

エピソードネットワークを用いた物語理解

野崎 広志 , 重永 実
山梨大学 工学部

本研究では物語理解を、物語の筋の展開の痕跡をエピソードのネットワークとして記憶の中に残す過程と見なし、「こぶとりじいさん」等の物語テキストを例にとり、質問応答、あらすじ生成、言い替えの出来るシステム JStory を作成した。質問応答ではエピソードネットワークや辞書知識を利用してテキストに陽に述べられていない事も答えられるようになっている。あらすじ生成では”あらすじ = 物語の設定部 + キーセンテンス”という図式を利用し、指定によりあらすじの長さ、生成の仕方を選べるようにした。最後にアンケートを採り、システムの実行結果と比較し、有効性を確かめた。

STORY UNDERSTANDING WITH
EPISODE NETWORK

Hiroshi NOZAKI and Minoru SHIGENAGA
Faculty of Engineering Yamanashi University
Takeda-4, Kofu, Japan

We have constructed a Japanese story understanding system (named JStory) which regards story understanding as a process of leaving an episode network representing semantic tracks of a story in memory, and it has question-answering function and gives an outline of a fairy story. It can answer questions about not only explicit statements described in the story but also implicit events, using the episode network and knowledge stored in the system's dictionary. Using a scheme: "outline = setting + expanding", JStory composes an outline of a story, and a user can specify the number of sentences composing an outline and the method of choosing key nodes in an episode network. Finally, the authors asked ten persons the same problems as the ones for the system and found out that JStory behaved pretty well as human did.

1. はじめに

計算機に物語を理解させようとする研究はいままでも数多くあり、特にR.C.Schankら^{3), 9), 10), 11)}の研究は先駆的であり、物語を理解させるには記憶の階層構造とも関連して様々なレベルの知識が必要であることを明らかにした。物語の構造についても様々な研究がなされ、D.E. Rumelhartら^{6), 7)}の物語文法は代表的なものである。

そこで、我々も人間が物語を理解しようとする場合には、様々なレベルの知識が必要であるとの観点から、スクリプト、プラン、ゴール等⁹⁾の知識や物語構造知識を使い理解を試みるための日本語物語理解システムJStory(ジェイストリー; Japanese Story Understanding System)を作成した。

物語の理解には例えば、

- 物語の内容についての質問に答えることができる
- 物語のあらすじを言うことができる
- 物語を言い替える事ができる
- 類似した物語を思い浮かべることができる

等、様々な側面がある。

そこで本研究では人が物語を理解する過程を、読み手の持っている実世界に関する知識や常識、物語の次の展開についての予測などを使って、物語中の事象を意味的に結び付け、物語の筋の展開の痕跡をエピソードのネットワークとして記憶の中に残す過程であると見なし、なおかつ物語理解の上記の側面を実現すべきであると考えた。そのため、次の方針の基にシステムを作成した。

- 物語中の事象と事象との間の関係を求めリンクでつなぐ。
- 物語テキストの中には陽に述べられていない事も、常識を用い物語表現に組み込む。
- こうして出来上がったエピソードネットワーク(意味ネットワーク)を物語表現とし、質問応答、あらすじ生成に利用すること。
- 質問応答とあらすじ生成は別々のシステムとして作るのではなく、理解を目的としているので、1つのシステムでどちらでも出来ること。

本研究では物語テキストとして「こぶとりじいさん」、「大きなかぶ」、「舌切り雀」を例にとり、質問応答、あらすじ生成を実際に行い、物語理解を試みた。質問応答ではエピソードネットワークや辞書知識を利用してテキスト中に陽に述べられてはいない事も答えられるようになった。あらすじ生成では「あらすじ=物語の設定部+キーセンテンス」という図式を利用し、ユーザーの指定によりあらすじの長さ、生成の仕方を選べるようにした。最後に、アンケートを採りシステムの実行結果と比較し、有効性を確かめた。

2. システムの概要

本システムはVAX11/785上のFranz Lispで書かれており、処理の概要は大きく分けて構文解析部、意味解析部、物語文脈解析部、質問文理解部、文生成部、あらすじ生成部から成る。構文解析部では、ローマ字で分かち書きされた入力文章を構文状態遷移ネットワークを辿って解析し、構文木をつくる。意味解析部では、構文解析部でできた構文木をもとにして、係り受け等の処理を行い、入力文の意味表現をつくる。文の意味表現は格構造、テンス、アスペクト等の情報を取り出したものと、CD表現⁸⁾からなる。物語文脈解析部では、意味解析部でできた意味表現から適切にスクリプト等の知識構造に起動をかけ、事象間のつながりをつくる(エピソードネットワーク)。質問文理解部では、質問文の類別や何を答えればよいのか判断し、質問文に対する答えを見つけるために、エピソードネットワーク又は辞書知識の中を探索する。文生成部では、質問文に対する応答文または、あらすじ文を実際に文としてカタカナ出力する。あらすじ生成部では、キーセンテンスの抽出と、選択されたキーセンテンスを縮約してあらすじ文の形にする。

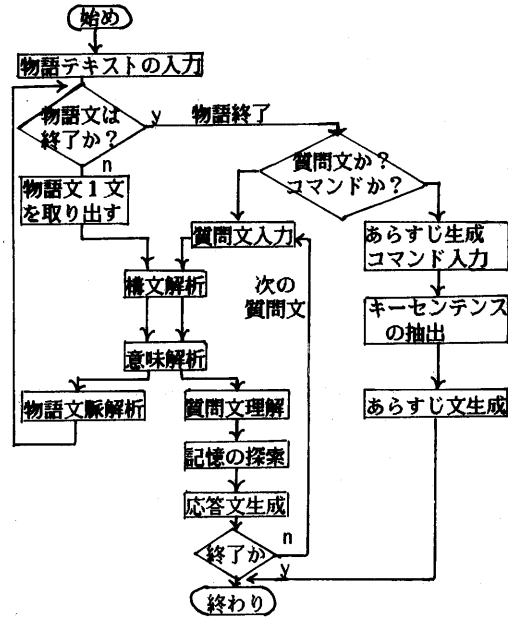


図2.1システムの処理の流れ

3. 文脈解析

文脈解析部では、意味解析部で意味表現にまで落とした文と、前文までの意味表現との間になんらかの関係を求め、事象と事象とをつなぎ、物語(文章)全体としての構造を抽出する。物語中の事象と事象とを意味的に結んでできるネットワークは、物語の進行の軌跡を残しており、これをエピソードネットワークと呼ぶことにする。文脈解析での目標は、大筋としてこのエピソードネットワークを作り上げることに他ならない。文脈解析で行う処理は簡単にまとめると、次のようになる。

- 1: 物語中の事象と事象との間の関係を求める。
- 2: 物語テキストのなかには陽に述べられていない事象を推論し、エピソード記憶の中に組み込む。

これらの処理のために本システムでは、スクリプト、プラン、ゴール、因果規則、物語の時空マップ、物語構造知識等の知識を利用している。ここで、物語構造知識は理解過程時には使っていない。

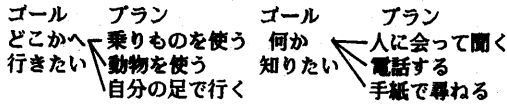
3.1 スクリプト⁹⁾

JStoryでは、スクリプトはプラン、ゴールの処理とも結び付いているため、シーンは固定されていず、いくらか柔軟である。例えば\$pos-effectのスクリプトは簡単に述べると次のような知識を表している。「誰か(X)が相手(Y)に好印象を与えたら、YはXにとってポジティブ(positive; +)などをする」

このスクリプトの成功条件は、YのすることがXにとって+のことでなければならない。ここでXにとって+のことというのは、Xが持っているゴールに貢献する様なことであり、このスクリプトはこの部分でプラン、ゴールによる処理が有効になる。このように本システムでは、スクリプトはシーンの固定化したものばかりではなく、プラン、ゴールの処理とも結び付いている。スクリプト中のシーン間にあるリンクはスクリプト辞書中に記述しており、スクリプトの各シーンが実体化(インスタンス)されたとき、エピソード記憶の中でリンクが張られる。

3.2 ゴール、プラン^{(9), (11)}

プランとゴールによる処理はプラン・ゴールルールによるプロダクション規則の他にプラン辞書、ゴール辞書を参照しながら行っている。プラン辞書には、そのプランに対するゴールやそのプランの選ばれた理由などが記載されている。ゴール辞書には、そのゴールに対して選択できるプランや反対のゴール、状況などが記載されている。プランとゴールの関係の例として例えば次のような関係でルールが組まれている。



3.3 因果規則

因果規則は事象の因果関係を、後向きと前向きの方向に予想される事象を記述したルールで、最も低レベルの規則である。一つの文から予想される(された)事象は後向きの場合には文脈を辿りリンクをつなぐ。前向きに予想されている事象がある場合は、それを待ち合わせパターン、条件、リンク名をリンク生成デモンに渡し、実際のリンク張りは、このデモンが次に入力されてくる文の意味表現を調べて行う。例えば、因果ルールには“壊れやすい物体Xに何か力が加わる”という事象に対しては前向き(result)には“Xは壊れる”という予想が記述されている。従って“花子は床に花瓶を落とした”の後文に“花瓶は粉々になった”とあればresultリンクが張られることになる。

3.4 物語の時空マップ

物語の時空マップとは、物語中で出来事の起こった場所関係や時間関係を表現したもので、エピソード記憶中の各イベントに対するポイントを含んでいる。

例えば「こぶとりじいさん」では次のような時空マップができる。これらのマップ情報は推論規則を適用する人物を限定したり、また時間や場所に関する質問文処理にも利用できる。例えば、“雨が降る”事象が起これば、雨を避けるゴールは建物の中にいない登場人物に推論される。

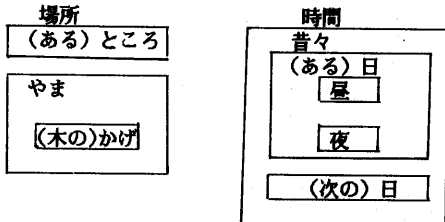


図3.4 時空マップ

3.5 物語の意味の表現

物語をどのように表現したら良いかという問題は、人間が物語を読んで(聞いて)理解した後、物語が記憶の中でどの様に表現されているかという問題と同じく複雑である。人間の記憶の中では単にイベントをリンクでつないだだけとは思えない。しかし、現在のところ文脈解析によりイベントをリンクでつないで出来上がったエピソード記憶は物語の展開の骨格をなしており、これを物語の内部表現としている。

3.5.1 エピソードリンク

事象間をリンクでつないでエピソードネットワークをつくるが、このリンクをエピソードリンクと呼ぶことにする。

表3.5 エピソードリンクの分類

リンク名	記号	説明
planfor	p	あるゴールに対して使われるプランを示す
subgoal	s	あるプランを達成させるためのサブゴールを示す
instan	i n	プランから行為イベントに向かい、そのプランの具体化(インスタンス)したものを示す
motive	m	ゴールとそのゴールに動機を与えたイベントをつなぐ
reason	R	精神的な事象または状態は行為の理由となり得る
initiate	i t	ある客観的事象または状態は偶然に精神的な事象を引き起こし得る
result	r	ある行為はある状態を引き起こす
enable	e	ある状態はある行為を可能にする
disable	d a	ある状態はある行為を不可能にする
achieve	a	あるイベントがゴールを達成させる
fail	f	あるイベントがゴールを失敗させる
sum	s m	いくつかの複数の事象から合算される

注：ただし、ある行為の結果(result)当然生じるある状態が他のある行為を可能(enable)にしている場合はresult-enableではなくenableとしている場合もある。例えば、“太郎は山へ行った。” enable “太郎は果の木に登った。”である。

3.5.2 物語構造についての知識

JStoryでは物語を物語=設定+展開という構造で見ようとする。物語文法^{(6), (7)}のように更に下位レベルにまでは展開しようとはしていない。なぜなら、Black⁽¹⁾の指摘のように物語を一文一文処理している理解過程時に、それが物語文法規則のどのカテゴリーになるか決定するのは難しいと思えるからである。物語を全て最後の文まで処理し終ってからなら不可能ではないように思えるが、それでは文法規則からの予測は使えない。

JStoryではこの構造は単に物語全体を設定部と展開部に分ける以外には現在のところ使っていない。これは、あらずし生成の処理時に物語の定型、つまり物語=設定+展開を用いてあらずしを生成しようとするためである。

処理は次のようにして設定部と展開部に分ける。

- 1) 物語は展開部が始まるまでは設定部である。(物語→設定+展開)
- 2) 物語の最初に設定部がなく、最初から登場人物が何か行動をするときには、その行動の前提を推論し、それを設定部とする。(物語←設定+展開)例えば、「大きなかぶ」の物語はこれにより設定部を抽出する。
- 3) 展開の始まりの識別は登場人物の行動の記述や“ある日”、“やがて”等のキーワードに頼る。
- 4) 状態性の述語文や登場人物の行動を記述した文以外の文が続いているならまだ設定部が続いているとみなす。

3.5.3 物語間の類似性について

人間が物語を読み進めている時に今読んでいる物語と同じ筋の展開を持った、過去に読んだ物語を思い出すことがある。そしてその思い出した類似した物語の筋の展開から今の物語の次の展開を予想したりする。過去に読んだいくつかの物語の記憶から似た物語を思い出すというのは難しいかもしれないが、もしも複数のテキスト間で類似した筋の展開があるなら、それは、それらのテキストの物語表現の中にも反映されるべきものであると考えられる。JStoryではこの類似性の処理は不完全ではあるが、物語間の類似は同じスクリプトが使われていることで捉えられている。もっと上位のレベルから類似性を捉えようとするなら、勧善懲悪のレベルから、良い人(+man)は良い結果(+)を得

るが、悪い人(-man)は悪い結果(-)を得る、というパターンが捉えられるが、JStoryでは残念ながら、このレベルまでの処理はしていない。

3.5.4 物語のエピソードネットワーク例

付録に物語テキストに対してJStoryの作ったエピソードネットワークを示す。各エピソードネットワーク図中の、例えばE-4.2は物語テキスト中に陽に現れている命題番号を示す。また、例えばD-16は物語テキスト中には陽に現れていないが、システム内部で推論されて組み込まれた命題(ノード)を示す。点線で囲まれている部分は、そのテキスト理解に於て起動されたスクリプトのうちのいくつかで覆われたノードを示す。一点鎖線での区切りは各命題により、どの登場人物が主語になっているかを示している。

4. 質問文理解

入力された質問文は意味解析部で意味表現に落とされると、質問文理解部に渡され、何を答えれば良いのか質問文の類別を行う。また、エピソード記憶の探索において、どこから探索を始めれば良いのかの決定や、質問文の前提、新・旧パターン、焦点も抽出する。

例えば、Q1="山へ行ったのは誰ですか?","Q2="なぜおばさんは雀の舌を切ったか?"という質問ではそれぞれ、

Q1	Q2
前提: 誰かが山へ行った	おばさんが雀の舌を切った
分類: 穴埋め	目的または因果前件
新旧: 旧・新パターン	旧・新パターン
焦点: 誰	(雀の)舌
となる。	

質問文の分類は文獻3),4)に依って表4.1のようにした。システムで実際に扱っている分類には"."が付いている。質問文の分類は「なぜ」、「どうやって」等の疑問副詞や格構造の中に変数成分があるかどうか等により行う。

表4.1 質問文の分類

・因果前件	・手段/操作	・穴埋め	・目的
・確認	・属性	・感情/感覚	・時間
・場所	・可能化	・選択	・できごと
why-not	因果結果	判断	同定
量/質	期間		

次に、記憶(エピソードネットワークや辞書知識)の中を探索し質問文に対する答えを見つける。探索の仕方は質問の分類にもよるが、もしもエピソードネットワークの中を探索しても直接に答えとなるものが見つからないときには、スクリプトやプラン、ゴール辞書又は単語辞書知識から何らかの結果を文生成部に渡す。

5. 文生成

文の生成は形態的には単一文の生成と連続文の生成に分けている。

5.1 単一文の生成

文の生成は基本的に格構造表現から作る。しかし、スクリプトやプラン、ゴール辞書中のCD表現からも文を作る必要があるため、CD表現を文にするときには一旦格構造に落とし、その格構造を文にしている。

CD表現から格構造へ変換するときは、CDの「リミティブ」とパターンから主動詞とその格要素(つまり格構造)を求めらる。

格構造から文を生成する時には、その述語の辞書項目をひき格に付き得る助詞を付ける。一般に一つの格に複数の助詞が付き得るが助詞の選択は行っていない。辞書項目中で最初に見つかる助詞を付ける。格要素に助詞の付いたも

の、すなわち句、のうちで長い句ほど文の先頭へ移動する。また埋め込み文がある時に、主文の主語と埋め込み文の主語が同じなら埋め込み文の主語を消去する。最後に、述語を活用させ、文として出力する。出力しようとするとき否定なら主動詞の反意語を主動詞として肯定にする。また必要なら視点を換えた文(視点変換文)をつくる。

5.1.1 視点変換文の生成

単一文生成の時に視点を換えた文を作るときには主動詞の辞書から、その視点の違う動詞を求め、その動詞の格構造に応じて、格を変更する。ただし、JStoryでは視点の変換は格構造に動作主体格と相手格を持ち、動詞辞書項目のVIEW-POINTに値を持つ動詞または受給表現¹⁴⁾を作る動詞にしか行っていない。

視点変換の指示はview-listというリストがNILかどうかで判別し、NIL以外、例えば((おじいさん-1 私-1))ならおじいさん-1の視点よりの表現にして、おじいさん-1を私-1に入れ換えて文を生成する。

5.2 連続文の生成

連続文の生成は単一文生成を連続して行ってゆくが、2つの連続した文の主語が同じであれば後文の主語の省略、2つの文を埋め込み化して1文にする、代名詞を使う、接続詞を挿入するという事で単一文生成とは異なる。

埋め込み化は前文と後文の句のうちで同じ句要素があれば、その句を前文の句から消去し、前文を後文のその句を連体修飾させることで埋め込む。その他に同じ句要素があれば、後文のその句には、性に応じて「彼」、「彼女」、「それ」の代名詞を使う。また発話しようとするエピソードネットワーク中のノードがFAILリンクを前向き方向に持っていれば、接続詞「しかし」を文の先頭に挿入する。ただし、発話しようとする文が1文だけのときは接続詞の挿入は行わない。

5.3 応答文の生成

質問文の分類とエピソード記憶の探索結果に応じ、応答文を生成する。

応答文の形式は質問文の分類や記憶の探索結果に強く結び付いている。

因果前件質問の場合にはエピソードネットワークを辿って応答文になるパスを質問の前提とマッチした物語表現のノードを後向きにリンクを入力イベント(先頭に"E"の付いたノード)まで辿って見つける必要があるが、そのとき見つかったパスは次のように応答させる。ここでパスの形式は

((<リンク名1> <ポイント1>)
 (<リンク名2> <ポイント2>)
 ...) } パスの長さ

である。表5.3の[]はその中のポイントの指しているノードを文生成したものを表す

表5.3 パスと応答形式

パスの長さ	条件	応答形式
1		[<# ¹ /N1>]からです
2	リンク名2≠planfor [<# ¹ /N1>]=data	[<# ¹ /N1>]ので [<# ¹ /N2>]からです
	それ以外	[<# ¹ /N1>]からです
それ以上	リンク名2=planfor	[<# ¹ /N2>]ので[<# ¹ /N3>] と思っただけです
	リンク名1=motiv	[<# ¹ /N1>]ので[<# ¹ /N2>] と思っただけです
	それ以外	[<# ¹ /N1>]ので[<# ¹ /N2>] それで[<# ¹ /N3>]からです

確認質問の場合には、

前提がマッチするものがあれば、

「はい」+<質問文の述語>

マッチするものがなく、穴埋め質問の答えが求まれば、

「いいえ」+<前提の訂正>

求まらなければ、

「いいえ」+<焦点要素の疑問代名詞化>+も+<否定形>

と答える。例えば、「花子は山へ行きましたか?」に対して「いいえ、花子は川へ行っただです。」や「いいえ、誰も行きませんでした。」である。

また、応答文はできるだけ文脈を参照した表現にした方が良く、文生成用の物語表現がスクリプトを参照しており、そこで使用され易い動詞、又は、その求め方がスクリプトに記載されていれば、生成する文の動詞の決定に利用している。例えば、「とる→とってあげる」(もちろん、これは視点の影響を受け、「とる→とってもらう」にも成り得る)。

その他に、応答文生成にあたっては、繰り返されている語の省略や代名詞の使用等により自然なつながりのある応答をする必要があるので、応答しようとする文が複数ある時には連続文、生成をする。

6. あらすじ生成

物語テキストを全て文脈解析した結果、その物語を表すエピソードネットワークが出来上がる。あらすじ生成におけるキーセンテンスの抽出はこのエピソードネットワーク中のノードから選ぶ。

6.1 キーセンテンスの抽出

本システムではキーワードからキーセンテンスを抽出することはせず、エピソードネットワークを利用する。

エピソードネットワークから選ぶノードは、次のような仮定¹³⁾のもとに選ぶ。

- ①どこにもリンクがつながっていないノードは忘れ去られて良いノードである。
- ②リンク数の多いノードはそれだけ物語の内容と強く結び付いているので、大事なノードである。
- ③ある問題、課題を解決しようとするゴール指向の物語の場合には、ゴールの失敗や達成を表すノードは重要である。
- ④予測していたことと異なることが起きた事象のノードは重要である。

本論文では④の予測の失敗は処理していないため①、②、③のことからあらすじ生成を試みた。リンク数の少ないノードほど取り出されないので①は考慮する必要はない。②ではユーザーにより、(i)リンク数、(ii)取り出す命題の数、(iii)全命題に対して取り出す命題数の何れかを指定してノードを取り出すことが出来るようにした(これをあらすじ生成1とする)。③ではachieveリンク、failリンク(この二つをゴール状況リンクと呼ぶことにする。)の付いている命題、或はそれに関連した命題のうちどの種類の命題を取り出すかを指定できるようにした(これをあらすじ生成2とする)。

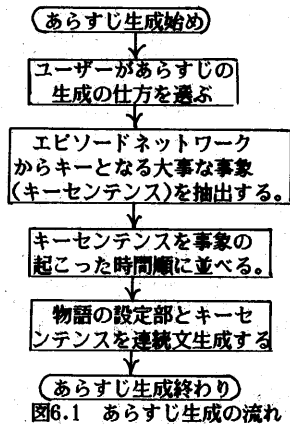


図6.1 あらすじ生成の流れ

じ生成2とする)。

従って、本研究ではあらすじを

あらすじ = 物語の設定部 + キーセンテンスとして捉えている。

また、あらすじ生成によって最終的に得られる文の割合を表すために要約率を次のように定義した。ただし、これには厳密な意味はなく、単に元の物語文に対する目安ではない。

$$\text{要約率}(\%) = (1 - \text{あらすじ文の数} / \text{全命題数}) \times 100$$

6.2 あらすじ生成1

これは全てのノードの中で付いているリンクの数が多し順にいくつか取り出しそれらをキーセンテンスとしてあらすじを生成しようとする方法である。

抽出の指定方法

ノードにつながっているリンク数	そのリンク数以上のリンクを持ったノードを選ぶ
ノード数	リンクの多い順からその数だけ選ぶ
要約率	要約率が指定された値になるように選ぶ

6.3 あらすじ生成2

あらすじ生成1では、リンクの数だけしかみていないが、あらすじ生成2ではリンクの種類を見てキーセンテンスを選ぶ。注目するリンクは、ACHIEVEリンクとFAILリンクとしている。この2つのリンクはゴールの達成と失敗に関与するリンクであるので他のリンクよりも重要だと思われるからである。従って、このリンクと結び付いているノードを、またはそのノードに関係しているノードを選択してキーセンテンスとする。選択の仕方はnormal、all、onlyの3つの仕方どれかを指定する。図6.3.1で“a”はACHIEVEリンク、“f”はFAILリンク、“E”は入力イベントを表す。

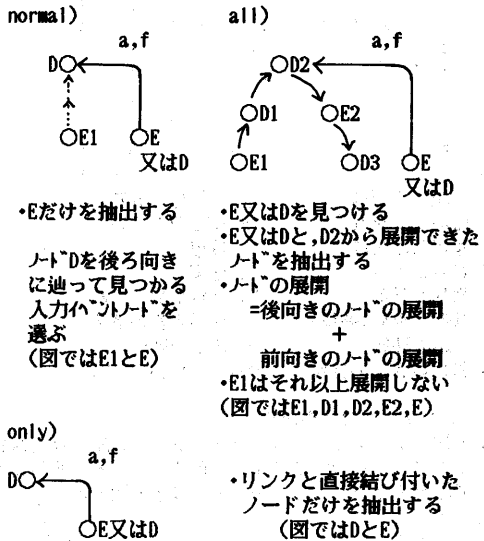


図6.3.1 あらすじ生成2でのノード抽出の仕方

7. アンケート調査

「Story」で処理した3つの物語テキストと同じものを計10人の人に読んでもらい、各項目の質問に紙に書いて答えてもらった。質問の項目は、①各物語テキスト中の文から大事と思われる文(命題)を選ぶ。②各物語内容についての

質問(JStoryでの質問応答と同じもののいくつか)に答える。
 ③各物語のあらすじを書くことである。このアンケートの目的は人間と本システムとの実行結果を比較検討するためである。本稿ではあらすじについての①, ③についての一部を示す。

7.1 アンケート結果の集計

7.1.1 ①の結果

①物語テキスト中で大事な文を選ぶ(複数可)

(図7.1.1参照)

7.1.2 ③の結果

③で集まった資料から、そのあらすじで言われている成分(これを機能因子と名付けることにする)を抽出し、それらを記号で表したものを表に示す。例えば、「良いおじいさんと悪いおじいさんがいました。良いおじいさんは鬼の前で踊りを踊りこぶを取ってもらいましたが、悪いおじいさんは良いおじいさんのこぶを付けられてしまった(被験者名H)」である。A, B, C, ...は機能因子、Tn, O, Kz, ...は被験者名である(表7.1.2参照)。

「こぶとりじいさん」		
命題番号	人数	
E-12	*****	6
E-17.2	*****	6
E-1	*****	5
E-13	*****	5
E-17.1	*****	5
E-2	****	4
E-16	***	3
E-14	*	1

図7.1.1 重要な命題番号頻度

表7.1.2 「こぶとりじいさん」機能因子

	Tn	O	Kz	Sz	H	K	Se	T	M	A	計	必	
A	*	*	*	*	*	*	*	*	*	*	10	+	*
B	*	*	*	*	*	*	*	*	*	*	10	+	*
C	*	*	*	*	*	*	*	*	*	*	5	+	*
D	*	*	*	*	*	*	*	*	*	*	7		
E	*	*	*	*	*	*	*	*	*	*	6		
F	*	*	*	*	*	*	*	*	*	*	2		*
G	*	*	*	*	*	*	*	*	*	*	9	+	*
H	*	*	*	*	*	*	*	*	*	*	5		
I	*	*	*	*	*	*	*	*	*	*	4		
J	*	*	*	*	*	*	*	*	*	*	2		
K	*	*	*	*	*	*	*	*	*	*	1		
α	*	*	*	*	*	*	*	*	*	*	5	+	*
τ	*	*	*	*	*	*	*	*	*	*	1	+	*

記号 記号の意味

- A 鬼がこぶを取る
- B 鬼がこぶを付ける
- C 踊りが上手
- D 踊りが下手
- E 真似(-MANの願望)
- F こぶの評価
- G おじいさんの評価
- H 鬼との出会い
- I 脅迫
- J 仕事
- K -MANが踊られる
- α 登場人物の導入
- τ 背景の時代
- MAN 悪いおじいさん

7.2 アンケートの結果から

①の結果から半数以上の人から大事だと判断された命題を文の形にし、その中に含まれる機能因子をあらすじに必須の機能因子ということであらすじの必須因子と名付けることにする。

- ・「こぶとりじいさん」・・・5人以上の人によるあらすじ
昔々あるところに良いおじいさんと悪いおじいさんがいました。良いおじいさんの踊りは大変上手だったので鬼はおじいさんの顔からこぶを取った。鬼は良いおじいさんの顔から取ったこぶを悪いおじいさんの顔に付けてしまった。
・「舌切り雀」・・・5人以上の人によるあらすじ

そして、(おばあさんは)雀の舌を切っていました。おじいさんは雀から小さいつづらを買いました。おばあさんは雀から大きいつづらを買いました。おばけがいっぱい

出て来ました。

③の各表には①の結果から求めた必須因子とシステムの作ったあらすじの中に含まれる機能因子とを一緒に載せてある(“必”の欄が必須因子で、“J”がシステムの作ったあらすじの欄である)。あらすじの評価は、必須因子を全て含んだあらすじであるかどうかにより評価することにする。ただし、ある機能因子X Xが他の機能因子Y Yを含んでいるなら、X XがあればY Yはなくてもあるものとして考える。例えば、「舌切り雀」では、

H(+MANは宝物を得る)はG(+MANはつづらを得る)を含む
 E(-MANはおばけを得る)はD(-MANはつづらを得る)を含む
 である。また、殆ど同じ意味を表していると考えても良い機能因子どうしは、そのどちらかがあれば良いものとする。例えばK(不純な動機)とI(真似)。

システムの作ったあらすじは、これら必須因子を全て含んでいるので、あらすじ生成の動作もある程度、正当なものと評価できるのではないだろうか。

しかし、あらすじはこれら必須因子だけから成るのではない。表を見て解るように人間の作ったあらすじには多く含まれるのに、システムからのあらすじには含まれていない機能因子もいくつかある。「こぶとりじいさん」では、DやEの欠如、「舌切り雀」ではAの欠如である。これらの欠如は抽出する命題の数にも影響されるのでシステムにもっと多くの命題を抽出するよう指示すれば、欠如をなくすることができる。しかし、あらすじの長さは長くなる。もっと、あらすじらしいあらすじは、登場人物の対比的な物語なら対比的なあらすじを生成できるようにすることや、また、原因-結果の枠組みを使う等の物語の内容に応じてあらすじを作り出す必要があると思われる。しかし、これらのヒューリスティックな方法は、それだけ物語内容に依存したあらすじ生成の仕方であり、もっと広く一般的なあらすじ生成の仕方、さらに検討が必要だろう。また、本研究では”あらすじ=設定+キーセンテンス”として捉えたがアンケート結果であらすじに設定部が言われているのは「こぶとりじいさん」だけであった。今後、あらすじをどういう枠組みで捉えるかについても検討が必要である。

以上、アンケート結果とシステムの実行結果例を比較してみた。より良い結果を得るためには、まだまだ問題は山積している。現段階では評価法としてアンケート結果とシステムの実行結果に含まれる成分を比較したに過ぎない。実行結果の評価法の確立も必要だろう。

8. おわりに

物語テキストからエピソードネットワークを作り、それを利用して質問応答、あらすじ生成に利用してみた。物語りテキストをエピソードネットワークで表現することの利点には、物語中の事象間のつながりが解りやすい、質問文に対する探索と応答が分かりやすい、リンク数で結び付きの強い大事な事象が解る、他の物語と似ている部分がある等がある。現システムで今後、取り組むべき主な問題点として次のことが挙げられる。

- ①以前に行われた推論結果を修正できる機構が必要
これに似た機構がTMS(Truth Maintenance System; 真理値維持機構²⁾)であるが、JStoryではエピソードネットワークを修正する必要がある。
- ②予測の破壊の処理とそれに伴った動的予測の機構が必要
これは①の問題とも関連している。予測の破壊^{1(9), 44)}に対しての処理はJStoryでは扱っていないが検出はできる。これはスクリプトからの逸脱、ゴールの失敗、プランの選択等からである。予測の破壊が起こっているかどうかの認識は通常の出来事の流れを知っている必要があるわけであるから、その意味からもスクリプトは重要である。予測の破壊が起こっている事象は物語の中で重要な事象を表しや

すいが、予測の破壊が起これば、それに伴い新たな予測をダイミツに創り出し理解に役立てる必要がある。

③質問に対する応答の評価法。あらずじの評価法の確立とシステム自体がそれらの評価法を利用して最適な結果を出せるようにすることが必要である。

質問に対する応答の評価法には3),4), あらずじ(要約)の評価法としては5)に一案が述べられている。しかし、使用している知識や構構、考え方の違いによりインプリメントすることが出来なかった。今後、本システムでの実行結果の評価法を確立することが望まれる。

参考文献

1)Black,J.B. & Wilensky,R.: "An evaluation of story grammars", Cognitive Science 3,(1979)
 2)Doyle,J.: "TMSによる問題解決",産業図書,(1986)
 3)Dyer,M.: "In-Depth Understanding",MIT press,(1983)
 4)Lehnert,W.G.: "The Process of Question Answering", Lawrence Erlbaum Associates,(1979)
 5)Lehnert,W.G.: "Plot Units and Narrative Summarization",Cognitive Science 4,(1981)

6)Mandler,J.M. & Johnson,N.S.: "Remembrance of things parsed:Story structure and recall", Cognitive Psychology 9,(1977)
 7)Rumelhart,D.E.: "Notes on a schema for stories", In D.G.Bobrow & A.Collins, Representation and Understanding Studies in Cognitive Science,(1975)
 8)Schank,R.C.: "Conceptual Information processing", North holland,(1975)
 9)Schank,R.C.: "Scripts,Plans,Goals and Understanding",Lawrence Erlbaum Associates,(1977)
 10)Schank,R.C.: "Reminding and Memory Organization: An Introduction to MOPs" Res.Rep.#170,Comp.Science Dept. Yale Univ. (1979)
 11)Schank,R.C.& Riesbeck,C.K.: "Inside Computer Understanding", Lawrence Erlbaum Associates,(1981)
 12)Schank,R.C.: "Failure-Driven Memory",Cognition and Brain Theory,(1980)
 13)Schank,R.C.: "E-ソートの記憶構造",In D.G.Bobrow & A.Collins, Representation and Understanding Studies in Cognitive Science,(1975)
 14)吉本啓: "英日翻訳における受給表現の扱いについて",自然言語処理45-2,(1984)

実行結果例

1 質問応答例

以下に、JStoryでの質問応答例を示す。応答文以外のものは質問文の答えとなるエピソード記憶間のリンクとそのリンクにつながっている事象を(リンク名 事象へのポインター)で表したリストである。

「こぶとりじいさん」

### Question is ...### 地`お`イ`ン`ル アマト`リ`ヲ`マ`シ`タ ### Answer is ...### ア`ク`ヲ`モ`ッ`テ`イ`カ`ツ`カ`ラ`テ`ス	### Question is ...### ヨ`イ`お`イ`ン`ル`地` キ`ノ`カ`ニ`ル`リ`マ`シ`タ ### Answer is ...### ((motiv (event 4 con-7)) (planfor (data 3)) (subgoal (data 8)) (planfor (data 9)) (instan (data 10))) ヨ`イ`お`イ`ン`ル`ア`メ`ニ` ル`タ`カ`ツ`カ`ラ`テ` アマト`リ`ヲ`シ`ヨ`ウ`ト`ト`ク`カ`ラ`テ`ス
### Question is ...### お`イ`ン`ル`お`イ`ン`ル`ヲ`モ`ッ`テ`マ`シ`タ ### Answer is ...### イ`イ`ヨ`イ`お`イ`ン`ル` お`イ`ン`ル`ヲ`モ`ッ`テ`イ`カ`ツ`カ`ラ`テ` アマト`リ`ヲ`シ`ヨ`ウ`ト`ク`カ`ラ`テ`ス	

2 あらずじ生成例

「こぶとりじいさん」

・5つ選んだ場合

ワ`イ`お`イ`ン`ト`ヨ`イ`お`イ`ン`カ`ア`ル`ト`コ`ニ`ム`カ`シ`ム`カ`ニ`イ`
 ワ`イ`お`イ`ン`ト`ヨ`イ`お`イ`ン`コ`ノ`ミ`ク`イ`コ`ノ`カ`お`カ`ニ`ア`ツ`
 ヨ`イ`お`イ`ン`カ`シ`ハ`加`リ`ヲ`シ`テ`イ`
 ヨ`イ`お`イ`ン`ノ`ホ`リ`ハ`シ`ヨ`ウ`ト`ク`タ`ツ`
 シ`ヨ`ウ`ヨ`イ`お`イ`ン`ノ`お`カ`に`カ`ト`ツ`テ`ア`ク`タ`ヨ`イ`お`イ`ン`ノ`
 ミ`ク`イ`コ`ノ`ヲ`
 ワ`イ`お`イ`ン`ノ`お`カ`ニ`ホ`リ`ヲ`ク`テ`マ`ツ`
 Summary Rate=76%

「大きなかぶ」

・ゴール状況リンクと結び付いているノードだけ(only)を選んだ場合

お`イ`ン`ル`カ`ノ`ヲ`モ`ッ`テ`イ`
 お`イ`ン`カ`オ`キ`カ`ツ`カ`ラ`カ`オ`キ`カ`ツ`
 お`イ`ン`ヲ`ヨ`ン`テ`キ`キ`お`イ`ン`ル`カ`ヲ`オ`キ`タ`メ`ニ`お`イ`ン`ニ`
 キョウヨク`テ`キ`キ`
 シ`ヨ`ウ`お`イ`ン`カ`オ`キ`カ`ツ`カ`ラ`カ`オ`キ`カ`ツ`
 コ`ノ`ヨ`ン`テ`キ`キ`お`イ`ン`ル`カ`ヲ`オ`キ`タ`メ`ニ`コ`ノ`ニ`
 キョウヨク`テ`キ`キ`
 シ`ヨ`ウ`コ`ノ`カ`オ`キ`カ`ツ`カ`ラ`カ`オ`キ`カ`ツ`
 イ`ク`ヲ`ヨ`ン`テ`キ`キ`コ`ノ`カ`ヲ`オ`キ`タ`メ`ニ`イ`ク`ニ`
 キョウヨク`テ`キ`キ`
 イ`ク`オ`キ`カ`ツ`カ`ラ`カ`オ`キ`
 Summary Rate=57%

「舌切り雀」

### Question is ...### お`イ`ン`ル`地`お`イ`ン`ル`カ` ### Answer is ...### ((result (event 4 con-8)) (disable (data 4 con-9)) (initiate (data 5 con-10))) ス`メ`カ`ヲ`リ`ヲ`ク`テ`マ`シ`タ`テ` ル`ハ`カ`ツ`カ`ラ`シ`タ` ル`テ`お`イ`ン`ル`シ`ヨ`ウ`ト`ク`カ`ラ`テ` ス`メ`カ` テ`キ`カ`ツ`カ`ラ`テ`ス	### Question is ...### お`イ`ン`ル`地`ス`メ`ノ` シ`ヲ`キ`リ`マ`シ`タ` ### Answer is ...### ((result (event 4 con-8)) (disable (data 4 con-9)) (initiate (data 5 con-10)) (motiv (event 5 con-11)) (reason (data 7 con-14))) ス`メ`カ`ヲ`リ`ヲ`ク`テ`マ`シ`タ` お`イ`ン`カ`オ`キ`カ`ツ` ル`テ`シ`ヨ`ウ`ト`ク`カ`ラ`テ`ス
--	---

「大きなかぶ」

### Question is ...### お`イ`ン`ル`カ`ヲ`オ`キ`タ`メ`ニ`コ`ノ`ハ` お`イ`ン`ル`カ`ニ`お`イ`ン`ニ` キョウヨク`マ`シ`タ`	### Answer is ...### 2`ハ`ン`デ`ス`
--	------------------------------------

「舌切り雀」

・ゴール状況リンクと結び付いているノードだけ(only)を選んだ場合

お`イ`ン`ト`お`イ`ン`カ`ア`ル`ト`コ`ニ`ム`カ`シ`ム`カ`ニ`イ`
 カ`ラ`シ`ヲ`お`キ`カ`ツ`お`イ`ン`ニ`ス`メ`ノ`お`イ`ン`カ`ヲ`ワ`シ`マ`シ`タ`
 Summary_Rate=90%

3 視点変換例

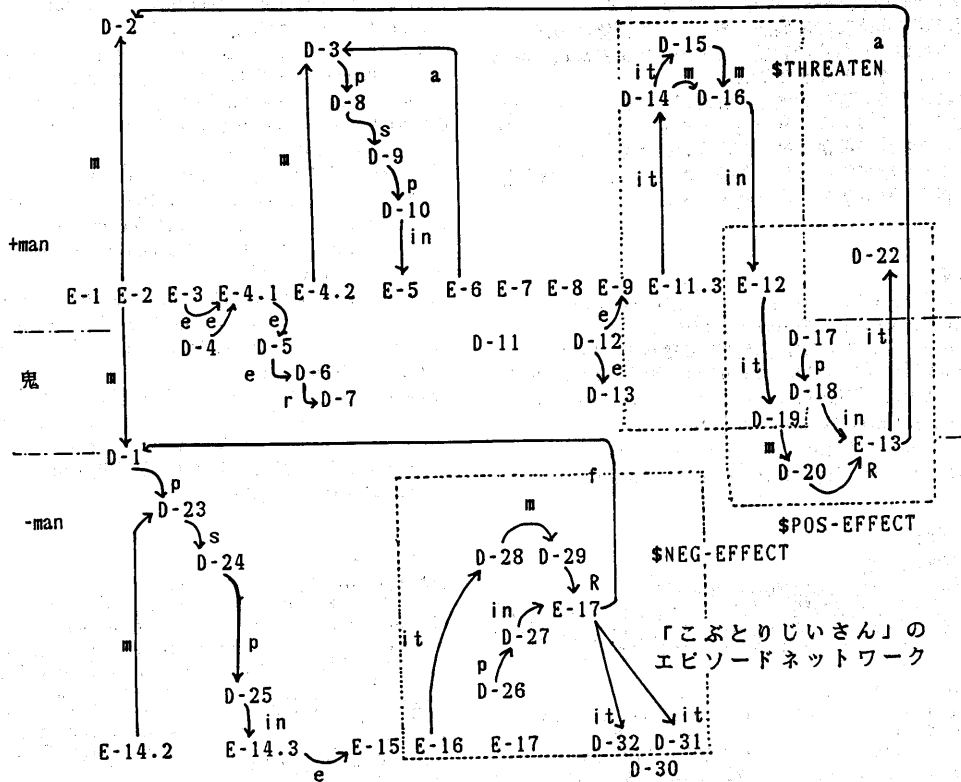
「こぶとりじいさん」

・良いおじいさんからみたとき

ワ`イ`お`イ`ン`カ`ニ`ミ`ク`ラ`レ`シ`マ`ツ`
 ワ`イ`ノ`ミ`ク`イ`コ`ノ`ヲ`ワ`イ`ノ`お`カ`に`ホ`リ`ト`ツ`テ`ク`レ`タ`

・鬼からみたとき

ヨ`イ`お`イ`ン`ヲ`ホ`リ`ニ`ミ`ク`ラ`レ`シ`マ`ツ`
 ヨ`イ`お`イ`ン`ノ`ミ`ク`イ`コ`ノ`ヲ`ヨ`イ`お`イ`ン`ノ`お`カ`に`ホ`リ`
 ト`ツ`テ`ク`レ`タ`



1.1 「こぶとりじいさん」テキスト

昔々あるところに良いおじいさんと悪いおじいさんがいました(E-1)。良いおじいさんにも悪いおじいさんにも顔に醜いこぶがありました(E-2)。ある日、良いおじいさんは山へ仕事に行きました(E-3)。芝刈りをしていると、雨が降ってきました(E-4.1 E-4.2)。おじいさんは急いで大きな木の陰に入りました(E-5)。夜になって、雨がやみました(E-6 E-7)。その時、たくさんの鬼が現れました(E-8)。おじいさんは鬼に見つかってしまいました(E-9)。もしもおじいさんが鬨りをしなければ、鬼はおじいさんを食べるとおじいさんに言いました(E-11.1 E-11.2 E-11.3)。しかし、おじいさんの鬨りは大変上手だったので鬼はおじいさんの顔からこぶをとりました(E-12 E-13)。鬼が良いおじいさんの顔からこぶを取ったことを聞いた悪いおじいさんは、次の日、山に行きました(E-14.1 E-14.2 E-14.3)。悪いおじいさんは鬼と一緒に鬨りをしました(E-15)。おじいさんの鬨りは大変下手でした(E-16)。鬼は良いおじいさんの顔からとったこぶを悪いおじいさんの顔につけてしまいました(E-17.1 E-17.2)。

- | | |
|-------------------|-------------------|
| D-1 -manはこぶをとりたい | D-17 鬼はこぶをとりたい |
| D-2 +manはこぶをとりたい | D-18 (鬼)とるプラン |
| D-3 +manは雨を避けたい | D-19 鬼は喜ぶ |
| D-4 +manが仕事場へ行く | D-20 鬼はこぶをとる |
| D-5 +manが柴を掴む | ことにする |
| D-6 +manが柴を入れ物に | D-22 +manは喜ぶ |
| いれる | D-23 -manは鬼に自分の |
| D-7 +manは柴を持っている | ゴールを代行させたい |
| D-8 +manの雨宿りプラン | D-24 -manは山へいきたい |
| D-9 +manが木に近づく | D-25 -manは自力で移動する |
| D-10 +manは自力で移動する | D-26 鬼は何かつきたい |
| D-11 +manは晴れを待つ | D-27 (鬼)つけるプラン |
| D-12 +manと鬼は近くにいる | D-28 鬼は怒る |
| D-13 +manは鬼を見る | D-29 鬼はこぶをつける |
| D-14 +manが恐怖を感じる | ことにする |
| D-15 +manは鬼に食べられ | D-30 -manは山からどこかへ |
| たくない | 行く |
| D-16 +manは鬨ることにする | D-31 -manは怒る |
| | D-32 -manは泣く |

注: +man 良いおじいさん
-man 悪いおじいさん