

holds ： 0.000
position ： 0.271(相手単語：source)

このようにスコアとして十分な数値が出ない．ここに変換関数を通すことで，position の値が 0.720 に上昇する．これも，変換関数が有効に作用する一例である．

単一の類似度計測手法と対応する変換関数のみを用いた場合，最大の性能は Lin の 45.28 であった．Lin の類似度計測手法では最も下位の共通上位概念があるものに対して，すべてに類似度スコアがふられてしまう．しかし変換関数を用いることで，上位下位の距離が近い類似表現のみが特徴化され，含意関係に重要な情報のみを扱うように改善された．

サポートベクトルマシンを導入した設定 3 も設定 1 と比べて性能が向上しており，しかも改善の度合いは変換関数を導入した設定 2 よりも高い．最も高い性能が出たのは両者を導入した設定 4 である．例えば，以下は含意しているペアである．スコアは高く出なければならない．

(16)**T:** *At least 17 Georgian soldiers were killed this summer in clashes with South Ossetian forces, raising tensions to fever pitch in the rebel territory, which like Abkhazia declared independence following a civil war in the early 1990s.*

(17)**H:** *South Ossetia is a separatist territory of Georgia.*

ところが単語「separatist」のスコア（相手単語：rebel）をみると，

Path：0.167
Wup：0.571
Res：0.688
Lin：0.000
Jcn：0.000
Lch：0.507

であり，仮に Jcn のみに注目していた場合，スコアが 0 になってしまう．一方で Res と Wup などは大きな数値が出るため，全ての数値を考慮すれば含意判定が改善する．サポートベクトルマシンと変換関数両方を導入することで，この様なペアも適切に判定可能になる．

特にカーネル関数としてシグモイド基底関数を併用した場合に最大の性能が得られた (F-score = 46.48)．変換関数もシグモイド関数をベースに作られているため，相性が良かったのだと考えられる．

最後に，Text Analysis Conference RTE-7 に参加した他のチームと，本システムの比較を表 3 に示す．各々の数値は，RTE-7 が公式に発表している micro-averaged F-score である．13 チームが総計 33 のシステムを提出した．また，U-TOKYO は本研究の部分的な成果を実装して RTE-7 に提出した結果である．U-TOKYO では，計算式 (6) において，$\boldsymbol{\alpha}_i$ を α_{i1} のみに限定した．これは，s_i に対して何らかの閾値を設け，その値を基準に値を 1,0 の 2 値にマッピングすることに対応する．U-TOKYO よりも提案手法の方が性能が向上しており，α_{i1} と α_{i2} の 2 変数を用意して学習する変換関数が有効であることを示している．表 3 では各チームの最高値のみを掲載しているが，より詳細な情報は [Bentivogil 11] で確認可能である．本提案手法は，第 2 位の性能を記録した．例えば第 1 位の IKOMA が 3 つの言語資源（WordNet, CatVar [Habash 03], an acronym list）を利用している様に，複数の言語資源をシステムに組み込んでいるチームが多い中，U-TOKYO と提案手法では WordNet1 つのみであった．にも関わらず 2 位という高い成績を収め，本提案手法の高い有効性を示している．

表 3 micro-averaged F-score を指標にして，RTE-7 dataset で評価した各システムの性能の比較.

Method	F-score	Method	F-score
IKOMA	48.00	FBK	41.90
本提案手法	46.48	TE-IITB	30.78
U-TOKYO	45.15	JU-CSE	30.47
BUPT	44.99	ICL	29.73
CELI	44.10	UAIC	27.85
DFKI	43.41	SJTU	23.31
BIU	42.34	SINAI	14.72

なお，本研究は結果的に，含意関係の有無を決定する重要な要因である「H の全ての単語が T の単語のうち少なくとも一つと類似していること」を考慮することなく，最先端の性能を得ている.「H の全ての単語が T の単語のうち少なくとも一つと類似しているかどうか」の判定は難しい．例えば「H の全ての単語が T の単語のうち少なくとも一つと類似していれば含意関係有りとみなす」場合,「H の中に T の単語のいずれとも類似していない単語が一つでもあれば含意関係無しとみなす」ということである．この時，含意関係にある T,H ペアの H の中に類似度が計測できない単語が一つでも含まれていれば，判定を間違えてしまうことになる．例えば WordNet の情報に基づいた意味類似度計測手法であれば，H の中に固有名詞など WordNet に登録されていない単語が多数存在すれば，意味類似度が計測できず判定を間違える可能性は非常に高い．従って，少なくとも他のシステムと性能を競い合うという場面においては，無理に難しい問題を解こうとするよりむしろ解くことを回避した本研究の手法が戦略として有効だったと考えられる．

また，本研究で開発したシステムはソースコード[*3]を公開している．

8. 関 連 研 究

テキスト含意認識は，質問応答や情報検索，文章要約を初めとする様々な自然言語処理の分野から，その必要性が注目されている．結果としてテキスト含意認識の研究に関連するワークショップが数多く現れた [Callison-Burch 09, Sekine 09]．The Recognizing Textual Entailment (RTE)

*3 http://dl.dropbox.com/u/7174664/joint.zip

challenges [Bentivogil 11, Dagan 04] はその中でも特に有名なワークショップである．以下，近年の RTE への主要なアプローチについて述べる．なお，既存研究に関するより詳細な調査については [Androutsopoulos 10] らが報告している．

仮に T と H が一定量の単語を共有していた場合，その事実は T が H を含意していること示す有力な情報だとみなす考え方がある．この直観的な理論に基づき，T と H の表層的な単語の共有度を比較するようなシステムが提案された [Burchardt 09, Malakasiotis 07]．これらの手法は，T と H の間にある単語の対応関係をいかに正確に見つけられるかが，精度に大きく影響する．統計的機械翻訳の分野で，テキスト間の単語の対応関係を正確に見つける手法が存在するにも関わらず，これらはテキスト含意認識の分野においてはそれほど効果的に作用しなかった．その理由としては，T と H の長さがしばしば大きく異なること，T と H が必ずしも同じ情報を保持していない事，あるいは，テキスト含意認識のデータセットのサイズが，統計的機械翻訳を行うには十分ではないことなどが考えられている [MacCartney 08].

単語の表層的な被覆度ではなく，係り受け木で比較する手法も提案された [Iftene 07, Zanzotto 09b, Wang 07a, Wang 07b]．例えば，H の構文木が，T の構文木の一部と非常に似ていたとする．この事実は，T が H を含意する根拠かもしれない．係り受け木であれば，テキスト中における単語間の文法的関係まで情報として持つことができるので，先に紹介した純粋な単語の被覆度の比較より精度が出そうに見える．しかしながら，構文解析の過程で無視できないレベルのエラーが生じる場合があり，実際の所は必ずしも上回る訳では無い．

[Bos 05] は，テキスト含意認識を論理的推論問題としてモデル化した．彼らは，自動推論の分野でよく使われるようなモデル構築を行い，既存の表層的単語被覆度を計測する手法にそれを導入した．この手法は，RTE テストセットにおいて高い適合率を記録したものの，十分な背景知識を取り込むことが出来なかったため，再現率が上がらなかった．一方で，通常のコーパスから必要な背景知識を抽出し，含意ルールを学習しようという研究もあり，この様な問題に対処できる可能性が見込まれている [Haghighi 05, Berant 10]．また，T と H を bags-of-words で表現し，概念辞書 WordNet に基づくものを初めとした様々な意味類似度を適用した研究がある [Zanzotto 06, Geffet 04, Mirkin 06, Geffet 05]． 上位語，下位語，類義語の様な意味的な広がりを考慮することで，曖昧なマッチングが可能である．

本研究の提案手法はこれらのいずれとも異なり，特徴量の適切な変換，及びそれらの適切な組み合わせを同時に行ってテキスト対の含意関係を推測するものである．そして．既に RTE-7 のデータセットを用いた実験で示したように，この手法は最先端の研究に十分匹敵する性能を発揮した．

9. 結　　論

本研究では，テキスト含意認識のための意味類似度変換とその獲得法，及び変換結果を用いて含意判定を行う方法について提案した．そして，提案手法が様々な意味類似度計測手法に対して有効であること，最先端の研究に十分匹敵する性能を引き出せることを実験によって確認した．

ただし，本提案手法では，テキストの構文的な情報をほとんど取り扱っていない．また，向きに関する情報が欠落しており，例えば T と H という区別が無い二つのテキストに対してスコアを算出した場合，どちらが T でどちらが H かを識別することはできない．これらを解決することで，本提案手法の有用性と性能をさらに高める可能性は十分にある．

謝　　辞

本研究に際し，NEC 情報・ナレッジ研究所の石川開氏，土田正明氏，福島俊一氏の協力を得たことを記し，感謝します．

◇ 参 考 文 献 ◇

[Aharon 10] Aharon, R. B., Szpektor, I., and Dagan, I.: Generating Entailment Rules from FrameNet, in *ACL'10*, pp. 241 – 246 (2010)

[Androutsopoulos 10] Androutsopoulos, I. and Malakasiotis, P.: A Survey of Paraphrasing and Textual Entailment Methods, *Journal of Artificial Intelligence Research*, Vol. 38, pp. 135 – 187 (2010)

[Bentivogil 11] Bentivogil, L., Clark, P., Dagan, I., Dang, H. T., and Giampiccolo, D.: The Seventh PASCAL Recognizing Textual Entailment Challenge, in *RTE-7* (2011)

[Berant 10] Berant, J., Dagan, I., and Goldberger, J.: Global Learning of Focused Entailment Graphs, in *ACL'10*, pp. 1220 – 1229 (2010)

[Bos 05] Bos, J. and Markert, K.: Recognizing Textual Entailment with Logical Inference, in *EMNLP'05*, pp. 628 – 635 (2005)

[Burchardt 09] Burchardt, A., Pennacchiotti, M., Thater, S., and Pinkel, M.: Assessing the impact of frame semantics on textual entailment, *Natural Language Engineering*, Vol. 15, No. 4, pp. 527 – 550 (2009)

[Callison-Burch 09] Callison-Burch, C., Dagan, I., Manning, C. D., Pennacchiotti, M., and Zanzotto, F. M.: TextInfer'09, in *ACL-IJCNLP'09 Workshop on Applied Textual Inference* (2009)

[Dagan 04] Dagan, I. and Glickman, O.: Probabilistic Textual Entailment: Generic Applied Modeling of Language Variability, in *PASCAL Workshop on Text Understanding and Mining* (2004)

[Geffet 04] Geffet, M. and Dagan, I.: Feature Vector Quality and Distributional Similarity, in *COLING'04*, pp. 247 – 253 (2004)

[Geffet 05] Geffet, M. and Dagan, I.: The Distributional Inclusion Hypothesis and Lexical Entailment, in *ACL'05*, pp. 107 – 114 (2005)

[Glickman 05] Glickman, O., Dagan, I., and Koppel, M.: A Probabilistic Lexical Approach to Textual Entailment, in *IJCAI'05*, pp. 1682 – 1683 (2005)

[Habash 03] Habash, N. and Dorr, B.: A Categorical Variation Database for English, in *HLT-NAACL'03*, pp. 17 – 23 (2003)

[Haghighi 05] Haghighi, A., Ng, A. Y., and Manning, C. D.: Robust Textual Inference via Graph Matching, in *HLT/EMNLP'05*, pp. 387 – 394 (2005)

[Iftene 07] Iftene, A. and Balahur-Dobrescu, A.: Hypothesis Trans-

formation and Semantic Variability Rules used in Recognizing Textual Entailment, in *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing* (2007)

[Jiang 97] Jiang, J. J. and Conrath, D. W.: Semantic similarity based on corpus statistics and lexical taxonomy, in *10th Intl. Conf. Research on Computational Linguistics (ROCLING)* (1997)

[Leacock 98] Leacock, C. and Chodorow, M.: Combining local context and WordNet similarity for word sense disambiguation, *WordNet: An Electronic Lexical Database*, Vol. 49, pp. 265–283 (1998)

[Lin 98] Lin, D.: An Information-Theoretic Definition of Similarity, in *ICML'98*, pp. 296 – 304 (1998)

[LoBue 11] LoBue, P. and Yates, A.: Types of Common-Sense Knowledge Needed for Recognizing Textual Entailment, in *ACL'11*, pp. 329 – 334 (2011)

[MacCartney 08] MacCartney, B., Galley, M., and Manning, C. D.: A Phrase-Based Alignment Model for Natural Language Inference, in *EMNLP'08*, pp. 802 – 811 (2008)

[Malakasiotis 07] Malakasiotis, P. and Androutsopoulos, I.: Learning Textual Entailment using SVMs and String Similarity Measures, in *ACL'07 Workshop on Textual Entailment and Paraphrasing*, pp. 42 – 47 (2007)

[Mirkin 06] Mirkin, S., Dagan, I., and Geffet, M.: Integrating Pattern-based and Distributional Similarity Methods for Lexical Entailment Acqusition, in *COLING/ACL'06*, pp. 579 – 586 (2006)

[Pedersen 04] Pedersen, T.: WordNet::Similarity - Measuring The Relatedness of Concepts, in *HLT-NAACL'04 (Demos)*, pp. 267 – 270 (2004)

[Resnik 95] Resnik, P.: Using Information Content to Evaluate Semantic Similarity in a Taxonomy, in *IJCAI'95* (1995)

[Salton 83] Salton, G. and Buckley, C.: *Introduction to Modern Information Retreival*, McGraw-Hill Book Company (1983)

[Sekine 09] Sekine, S., Inui, K., Dagan, I., Dolan, B., Giampiccolo, D., and Magnini, B.: Workshop Proceedings, in *ACL-PASCAL Workshop on Textual Entailment and Paraphrasing* (2009)

[Tatu 05] Tatu, M. and Moldovan, D.: A Semantic Approach to Recognizing Textual Entailment, in *HLT/EMNLP'05*, pp. 371 – 378 (2005)

[Vapnik 98] Vapnik, V.: *Statistical Learning Theory*, Wiley, Chichester, GB (1998)

[Wang 07a] Wang, R. and Neumann, G.: Recognizing Textual Entailment using a Subsequence Kernel Method, in *AAAI'07*, pp. 937 – 945 (2007)

[Wang 07b] Wang, R. and Neumann, G.: Recognizing Textual Entailment Using Sentence Similarity based on Dependency Tree Skeletons, in *ACL'07 Workshop on Textual Entailment and Paraphrasing*, pp. 36 – 41 (2007)

[Wu 94] Wu, Z. and Palmer, M.: Verb Semantics and Lexical Selection, in *ACL'94*, pp. 133 – 138 (1994)

[Zanzotto 06] Zanzotto, F. M. and Moschitti, A.: Automatic learning of textual entailments with cross-pair similarities, in *ACL 2006*, pp. 401–408 (2006)

[Zanzotto 09a] Zanzotto, F. M. and Dell'Arciprete, L.: Efficient kernels for sentence pair classification, in *EMNLP'09*, pp. 91 – 100 (2009)

[Zanzotto 09b] Zanzotto, F. M., Pennacchiotti, M., and Moschitti, A.: A Machine Learning approach to Textual Entailment Recognition, *Natural Language Engineering*, Vol. 15, No. 4, pp. 551 – 582 (2009)

〔担当委員：平 博順〕

2012 年 6 月 24 日 受理

著 者 紹 介

横手 健一

2012 年東京大学工学部電子情報工学科卒．現在 同大学院情報理工学系研究科修士課程在学中.

著 者 紹 介

ボレガラ ダヌシカ

2005 年東京大学工学部電子情報工学科卒．2007 年同大学院情報理工学系研究科修士課程修了．2009 年同研究科博士課程修了．博士 (情報理工学)．現在: 同研究科・講師．自然言語処理に興味を持つ．WWW, ACL, ECAI などの会議を中心に研究成果を発表.

著 者 紹 介

石塚 満（正会員）

1971 年東京大学工学部卒,1976 年同大学院工学系研究科博士課程修了. 工学博士. 同年 NTT 入社, 横須賀研究所勤務.1978 年東京大学生産技術研究所・助教授 (1980-81 年 Perdue 大学客員准教授).1992 年同大学工学部電子情報工学科・教授. 現在: 同大学院情報理工学系研究科・教授. 研究分野は人工知能,Web インテリジェンス, 意味計算, 生命的エージェントによるマルチモーダルメディア.IEEE, AAAI, 電子情報通信学会, 情報処理学会等の会員, 本会元会長.