

自然言語を構造化表現する論理データベースの構築について

鈴木 秀明[†]

[†] 情報通信研究機構 脳情報通信融合研究センター 〒 651-2492 神戸市西区岩岡町岩岡 588-2

E-mail: hsuzuki@nict.go.jp

Abstract セマンティック Web で標準で用いられつつある RDF、Google の Knowledge Graph を引き合いにだすまでもなく、関連のある情報同士を予めリンクで結び、関連情報への高速なアクセスを可能にするグラフデータベースの手法は、ビッグデータに基づいた様々なサービスを高速に提供しようとする時、一つの必須な技術である。本論文では、人間思考の最も深い‘意味’の一つである『論理』を対象とし、それをグラフデータベースで表わす一つの方法論を検討する。知識表現形式としてペトリネット型知識推移ネットワーク (Petri-net-based Knowledge Transitive Network, pKTN) を取り上げ、それを一階述語論理相当のホーン節論理プログラミング、および自然言語文 (英文) から構築する手法について述べる。とりわけ後者の構築方法は、我々が自然言語文に基づき、深い意味に立ち入った検索や推論を行なおうとする時、一つの有効なアプローチとなりえる。

Keywords ペトリネット、pKTN、一階述語論理、自然言語、グラフデータベース

1. はじめに

知識ベースに基づいて検索や推論を行なおうとする時、データベース・サイズが巨大化するほど、その中の繋がり関係を探し出すのは容易ではなくなる。これに対処するために有効なのが、知識をリスト形式ではなく、関係した情報を予めリンクで繋いだ構造化形式 (ネットワーク形式) で蓄えるという方法である。近年セマンティック Web の分野で標準として用いられている RDF (Resource Description Framework) を用いたアプローチでは、文の深い意味には立ち入らず、“SVO トリプル”で表わされる基本的な意味に的を絞って構造化データベースを構築し、高速推論を行なおうとしている。また RDF の拡張である OWL (Web Ontology Language) やその理論的基盤を与えるとされる記述論理 (Description Logic, DL) [1], [3], [8], [9] は、一階述語論理 (first-order predicate logic, FOL) の中の一部、即ち 3 個以上の引数を持つ述語や関数記号は含まない論理に記述対象を限定する一方で、豊富な記号ライブラリを用意することによってよりユーザ・フレンドリ性を高めている。そういった意味で、今日のウェブ・データベース構築のアプローチは、論理表現能力はある程度犠牲にしつつ、シンプルで分かりやすいやり方で構造化 (グラフ化) されたデータベースを記述する方法を取ってきたと言える。

しかしながら上記のことは、我々がデータベースに深い意味を持ち込むことをやめてしまったことを意味するわけではない。実際これまで、多くの研究者が、論理のような深い意味を持つデータベースの構築を模索しており、それらの中から、我々はここで特に『演繹データベース』[11], [14], [16] と、それを記述する言語である『Datalog』[10] に注目し議論する。表 1 を見よ。この表に示されているように、Datalog の論理表現能力は、DL のそれよりは高いが、Prolog [6], [12], [22] のそれよりは低い。Datalog はデータベース言語に必要な『宣言性』という性質を満たしており、演繹データベースを記述する言語として広

く使われている。ところで一方、Datalog の一つの欠点と考えられるのが問合せへの処理 (推論) である。通常 Datalog の推論には、論理プログラミングの演繹アルゴリズムを適用し、ボトムアップ法 (前向き推論)、トップダウン法 (後向き推論)、あるいはそれらを組み合わせたマジックセット変形 [4], [7], [35] 等の手法が用いられる。これらには、予め生成される外延データベース (事実) の数が巨大になる (前向き)、処理の分散化が難しく並列実装時のスケーリング特性が良くない (後向き) といった欠点があり、ビッグデータ解析のための一つの課題として、今日まで精力的に研究が続けられている [15], [17], [34]。もし我々が、分かりやすいやり方で論理をグラフ構造化表現できたならば、深い意味を扱いつつ並列分散処理が行なえるデータベース記述のための有力なツールとなる可能性がある。

このような背景のもと本論文では、最近鈴木らによって提案された論理のグラフ表現形式であるペトリネット型知識推移ネットワーク (Petri-net-based Knowledge Transitive Networks, pKTN) [23], [26], [28] を取り上げ、それを構築するための 2 つの方法論について述べる。まず第 1 の方法は、pKTN を論理プログラムから作る方法である [28]。周知のように論理プログラムは、事実節 (外延データベース)、ゴール節 (問合せ)、ルール節 (内包データベース) の 3 種類の節を併置し、ゴール節の投げかける質問に答えるプログラムである。これら 3 種類の節を帰納的にネットワーク変換するための一般的な規則を用意することにより、任意の論理プログラムを pKTN へ変換することが可能である。第 2 の方法は、pKTN を自然言語から作る方法である [31], [32]。事実節に相当する平叙文、ゴール節に相当する疑問文、およびルール節に相当する仮定節または関係代名詞節を含む文の 3 種類の英文について、依存構造から pKTN を導く例を示す。この方法によって得られる pKTN は、従来の自然言語文を組合せ範疇文法 (CCG) [5], [21] とモンタギュー文法 [13] によって解析してから入式 (論理式) を介して概念グラフ (CG) [2], [19], [20] を導く手法に較べて、抽出され

Table 1 知識表現方法の比較

	関数記号	3 個以上の引数を持つ述語	宣言性	グラフ構造化
DL	× (許さない)	× (許さない)	○ (節順重要でない)	○
Datalog	× (許さない)	○ (許す)	○ (節順重要でない)	×
Prolog	○ (許す)	○ (許す)	× (節順重要)	×
pKTN	○ (許す)	○ (許す)	○ (節順重要でない)	○

DL (記述論理)、Datalog (演繹データベース言語)、Prolog (論理プログラミング言語)、pKTN についての比較表。第 2~4 列は主にデータベース言語としての表現能力に関する評価。また最後の第 5 列はマイニング/推論の効率に関わる評価である。

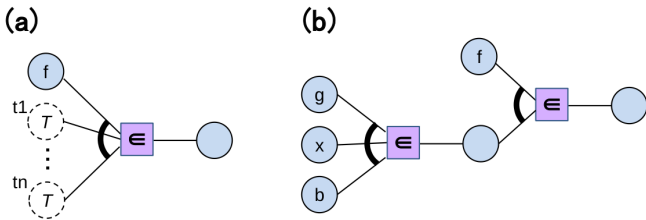


Figure 1 項から pKTN への変換規則。(a)、(b) はそれぞれ、 $f(t1, \dots, tn)$ および $f(g(x, b))$ に対応するサブネット。図中、円/楕円はプレース、四角はトランジションを表わす。太い弧でまとめられた一組のアークは論理 AND の関係を持つ。

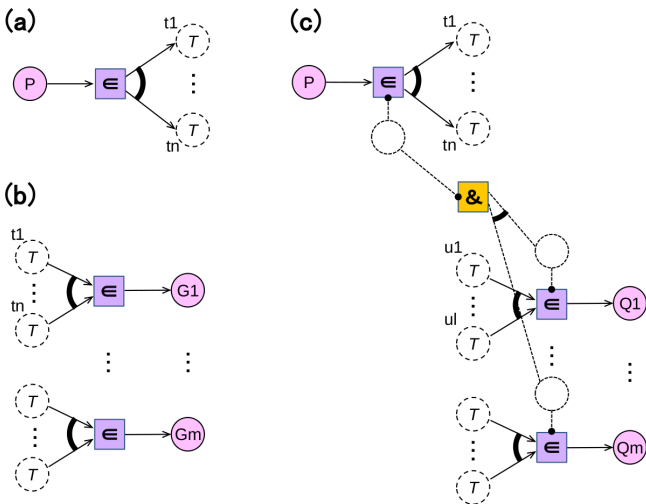


Figure 2 節から pKTN への変換規則。(a)、(b)、(c) はそれぞれ、事実節 ' $P(t1, \dots, tn)$.'、ゴール節 ' $\leftarrow G1(t1, \dots, tn), \dots, Gm(\dots)$.'、ルール節 ' $P(t1, \dots, tn) \leftarrow Q1(u1, \dots, ul), \dots, Qm(\dots)$.' に対応するサブネットを表わす。

るグラフが簡単、かつ意味の階層構造（節構造）をグラフに入れ子にすることなく表現できるという特徴を有している。得られた pKTN 上では、エリーゼ（ELiminating Inconsistency by SElection, ELISE）と呼ばれるトークンの並列伝搬を利用した単一化アルゴリズムにより演繹推論が行なえることが示唆されている [33]。

以下本論文では、まず第 2. 節で pKTN の構成を簡潔に述べた後、第 3. 節でホーン節で表わされる論理プログラムを pKTN に変換する方法を、第 4. 節では自然言語を依存構造を介して pKTN に変換する方法を述べる。第 5. 節に結論を与える。

2. pKTN のネットワーク構成

pKTN の構成を以下にまとめる。

【全体構成】 ネットワークはトランジションとプレイスから

成る2部グラフである。それらを繋ぐアークは有向で、(ループを持つ以外は)根から葉へと一方向に流れる向きを持つ。

【ブレース】 ブレースには大きく述語ブレースと項ブレースがある。項ブレースには定数ブレース、変数ブレース、またはそれらを組み合わせた項ブレースが含まれ、またこれとは別に項を作るための関数ブレースも用意される。

【トランジション】 トランジションとして、分解/合成 (Split/Combine; $\in$) と論理積 (Logical AND; $\&$) の2種類が用意される。 $\in$ トランジションは、項をまとめて述語の引数ベクトルを作ったり、関数記号と項を組み合わせて新しい項を作る働きを持つ。

3. 論理プログラムから pKTN へ

一階述語論理、特にホーン節の集合で表わされる任意の論理

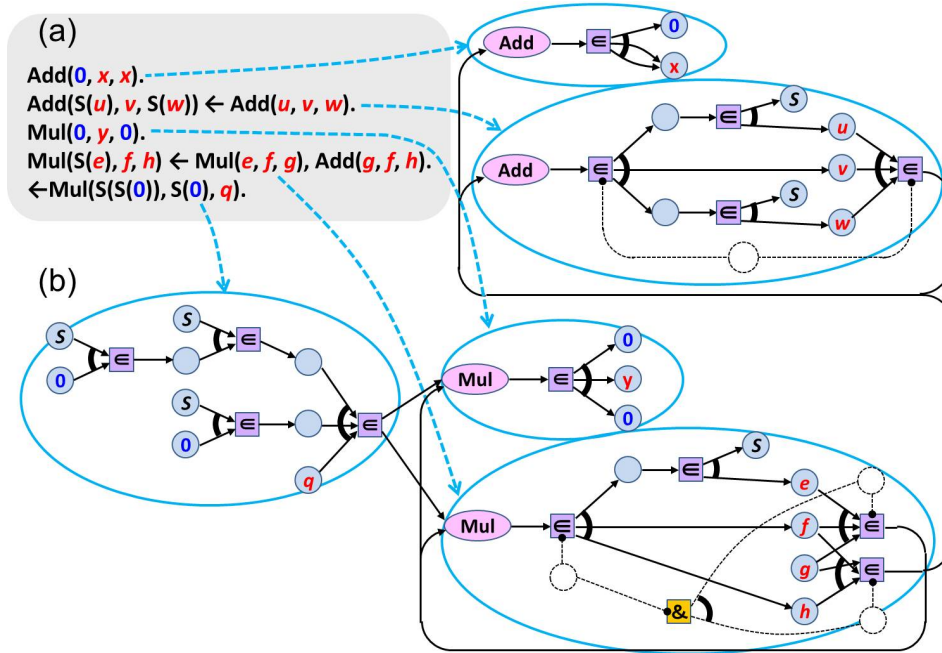


Figure 3 (a) 自然数の四則演算についての論理プログラム、および (b) その pKTN 表現。(a) で、 $'0'$ は定数、小文字のアルファベット ($'x', 'u', \dots$) は変数、特に $'q'$ はクエリ変数、 $'S'$ は関数記号 (引数の次の自然数を作る後者関数 (successor function))、 $'Add'$ と $'Mul'$ はそれぞれ加算・乗算を表わす述語記号である。

プログラムが、pKTN へと変換できる [28]。pKTN 上では述語論理中の定数、変数、関数記号は、いずれもプレースラベルとして表わされ、その意味で pKTN は一階述語論理プログラムのグラフによる“直接的”表現になっている。Figure 1 には項の、また Figure 2 には節の変換規則を示す。

これらの図で、点線で描かれた T プレースは一時的なもので、変換中別に作られたサブネットによって置き換えられる。例えば、Figure 1(b) を作るには、まずパラメータ t に対応するプレース T を使って $f(t)$ に対応するサブネットが作られ、続いて $g(x, b)$ に対応するサブネットが作られる。最後にプレース T をサブネット $g(x, b)$ で置き換える。項が持つ任意の入れ子構造がこのようにしてネットワーク表現できる。Figure 1 のアークは全て無向であるが、それらが Figure 2 にある一時プレース T と置き換えられる時、適切な向きが与えられる。

大まかに言って、事実節は述語プレースから項プレースに流れるサブネット、ゴール節は項プレースから述語プレースに流れるサブネット、ルール節は頭部述語プレースから本体述語プレースに流れるサブネットに変換される。ルール節を表わす Figure 2(c) は、事実節を表わす Figure 2(a) とゴール節を表わす Figure 2(b) を組み合わせて作られることに注意せよ。

Figure 3 に四則演算の論理プログラムとその pKTN 表現の例を示す。太い弧でまとめられた一組のアークは論理 AND の関係を持ち、それらがトランジションに入る／から出る順番は重要である。一方、太い弧が掛かっていない複数のアークは論理 OR の関係を持ち、それらがトランジションに入る／から出る順番には意味がない。また太い弧は描かれていないが、プレースに入る、またはプレースから出る複数のアーク (例: Figure 3(b)

のプレース x に入るアーク、プレース f, g から出るアーク) は論理 AND の関係を持つ。それらのプレースにとっての順番は重要ではない。

ホーン節を pKTN に変換する際、述語を表わすプレース (例: Figure 3 の Add と Mul) は、それらが節の頭部に現われた時にのみ新しく生成される。節の本体に現われた述語については新しい述語プレースは生成されず、代わりに同名の述語プレース全部にトランジションからアークが張られる。こうして論理プログラムを構成する複数の節に対応したサブネットは一繋がりになり、それと同時に pKTN はループを持つようになる。

4. 自然言語から pKTN へ

一部の自然言語文 (英文) は、依存構造 (動詞述語を頂点とする木構造) を介して pKTN へと変換できる [31], [32]。Figure 4~6 に変換例を示す。依存構造図には自然文に含まれる全ての単語や句が漏れなく含まれており、その間の文法的／意味的係り受け関係が、実線および点線で示されている。このような係り受け関係から、我々は品詞／句の変換ルール

- ・ 不定冠詞 + 名詞 $\Rightarrow$ 変数 + 述語
- ・ 定冠詞 + 名詞 $\Rightarrow$ 述語 or 関数記号
- ・ 代名詞 $\Rightarrow$ 変数
- ・ 固有名詞 $\Rightarrow$ 定数
- ・ be 動詞 $\Rightarrow$ (削除)
- ・ 動詞 $\Rightarrow$ 述語

に基づき、pKTN を構築する。Figure 4~6 に典型的な例を示す。

Figure 4 の (a) と (b) は、同じ内容を含む異なった平叙文の例である。(a) では be 動詞 $'is'$ が省略されて定冠詞 + 名詞 $'the$

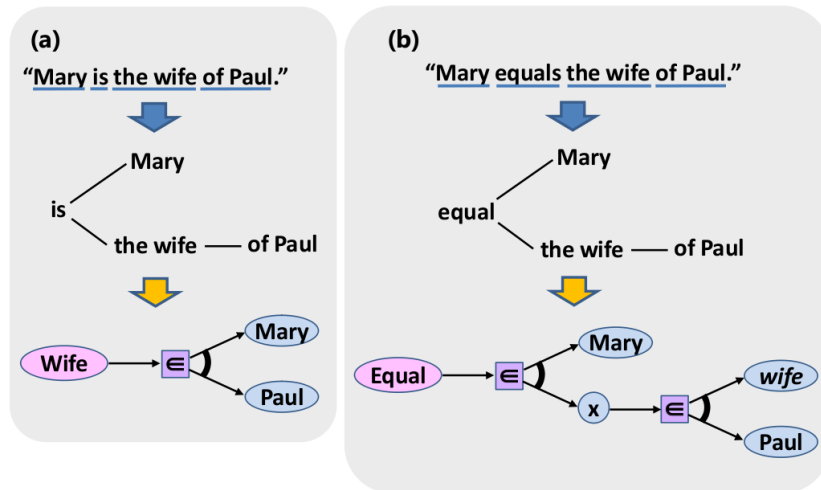


Figure 4 自然英語文（平叙文）から pKTN への変換の例。これ以下の図では、自然英語文、依存構造グラフ、pKTN が上から順番に示されている。依存構造グラフ中の実線は単語や句の間の直接の係り受け関係を表わし、点線は代名詞や関係代名詞によるリンクを表わす。pKTN に習い、依存構造グラフでも動詞述語を左に配置している。

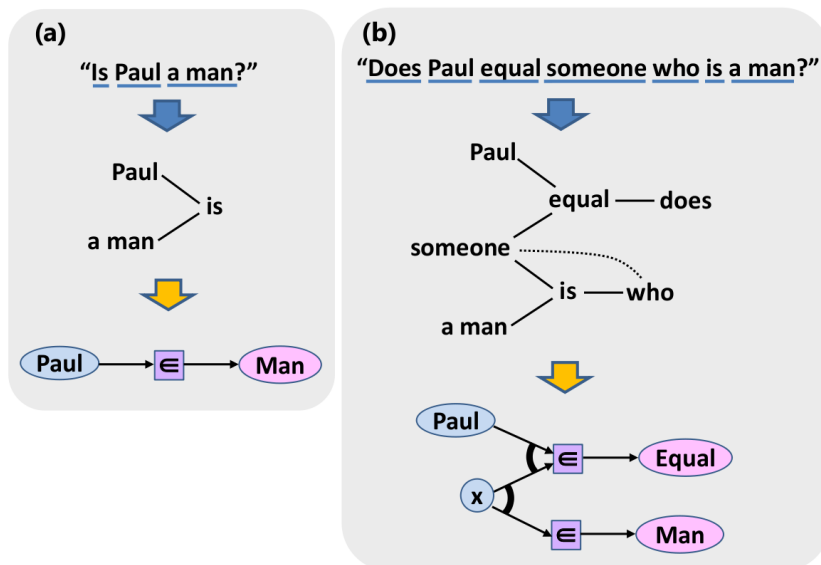


Figure 5 自然英語文（疑問文）から pKTN への変換の例。pKTN に習い、依存構造グラフでも動詞述語を右に配置している。

wife' が述語に変換されているが、(b) では動詞 'equal' が省略されず、'wife' は関数記号へと変換されている。作られる pKTN は (a) と (b) で異なるものとなるが、論理的には等価である。

Figure 5 の (a) と (b) には同様に、同じ内容を含む異なった疑問文の例を示す。疑問文であるため、Figure 4 と異なり、動詞／述語が図の右側に配置されている。(b) には疑問を表わす 'does'、関係節を表わす 'who' が含まれるが、これらは依存構造グラフで、それぞれ動詞 ('equal') や be 動詞 ('is') のさらに右側に配置される。なお、(a) では be 動詞が省略されるため、補語である 'a man' は述語へ変換されるが、(b) では 'a man' (不定冠詞＋名詞) は 'x' ＋ 'Man' (変数＋述語) へと変換されている。この 'x' は、代名詞 'someone' が変換されて出来る変数

を兼ねている。

Figure 6 の (a) と (b) は、良く用いられる『ロバ文』[18] の異なる表現である。まず (a) は、関係代名詞を使って仮定を暗示的に表わした例である。依存構造グラフでは、関係代名詞 'who' に引っ張れて動詞 'own' は右へ配置され、それによって pKTN でも述語 'Own' は右側へ来る。不定冠詞＋名詞 ('a farmer' と 'a donkey') はともに変数＋述語へと変換される。一方 (b) では、仮定が if-節によって明示的に表わされている。'if' に引っ張れた動詞 'own' はやはり右へ配置され、それによって述語 'Own' も右側へと来る。このロバ文の 2 例は、意味が等しいが構造が異なる自然言語文が同一の pKTN へと変換された。なお、ここで生成された変数 x,y は、形式上はともにスコレム

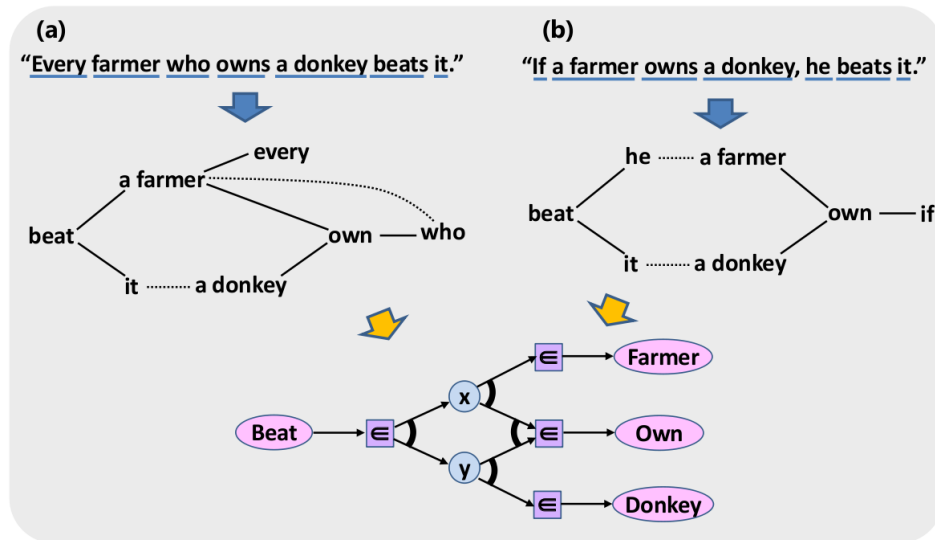


Figure 6 自然英語文 –(a) 関係代名詞節、(b) 仮定節を含む文– から pKTN への変換の例。pKTN に習い、依存構造グラフでも仮定節を含意する関係代名詞節や、仮定節を表わす部分では動詞述語を右に配置している。

標準形の中の全称限量された変数であるが、ELISE を用いた後向き推論では、クエリ起因のトークンだけが根側から流れてくるため、ロバ文が抱える限量子の問題をごく自然な形で克服することが出来る。

5. 結論と今後の課題

一階述語論理をペトリネット形式で表わす pKTN (Petri-net-baese Knowledge Transitive Network) は、ネットワークという構造化形式を取りつつ、RDF, OWL 等が持つ論理表現能力としての限界 (関数記号を記述できない、述語の引数は原則 2 個まで等) を取り払い、一階述語相当の論理を表わせるようにした知識表現形式である。本論文はこの pKTN に注目し、それを自動変換によって構築するための 2 つの方法について論じた。第 1 の方法はホーン節集合で表わされる論理プログラムから変換して作る方法、第 2 の方法は自然言語を依存構造解析し、得られる解析木を変換することによって作る方法である。前者の方法は全てのホーン節論理を変換できるという意味で完全であるが、後者の方法はまだいくつかの文例で示されているに過ぎず、完全とは言えない。

本研究の今後の課題は以下の通りである。

- 自然言語から pKTN を構築する方法をさらにリファインする。本論文では考慮しなかった信念・命令・時制・副詞句等の表現に含まれる論理知識を、ELISE による推論性を確保しつつ pKTN 化することにより、変換をできるだけ完全なものに近づける。(この段階で pKTN は、必ずしも (ホーン節) 論理プログラムとは対応の取れない、フリー・トポロジーの高階述語論理グラフになる可能性がある。)

- 大規模な pKTN データベース上で、推論アルゴリズム ELISE [33] を走らせる実験を行ない、推論効率を検証する。特に知識の大規模化に伴うスケーリング特性を調べ、ELISE の持

つ超並列性と、現在広く用いられている P2P, Hadoop 等の並列処理のためのソフトウェア/ミドルウェアとの親和性を調べる。

- 論理推論と不確実性との融合を検討する。知識ベースが大規模化するとどうしてもその中に瑕疵や矛盾が含まれることになり、効率性を維持したまま、ELISE のアルゴリズムをそれらに対応するように改良・変更することが課題になる。これに対応するため、ELISE で用いられるトークンの“信頼度”の中に、ベイズ的な手法 (統計的機械学習の方法) を取り入れる手法も検討する。

自然言語で書かれた大量の非構造化データからグラフ構造化された論理データベースを構築する利点は、人間が問いかける深い意味 (論理) に関わる質問に対して、大体の傾向ではなく正確に答えを返せるようになることにある。これを行なう pKTN のアプローチは、従来の大多数のテキストマイニング・ツールがターゲットとしてきた「大量の文章に含まれる大まかな傾向を分析して可視化する」機能を補完し、より正確な推論が求められる知的財産、法律、保険、報道等の産業上の諸分野で応用可能な技術になると期待される。

参考文献

- [1] 赤間 世紀: セマンティック・ウェブ入門. カットシステム (2011)
- [2] Amati, G., Ounis, I.: Conceptual Graphs and First Order Logic. The Computer Journal **43**(1) (2000) 1-12
- [3] Baader, F., Horrocks, I., Sattler, U.: Chapter 3 Description Logics. In: van Harmelen, F., Lifschitz, V., Porter, B. (eds.): Handbook of Knowledge Representation. Elsevier (2007)
- [4] Bancilhon, F., Maier, D., Sagiv, Y., Ullman, J.D.: Magic sets and other strange ways to implement logic programs. In: PODS '86 Proceedings of the fifth ACM SIGACT-SIGMOD symposium on Principles of database systems. ACM New York, NY, USA (1986) 1-15
- [5] Bos, J., Clark, S., Steedman, M., Curran J.R., Hockenmaier J.: Wide-Coverage Semantic Representations from a CCG Parser. In: Proceedings of the 20th International Conference on Computational Linguistics

- tics (COLING '04) (2004) Article No. 1240
- [6] Bratko, I.: Prolog Programming for Artificial Intelligence (Third Edition). Addison-Wesley (2001) 邦訳: Bratko, I.(著), 安部 憲広 (訳) Prolog への入門 (Prolog と AI). 近代科学社 (1990)
 - [7] Chen, W., Swift, T., Warren, D.S.: Efficient top-down computation of queries under the well-founded semantics. *The Journal of Logic Programming* **12** (1993) 1-19
 - [8] Horrocks, I., Sattler, U.: Ontology Reasoning in the SHOQ(D) Description Logic. In: *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence* (2001)
 - [9] Horrocks, I., Patel-Schneider, P.F., van Harmelen, F.: From SHIQ and RDF to OWL: The Making of a Web Ontology Language. *Journal of Web Semantics* **1**(1) (2003) 7-26
 - [10] Huang, S.S., Green, T.J., Loo, B.T.: Datalog and Emerging applications: An interactive tutorial. In: *SIGMOD '11 Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*. ACM New York, NY, USA (2011) 1213-1216
 - [11] 勝野 裕文: 演繹データベースの形式的意味論. *情報処理* **31**(2) (1990) 198-205
 - [12] Kowalski, R.: Logic for Problem Solving. Elsevier North Holland, Inc (1979) 邦訳: コワルスキー著、山田他訳: 論理による問題の解法 - Prolog 入門. 培風館 (1987)
 - [13] Montague, R.: Proper treatment of quantification in ordinary English. In: Thomason (ed.): *Formal Philosophy*, Yale University (1974) 247-270
 - [14] 森下 真一: 知識と推論 (情報数学講座). 共立出版 (1994)
 - [15] Motik, B., Nenov, Y., Piro, R., Horrocks, I., Olteanu, D.: Parallel materialisation of datalog programs in centralised, main-memory RDF systems. *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence* (2014) 129-137
 - [16] Ramamohanarao, K., Harland, J.: An introduction to deductive database languages and systems. *VLDB-Journal* **3** (1994) 107-122
 - [17] 斉藤賢爾: 6 並列計算技術の応用, 6-1 分散処理言語 Overlay GHC とその応用可能性. *情報通信研究機構季報* **54**(2/3) (2008) 59-69 <http://www.nict.go.jp/publication/shuppan/kihou-journal/kihou-vol54no2.3/0601.pdf>
 - [18] 白井賢一郎: 自然言語の意味論. 産業図書 (1991)
 - [19] Sowa, J.F.: *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley: Reading, MA. (1984)
 - [20] Sowa, J.F.: *Conceptual Graphs*. In: van Harmelen, F., Lifschitz, V., Porter, B. (eds.): *The Handbook of Knowledge Representation*, Chapter 5. Elsevier (2008) pp. 213 – 237
 - [21] Steedman, M.: *The Syntactic Process (Language, Speech, and Communication)*. A Bradford Book (2001)
 - [22] Sterling, L., Shapiro, E.: *The Art of Prolog, Advanced Programming Techniques*. MIT Press, Cambridge (1986) 邦訳: Prolog の技芸. 共立出版 (1988)
 - [23] Suzuki, H., Yoshida, M., Sawai, H.: A data-flow network that represents first-order logic for inference. In: Kuo, Y.H., Tseng, V.S.M., Kao, H.Y., Hong, T.P., Horng, M.F. (eds.): *The 2012 Conference on Technologies and Applications of Artificial Intelligence TAAI, Proceedings* (2012) 211-218 DOI: 10.1109/TAAI.2012.44
 - [24] 鈴木 秀明: 逆算と進化により連立方程式を解くデータフロー・ネットワークの提案. 第 57 回システム制御情報学会研究発表講演会講演論文集 (SCT'13) (2013) 317-4
 - [25] Suzuki, H., Iwasa, Y.: A network-based evolutionary method to solve inconsistent simultaneous equations approximately. *AIP Conf. Proc.* **1558** (11th International Conference of Numerical Analysis and Applied Mathematics - ICNAAM) (2013) 2486-2491 doi: 10.1063/1.4826045
 - [26] Suzuki, H., Yoshida, M., Sawai, H.: A network representation of first-order logic that uses token evolution for inference. *Journal of Information Science and Engineering (JISE)* **30**(3) (2014) 669-686
 - [27] 鈴木秀明: 入力語にあまい性を許す構文解析器の提案. *言語処理学会第 20 回年次大会発表論文集* (2014) 428-431
 - [28] Suzuki, H., Yoshida, M.: Direct graphical representation of first-order logic for inference. In: Lambert, M.J. (ed.): *Logic Programming: Theory, Practices and Challenges*. Chap.4. Nova Science Publishers, Inc. (2014) 117-142 ISBN: 978-1-63117-853-5
 - [29] 鈴木秀明: 制約ネットワークにおける不良設定問題を解く一手法. *計測自動制御学会 第 5 回コンピューショナル・インテリジェンス研究会予稿 PG0008/14/0000-0037* (2014)
 - [30] 鈴木 秀明, 巖佐 庸: 制約ペトリネット内でトークン進化が満たすべき必須な条件. *計測自動制御学会 第 6 回コンピューショナル・インテリジェンス研究会予稿 PG0014/14/0000-0094* (2014) pp.94-97
 - [31] 鈴木秀明: FOL を表現する論理ネットワーク: 依存構造からの変換の試み. *言語処理学会第 21 回年次大会発表論文集* (2015) 976-979
 - [32] Suzuki, H.: Translating dependency structure into a Petri-net-based logical network. In: *Pacific Association for Computational Linguistics Conference 2015 Proceedings Book (PACLING)* (2015) 47-52
 - [33] 鈴木秀明, 巖佐庸: 構造化論理データベース上で動作するトークン型推論アルゴリズム. *人工知能学会第 106 回知識ベースシステム研究会資料 SIG-KBS-B502-04* (2015) 17-25
 - [34] Ueda, K., Chikayama, T.: Design of the kernel language for the parallel inference machine. *The Computer Journal* **33**(6) (1990) 494-500
 - [35] Ullman, J.D.: *Principles of Database and Knowledge-base Systems*, Vol. 2 Computer Science Press (1989)