M0002380TP

1479079

AUTOMATING ABELL'S THEORY OF COMPARATIVE NARRATIVES

TOBY RICHARD CONYERS

A thesis submitted in partial fulfilment of the
Requirements of the University of Greenwich
For the Degree of Doctor of Philosophy

April 1999

FOR USE IN THE LIBRARY ONLY

Thesis

UNIVERSITY OF GREENWICH LIBRARY

The Automation of Abell's Theory
of Comparative Narratives

The purpose of this thesis is to demonstrate the progress that has been made towards the goal of producing a prototype computer model of Abell's Theory of Comparative Narratives, and subsequently, designing metrics to rigorously measure Abell's concept of 'closeness' of texts.

The production of such a model does not simply involve the mechanical (though distinctly non-trivial) transference of Abell's theory from paper to machine; various facets of the theory are not of a sufficiently high specification for a computer model and the fulfilment of such a computer model requires attention to these areas, specifically :

i) a repeatable method of comparing the structures of individual events;
ii) a consistent procedure of comparing the overall structure of a pair of texts, following on from Abell's basic concept of *paths of social determination*.
iii) metrics to demonstrate that the solutions proposed do indeed address the shortcomings of Abell's theory.

In order to preserve the qualitative nature of the theory and to demonstrate its potential real-world uses, the computer model attempts to avoid complex mathematics as far as possible and to produce transparent, non-expert results.

Acknowledgements

I would like to thank Professor Martin Everett and Leroy White for their assistance when required and their supervision of this project, for which I am most grateful.

Nick Croft was invaluable in getting around day-to-day programming problems, and I should like to extend my gratitude to him.

I should also like to acknowledge both my parents and the Engineering and Physical Sciences Research Council for the cash they provided which meant I could eat for the duration of the project.

Finally, I must thank Earl Grey for his tea, and the Circle Jerks for the 'Golden Shower of Hits' LP, particularly 'Coup D'Etat' without either of which this thesis would never have been finished.

Contents

I certify that this work has not been accepted in substance for any degree, and is not concurrently submitted for any degree other than that of Doctor of Philosophy (PhD) of the University of Greenwich. I also declare that this work is the result of my own investigations except where otherwise stated.

Toby Conyers

Professor M.G.Everett

CHAPTER ONE :
INTRODUCTION

Introduction

Much of the information that we as humans need to assimilate is presented to us in the form of text - speech is probably the most widely used form of inter-person textual communication, along with more literally textual data such as books, magazines and newspapers. On presentation of this information, human beings apply their (usually imperfect) knowledge of their environment and construct meaning from this. One form that this meaning construction could take pertains to narrative-style texts; that is, the description of a sequence of events that possibly culminate in some (perhaps undesirable) result state. Alternatively, the narrative may simply take the form of a sequence of actions with no particularly notable outcome.

Abell's theory of Comparative Narratives is an attempt to introduce a degree of formalism to a qualitative procedure for the analysis of stories given by actors immersed in the social world they help to shape. Narrative accounts of their views are taken and analysed to produce a series of directed graphs which show the causal connections between the events described in the text. Detail local to a text can then be gradually removed, or *abstracted*, and the resultant structures compared for degrees of similarity.

Tools used in the area of the analysis of qualitative data have often been little more than programs that assist the user in actually doing all the work, for instance GUI front-ends to speed the laborious task of coding a text and producing grounded theory. The lack of formal grounding to Abell's theory has been noted before (Heise, Willer). Although Abell's theory embraces the partiality and self-service to be found in an actor's account of an action in which they were involved, its lack of formal grounding means that no consistent analytical procedure is ever defined. A computer algorithm, once implemented, will produce the same results from the same input data *ad infinitum*. Such "repeatability" is central to the concept of scientific analysis and is one of the areas in which Abell's theory can be justifiably criticised and therefore improved.

Content analysis often incorporates statistical analysis of information (Popping & Roberts), an approach which was felt to breach the basically qualitative flavour of Abell's theory. Such approaches are themselves controversial; many people cannot see the link between qualitative data and the statistical results gleaned from them. In order to preserve the qualitative nature of the original theory, and to attempt to generalise the theory and its results for a potentially non-mathematical user, the aim was to avoid mathematical and statistical functions as much as possible; the implemented model contains no statistics at all, and the extent of the mathematics is to count the length of a vector!

1. Focus of the project

Abell's theory may be seen as an attempt to bridge the gap between very structured algorithms and more subjective methodologies. This project was undertaken to see whether such a theory did indeed bridge this gap. In order to fulfil the theory's desired goal as a method for comparing textual accounts of an event, the following question needs to be answered :

Is it possible to use Abell's theoretical structure to produce a reliable metric for the comparison of narratives?

Subsidiary to this central question are the following issues :

i) Although the theory of Comparative Narratives has been presented in an algorithmic fashion, *in what areas is the original theory insufficiently specified*?

ii) Subordinate to question i), *what solutions can be found to resolve these theoretical 'holes'*?

iii) In order to remove the high degree of subjectivity that Abell's analyst-led view encourages, *how is a computer model of the revised theory to be implemented*?

2. Subsidiary questions

In order to build up to the answer to the academic question, the following questions must be addressed :

i) *What form would the metric take?* Metrics have been suggested before, and require evaluation. Are they suitable, and if not, what form should a text comparison metric take, in the context of Abell's theory?

ii) *Can the metric be made user-independent?* By implementing a computer model of the thesis, a large degree of the subjectivity can potentially be removed, making the theory independent of the analyst.

iii) *Does the implemented metric behave as we would qualitatively expect?* By adjusting the model parameters and input texts, does the metric accurately reflect the qualitative judgements made by the user?

3. Programme of Work

In order to answer the questions above, a programme of work is needed which will lead to the answers required. The following tasks require completion before the question can be answered :

i) *An initial determination of the metric.*

ii) *A series of experiments to evaluate the metric.*

iii) *Adjustments to the evaluated metric.*

In order to implement the metric, a prototype computer model will be constructed which embodies Abell's original theory. The prototype will be used to test the metric independent of the user of the model.

4. Plan of the thesis

The remainder of chapter one takes a general overview of the basic tools needed to produce the prototype computer model, dealing with the parsing and planning elements of the model.

Chapter two presents Abell's theory of Comparative Narratives; this chapter is drawn exclusively from Abell's book, *the Syntax of Social Life*.

The algorithms discussed in chapter one and implemented in the model are explained, and provide details of how the model constructs a "meaning" from the text. The event structures upon which the metric operates are detailed, along with the algorithms by which the model disambiguates amongst a collection of possible meanings, and how the causal links are identified within the text.

Chapter four discusses the approaches suggested to the central flaw of Abell's thesis, that being how do you finely compare the structures that Abell's theory produces? The initial metric implemented is discussed, along with refinements to that metric. The three metrics implemented are then compared and contrasted.

Chapter five presents some example runs of the model on a series of input texts. The steps required to take the domain into new areas are discussed and displayed.

The thesis concludes with a discussion of the work done and the quality of the results obtained.

5. Literature review

A Natural Language Processing project, such as this, has associated with it two specific problems, those being the initial parsing of the text and the representation of such information as is drawn from the text. This particular project also utilises the concept of plans, and a central problem in their use has traditionally been how to recognise the plans that the user is following. A variety of algorithms have been suggested for this, and are explained in section 2.2.

5.1 Parsing and event representation

Although this is not intended as a parsing project *per se*, it may be of interest to briefly discuss the various options that are open to someone wishing to construct a parser.

A parsing algorithm generally falls into one of two camps; top-down or bottom-up. Top-down parsers traditionally decompose the text in some way, such as *formal grammars* (Chomsky) which are composed of sets of rules of the form 'S $\rightarrow$ NP VP', interpreted as 'A Sentence is composed of a noun phrase and a verb phrase' (Sabah), or *Transition Networks* (Lehnert, Tennant, Winograd). Bottom-up parsers work on the principle of analysing words in turn and constructing larger categories and relations from them via syntactic and semantic knowledge bases. Top-down grammars have been based on Definite Clause Grammars (Pereira & Warren) or Phrase Structure Grammars, in which context-free rules construct individual phrase structures by specifying valid word combinations and then establish relations between them. Syntactic rules have subsequently been divided into separate groups, immediate dominance rules detailing features and sub-categories of a notional lexical category, and linear precedence rules, describing grammatical constraints that need to be fulfilled in category ordering.

Top-down grammars have been popular because they are easy to write and the grammatical information they contain can be ordered to speed up the parsing process in some cases. However, they often don't degrade well (that is, the parse either succeeds or fails totally), and can wastefully create the same structures repeatedly because of the backtracking process.

Semantics are represented more explicitly by *case grammars* (based on work by Fillmore), which tie the deep structure of a sentence on the relations between the main verb of a sentence and its associated clauses. Case structures themselves divide into many forms. *Deep Case* signifies the relation between the predicate (often the verb) of a sentence, and one of its arguments. *Conceptual cases* are similar in some respects in that they are divorced entirely from any notion of syntactic structure; Schank's primitives (explained below) are an example of this. Then, roles within *case-frames* are fixed according to conditions attached to semantic properties of these clauses, rather than relying on more general lexical labels like Subject and Object (also known as *surface cases*). Thus, while a simple syntax-based parser will have difficulty distinguishing between "*the pecan pie baked to a golden brown*", and "*this oven bakes evenly*", a case system will simply sort through the alternatives until a meaning representation and its associated roles is found that matches with the specific language used by the source text. For instance "*The mirror broke*" can be disambiguated because an artefact in the subject slot is the Semantic Object, whereas "*the mirror polished*" makes no sense because the same rule doesn't hold true for the verb this time.

Various systems have been drawn from these distinct case categories - Fillmore defined a series of case roles such as Agent, Object, Instrument, Goal, and so on, which combined to weigh the likelihood of a case relationship between a verb and an associated noun group. Schank's *Conceptual Dependency* (see below) had a series of case roles (ACTs had an Agent, for example) which, when analysed in conjunction with primitive event structures, produced semantic structures corresponding to the meaning of the sentence. Grimes divided cases into 4 groups, *orientation*, *process*, *agentive* and *beneficiary*. Therefore, "*the wind opened the door*" is treated differently to "*the key opened the door*", because "*wind*" has an Agentive role as a *Force*, whereas "*Key*" does not, having an Agentive case as an *Instrument*.

Case grammars can also be represented with *Semantic Networks* (Demmer, Simmons & Correira) which once again relate clauses from the source text to the concept embodied in the verbs of the sentence. Simmons said that the *semantic definition* of a verb could be given by describing the properties of nominal concepts related to a verb. These relations were examined via the use of semantic nets.

Frames are once again used, in a hierarchical fashion in *Functional Grammars*, in which attribute-value (which can be represented by another frame) pairs are used to build the representation of a sentence. Zhang in his work on the story parsing grammar SPG also uses case frames, in hierarchical structures known as *case-frame forests*, to represent complex semantic phenomena from the stories.

Haun *et al* deal in the first instance with the text at the word level and subsequently impose structure on the input using semantic information derived from the concepts the words embody. A knowledge-base of information concerned with the types of complements that concepts are permitted to have helps to produce ambiguity-free relations between syntactic structures produced during the parse. Jacobs and Rau discuss a similar methodology, known as *relation driven control*. A *relation* is constructed as a triple, containing a *Head*, a *role* and a *Filler*. The lexicon associated with the parser includes information on the semantics of individual word senses, which is then used to rate the suitability of the potential Role and Filler to the Head. Cullingford and Pazzani use a similar algorithm as part of their DSAM system, but divide the disambiguation into three sections; i) that which can be resolved by utilising the syntax of a sentence to identify its unique meaning (for instance, "*visiting relatives is a nuisance*" means the narrator is visiting, not the relatives), ii) that which can be resolved by *surface semantics* (for instance, in "*the boy kicked the ball*", the combination of *kicked* and *ball* uniquely resolves the meanings of both the verb and noun, and iii) that which requires *world knowledge* and *context* (for instance, "*John had hair on his chest*" suggests, because of world knowledge, that John is hirsute, rather than the possessor of furry luggage.

Heuristics have also been put forward as a method of improving the efficiency of rule-based parsing systems (Huyck & Lytinen). They feel that grammars claiming to represent the entire structure of a language are too powerful to be made algorithmic, and that the alternative, *restricted grammars* fail for two reasons :

1) Such parsers often don't degrade gracefully; that is, they succeed in the parse or fail totally.

2) The kinds of sentence they can be made to parse are often not the kinds of sentence that people actually use.

On small domain studies, the heuristics proved to have a great deal of success in reducing the number of rules fired during a correct or partial parse of a relatively simple sentence. However, a repeat study on a much larger database of terrorist stories failed totally, falling back on a standard chart parser, until the set of heuristics was enlarged. The system was unaware of several syntactic structures and the heuristics were sometimes found to be too heavily based on syntax at the expense of semantics, thereby drawing the wrong conclusions. The system can be expanded of course, but the authors felt this would lead to a degradation in performance; reducing the number of rule hits, only to expand the number of heuristic checks.

Heuristic-like approaches have also been applied to syntactic parsers. Predominantly syntax-based parsers perform a greater or lesser degree of preprocessing on the text (as detailed in Hockey), including the removal of suffixes from the input text, and the *tagging* of words into tentative word types. A method of dividing clauses up prior to parsing is also explained, whereby the limits of clauses are searched for heuristically, the clause then being constructed backwards.

Regardless of the method of parsing, the form the parser's lexicon takes is very important. Many lexicons include semantic information (Haun *et al*), detailing word-sense restrictions on objects of verbs, and so on. Jacobs and Rau's relational-based parser includes lexical information on the confidence with which concepts can be associated together. This information is subsequently used to assist the parser in removing ambiguous meaning when confronted with more than one possible relational triple.

Studies have also been done at the more detailed phoneme (spoken) or morpheme (word shape) levels, neither of which are of interest here.

Abell's theory divides a text into series of discrete events, and therefore the method of representing these events is important within the scope of the model.

Traditionally, *frame-based* or *semantic net* structures, in one form or another, are used. Haun *et al* use a set of concepts linked via *is-a* and *has-<structure>* links, forming a network of related concepts - a semantic net. Individual elements have slots, therefore representing a frame structure. Luh describes a similar design for his object-oriented simulation system DEVS; a series of *entities* with *slots*, *aspects* or decompositions, and *specialisations*, or implicit *is-a* links.

Jacobs and Rau discuss *templates*, which are once again essentially frame-like structures, with associated roles or *slots* to fill. Allen, in his work on temporal logic also uses a structure that possesses many of the qualities of a frame, although in a stripped down form. He points out the problem of adequately defining a frame that can then deal with all forms of input that such a concept may take, using the following example :

"Jack lifted the ball".

"Jack lifted the ball onto the table".

"Jack lifted the ball onto the table with tongs", etc.

Although all these examples are based on the same LIFT event, a frame that has sufficient slots to deal with the final example is over-specified to varying degrees for the others. The addition of slots is somewhat problematic because more and more variables become necessary to keep track of all the extra information. Alternatively, Allen suggests a new predicate be created with the same basic intention (in this case, describing a LIFT event) but with the extra slots required. Allen goes on to say that this method is unsuitable because eventually the system will contain many predicates all doing the same job. His solution is akin to Schank's Conceptual Dependency *primitive* structures; each predicate is made to represent a single concept, and they are then combined to form more complex actions. The final example above is then represented by the following complex predicate structure (complete with dummy variable instantiations and a notional time, *t1*) :

$\exists e.\mathrm{LIFT}(Jack34, ball26, t1, e) \wedge \mathrm{dest}(e)=table5 \wedge \mathrm{instrument}(e)=tongs1.$

Schank's theory took this idea considerably further, reducing all actions to a set of 11 primitives, which then combined with each other and various case structures to produce the intended meaning. Thus, to look at a painting was defined by the predicate Attend(*eye*, *painting16*).

One of the drawbacks of either frame-based or semantic net systems is in completeness; a reasonably-sized set of case frames that can be made to cover every concept produced by a natural language has never been found, and many examples of semantic nets operate on very small domains (in Kautz's case, fixing pasta dishes, and in Haun *et al* very general computer hardware databases, Charniak's Ms. Malaprop was concerned with painting), demonstrating the theory adequately, but making no mention of the time and effort required to link all concepts together in a complete and consistent way, or how such a huge mass of information could be handled.

5.2 Plans and plan recognition

One of the problems facing the computer understanding of a piece of text lies in how to make connections between individual sentences. A broader analysis is required, sometimes known as *discourse analysis*. *Systemic Grammars* (Halliday, Winograd) can also operate on a very abstract level, by classifying the roles that utterances play in actor interactions. For instance, on a very simple level, an utterance may be either a *Question*, *Command* or *Statement*. Correlations may well exist between word order and the utterance's category : Noun Verb Noun may well be a Statement.

Story Grammars (Mandler & Johnson, Propp, Rumelhart, Simmons & Correira, Thorndyke, Van Dijk & Kintsch) have been proposed, which attempt to identify the global structure of the story in a 'top-down' fashion, in a way sometimes very reminiscent of phrase structure analysis, although dealing on a semantic rather than syntactic level. The assumption is that stories have conceptually separate parts that are generally identified inferentially by a reader. Such models are concerned not only with the process of understanding narratives, but with allowing for those properties of stories that narratives do not necessarily include, such as suspense, conflict and interestingness.

Propp postulated that a small set of actor roles and relationships between them could account for all the subtle differences between any pair of texts. For instance, by completely changing the story's actor roles, setting and so on, the same tale can be told in a different milieu.

Rumelhart proposed that the 'top-level' rules describing the structure of a story are :

Story → Setting + Episode

Setting → (State)*, meaning an unlimited number of states.

Episode → Event + Reaction

The successful application of these rules produces a tree structure with story propositions in the terminal nodes of the tree and more generic structures at the intermediate points. The phrase structure rules are adjusted to allow the binding of variables and testing of preconditions for rules. So the first rule Simmons & Correira display for a wild west story could be :

(OLD WEST SAGA) ← (SETTING GG BG L MOT OUTC)
(EPISODE GG BG L MOT OUTC),

where GG → Good Guy, BG → Bad Guy, L → Location, MOT → Motive, OUTC → Outcome. Subsequent rules are then used to bind these variables for future use to ensure a coherent story. Simmons & Correira suggest this as a model of story generation, whereby an action is submitted to the model in the form of a hypothesis; the model subsequently attempts to prove the plausibility of the hypothesis given the rules governing that story's generation. Meehan's TALESPIN operates in essentially the same way, telling a story by describing the solution of a particular problem. TALESPIN creates a *story world* at the start of its execution in which all characters are set up and their pasts filled in. The central concern of the model is in the development of suitable plans to assist the construction of a dramatic, interesting story.

Dehn produced another model called AUTHOR which worked from the basis that story generation models like TALESPIN failed to take into account the physical process of writing a story, by leaving out an author's ability to apply post-hoc justification for certain events that he/she knows in advance will appear in the story. Incidental characters can be created which provide an actor's motivation for performing some actions because of their relation with the character in the past. The author's motivation for this is to write a good dramatic story, not enter into Wilensky's notions of *goal co-operation* or *goal competition*. The author has a series of goals (the production of a plausible and dramatic story, illustrating facts important to the author) which undergo *conceptual reformulation* in light of the creation of the story. Initially, the author has an idea which is turned into a *kernel episode*, which in turn becomes a *sequence of episodes*, which possibly explain particular characterisations of an actor in the plot. Dehn calls this process *creative reasoning*, and lists its four characteristics, which are equally applicable to a planning model :

i) Sensitivity to unforeseen events,

ii) A willingness to be distracted by seemingly out-of-place events,

iii) the successive reformulation of goals,

iv) A sense of direction towards a) goals and b) environments in which a character can flourish, which is conceptually similar to checking for plan failure.

Brewer & Lidenstein see a story as more than a series of directed actions in which an actor struggles to achieve a goal. An actor could be taking part in a mundane planning situation (driving home from work) and not experiencing any difficulty, unaware that there is a bomb underneath the car. The reader is aware of this, and therefore the narrative becomes a story in their eyes.

Mandler & Johnson talk in general terms about linking these semantic story structures together in three ways : AND (simultaneous action), THEN (either regardless of order, or enabling) and CAUSE (although sufficiency rather than necessity). THEN and CAUSE are the most useful types, backing up the suggestion that legitimate links may exist between actions, because of their enabling behaviour rather than as a direct causal antecedent to an event.

Rumelhart goes on to suggest that the implementation of such a scheme may well involve 'bottom-up' structures known as *Scripts*, *Plans* and *Memory Organisation Packets*. Structures like these assist greatly in reducing the ambiguity in language meaning that humans are very good at decoding and machines are not. Yazdani argues that a computer can be made to write a story by taking a series of stock elements implemented as frame structures (also suggested by Thorndyke), allowing their variable role fillers to be bound at a later date. Such a generic structure allows stories to be re-usable, enabling an observer to be reminded of other stories which have similar elements (Schank). Yazdani also distinguishes between the *content* and the *shape* of a story, and contends that a script-driven generator is wholly concerned with the content of the story, ignoring its shape. Story Grammars, in his view, are the opposite, overly concerned with shape and ignorant of independent content. This argument perhaps lends some credence to Rumelhart's opinion that both 'top-down' and 'bottom-up' elements are needed to get the balance right. Yazdani's system, ROALD, creates a database of facts for each character in the story, and includes some universal motivations to act, such as hunger, loneliness and fatigue. Lebowitz's UNIVERSE story-writing program implemented *meta-plots*, units abstracted from the events they described, to try to produce soap opera-style stories. Frames were used to store information on the characters, including details of their offspring and marriages. The model would then produce a story by either the arbitrary creation of events, or by acting as a "writers aid" to a human author.

Simmons & Correira (see also Brewer, Kintsch & Van Dijk, Thorndyke) take the view that a tree-like structure such as that produced by a story grammar, if created properly, could be used to read off a summary of the story from the nodes nearest to the root. Kintsch & Van Dijk further suggest that the focus of a story can be obtained from the production of *coherence graphs* which detail how propositions taken from a story are linked. Such studies have been used to demonstrate the psychological validity of frame structures in recollection experiments. It has been shown (Brewer, Kintsch & Van Dijk, Mandler & Johnson, Thorndyke) that as *plot structure* (such as causal ordering and actor motivation) is removed, comprehension and recollection of the story becomes more difficult, and that if plot structure is present, so-called *micro-propositions* (specific events and characters) are overlooked in favour of more generalised *macro-propositions* when creating summaries of the story. This relationship shows that human memory constructs frame-like structures when reading and comprehending a story.

Reiser disputes the claims that events buried deep within a story hierarchy are less likely to be recalled, pointing out that readers generally identify with a single character through a story and are therefore more likely to remember events initiated by that character. Such an approach also helps control the possible combinatorial explosion of inferences to be expected if the goals and plans of all characters are analysed at once.

Kintsch and Van Dijk have put forward four *macro-rules* explaining how micro-structures are to be related to macro-structures; *deletion* (of propositions that detail 'an accidental property of a discourse referent'), *generalisation* (by substituting in 'a proposition defining the immediate super-concept of the micro-propositions'), *selection* (of a single proposition from a sequence of which all the others are simply conditions or components of) and *construction* (of a macro-proposition that replaces a sequence if the sequence consists of conditions or components of the macro-proposition).

Interestingly, Mandler & Johnson also mention this replacement algorithm, terming it *transformation* of well-formed stories. They deem such transformations necessary unless a well-formed story is to be defined as any that corresponds exactly to a story grammar. There clearly exist stories that are well-formed and breach this condition; consider for instance a story told in flashback. However, they are rather more cautious when defining the terms under which propositions can be removed from a story whilst maintaining the story's well-formedness.

Bales (see Kosaka) developed a series of categories to describe events, and proposed a pseudo-grammar that said that 'the next act of the next other is a reaction to the last act of the last other'. Whether or not anything as formal as a global semantic model of a story is created, the structures Rumelhart mentions can be useful, and are described below. A method attempting to find the global structure of a story in these terms may well suffer from the same problems as syntactically-based parsers, in that completeness is very hard to ensure without a huge and largely redundant structure.

Song & Cohen also use focal points of a story in an attempt to correctly order the events that occur within the narrative. Because the English language permits tense structures which do not always provide clues to the order in which a series of clauses are meant to be taken, a stack of local *temporal foci* are built up as analysis continues. When a contentious event structure is created, the recursion unwinds until the event representation's situational information (such as actors, location, and so on) matches with that of its surroundings.

From an actor's point of view, plans are an attempt to construct a method by which an actor in one state can achieve a particular goal. The problem facing this project is more akin to the opposite process; working out what an actor is doing given a series of his / her actions, and deciding how they are linked together; this phenomena is known as *explanation*.

Algorithms used in *Explainers* fall traditionally into one of two camps :

i) '*chain and control*' architectures, which apply large databases of inference rules (Wilensky) to build an explanation. The disadvantage of this method is that each observation is treated as being novel, and therefore such algorithms perform computationally expensive analyses of the data each time such data is presented. They have no knowledge of stereotypical situations to speed up the analysis of mundane data. Luria's question-answering system implemented this method, associating with each causal step in a path an idea of its importance. Thus given the example "*Sarah bought a maths book. She did well in her maths exam*", the answer to the question "*Why did Sarah buy the book?*" could be "*to own it*" or "*to read it*", but is probably "*to study maths*".

ii) So-called *Script Application* techniques work on the assumption that sufficient structures (implemented as scripts, Memory Organisation Packets [see below], plans or frames) to explain the observations reside in memory, and the problem lies in organising these structures to enable the correct example to be retrieved (DeJong, Schank, Dyer, Kass). This methodology suffers because a heavy reliance is placed on the existence of a suitable structure to explain exactly the observations being analysed. If the matching algorithm is to be inference-free, the structures themselves need to be highly specified, and necessarily stereotypical; the drawback lies therefore in the analysis of non-mundane data. *Plan Adaption* techniques have been applied to this problem (see below).

Plans are not unique to artificial intelligence problems. Shweder notes that 'normal adults can count on context, and a shared body of knowledge, beliefs and presuppositions...to contribute tacitly whatever information is required to make their utterances comprehensible.' A methodology conceptually parallel to plans exists in engineering, under the guise of *Functional Representation* (Forbus, Iwasaki *et al*). A system can be decomposed into a series of *model fragments*, each of which describes a phenomena and the conditions under which such a phenomena could be expected to appear. By combining these model fragments, a pattern of behaviour of the device under question can be built up. Functional Representation can therefore be used to simulate the device's behaviour or diagnose faults from a set of observations (Chandrasekaran, Finin & Morris). Chandrasekaran points out that this logical causal mechanism, based on a set of observations, can be applied with equal validity to questions like :

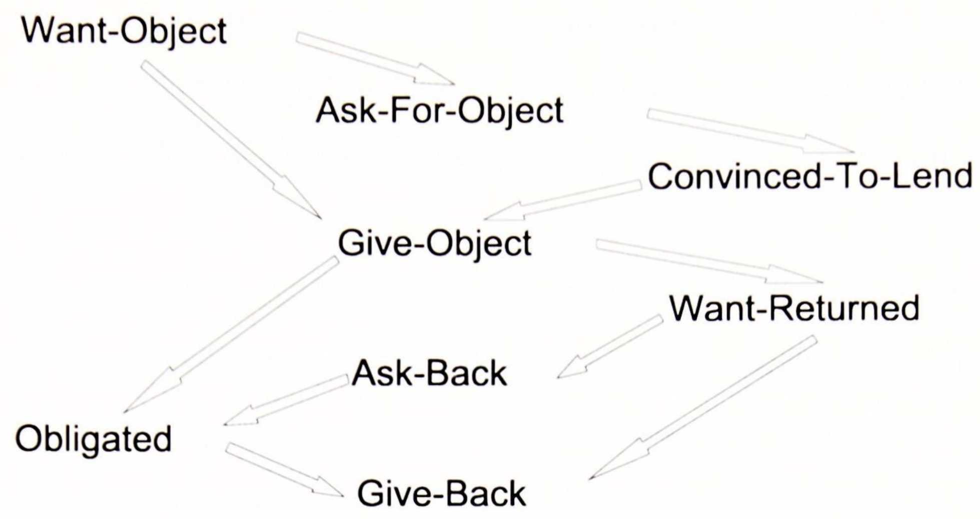
"How does this device work?",

"How do clouds make rain?", and

"How do clouds make rain?".

Such *abductive reasoning* has formed the basis of many planning systems (Schank & Abelson, Wilensky, Kautz, Bandini). Kautz also makes the point that although a set of observations may not uniquely identify a single plan, important information may still be gleaned from them. This is especially true within this project; any robust causal information is useful in the scope of the model. Bandini discusses *fabulae* (plots), which are very similar to the model fragments discussed above. They are constructed of a network of *states* and *transitions* in much the same vein as a transition network, with as much temporal and causal information filled in as is available. He proposes that this network can be segmented to provide the view each actor has of the events they are involved in. This causal chain is the actor's partial view of the narrative.

Once again, frame structures are implemented in numerous guises, although under different names. Schank created *Memory Organisation Packets* (MOPs) for use in case-based reasoning models. MOPs outline the possible steps that may be taken, in terms of events and plans, in order to achieve some objective. A MOP detailing the steps involved in borrowing an implement could be :



Tree structures again store hierarchies of increasingly MOPs relating to a particular story. A MOP may have *threads*, meaning that it can be instantiated from many perspectives; Dyer gives the example of a *restaurant* MOP, which can be called from the *diner / owner* or *diner / waiter* perspectives. Dyer's BORIS program introduces knowledge constructs called *Thematic Abstraction Units*, which "organize cross-contextual episodes which involve similar failures in planning"; for instance a farmer losing a horse, and a professor losing a researcher. TAUs consist of steps abstract from the actual content of the events they describe, concentrating instead on how a sequence of events is linked at this more abstract level. So the TAU-HYPOCRISY is described as :

x is counter-planning against *y*.

x is trying to get *z* to block *y*'s use of plan P-1 by claiming it is unethical.

y claims that *x* used a similar plan P-2.

Therefore *x*'s strategy fails.

Kolodner used MOPs in CYRUS, a question-answering program concerning the activities of diplomat Cyrus Vance. The MOPS used were divided into two categories; *simple MOPs* covered such activities as sM-MEETING, sM-VIPVISIT, sM-TRAVEL and so forth. *Intentional MOPs* were less stereotypical, covering long-range actions with a standard goal -I-NEGOTIATE, I-SOCIALIZE, etc.

The activation of plans is then based on the recognition of intention within actions. Assuming a hierarchical plan structure, activation of a plan given a new piece of information is based on finding the most specific set of plan preconditions that the new information fulfils (Bares *et al*, Vilain, Kautz). Subsequent information is then checked in the first instance against the currently active plans and their specialisations in the plan hierarchy. Laskowski & Hofman suggest a slightly altered version of this algorithm in which any script is activated whose preconditions are fulfilled.

Alternatively, Schank & Abelson attempt to identify the theme within a text to identify the goal the actor is attempting to bring about. Previously used structures (*Explanation Patterns*, or XPs) are then drawn from memory and a series of 'tweaks' (Alterman, Kass, Schank) are applied (based on a *description of the failure*, or XP-FD) to this structure to try to explain the current observation. For instance, given the story (paraphrased from Kass) :

'*A college football player died the day after being selected for the national team.* ', the explanation that '*A jogger who has a heart defect places too much strain on the heart and dies*' fails because the story does not contain a jogger. However, by applying a 'tweak' to change '*jogger*' to '*college football player*', the explanation pattern is perfectly valid, although not unique.

Alterman's tweaking mechanism is based around abstraction and specialisation of the structures used to explain observations. If a problem occurs in the preconditions to an apparently suitable plan, the model abstracts up the plan hierarchy until the problem element is removed, and then specialises back down other branches, searching for a suitable plan.

Wilensky implements a 'shortest-path' algorithm from the new input to an element of the story's representation already in memory. Wilensky's system uses *meta-plans* and *meta-goals* to guide the planner to, for instance *FULFIL_AS_MANY_GOALS_AS_POSSIBLE*, or *AVOID_IMPOSSIBLE_GOALS*.

A similar meta-planning principle is used in the DAYDREAMER package (Mueller & Dyer). Plans are activated on the basis that daydreams are instigated on various grounds (Rationalisation, Revenge, Preparation for the future, and Failure / Success reversal). The subsequent behaviour of the actor can then be explained in these terms. For instance, on being refused a date with a beautiful actress, the actor daydreams that he will one day be in a position to refuse a date from the actress. The planner then seeks to assemble actions which attain this goal.

The dialogue-based system ARGOT (Allen, Frisch & Litman) looked for an inference path from a *linguistic action* to an *expected communication goal*. Such goals are based around a set of expectations of what an actor is likely to say at certain points within a dialogue. If a dialogue opens with the question "*Could you mount a magtape for me?*" then the linguistic action associated with it becomes *User REQUEST that System INFORM User if System can MOUNT tape*. This action is then passed to the plan recogniser which produces 2 potential goals to fulfil :

1) *System INFORM User if System can MOUNT tape* (the literal meaning),

2) *System MOUNT Tape* (the indirect meaning).

Vilain once again implements an algorithm to find the most 'parsimonious' (specialised) plans to explain a sequence of observations. He also noted the similarities between planning and parsing algorithms - that is, a plan 'parses' a sequence of actions to give them structure and contextual meaning.

DeJong, in his FRUMP system, applied three algorithms to choose between candidates when activating structures called *sketchy scripts*. *Keywords* were used so that "*John Doe was arrested....*" will cause the activation of the $ARREST script because of the presence of "*arrest*". *Implicit Reference* is used when there is insufficient evidence to cause the activation of a sketchy script but a script that commonly occurs before it *is* activated - thus FRUMP is told that robberies often precede arrests, so the activation of a $CRIME script will lead to the implicit activation of an $ARREST script. Finally, *event-induced activation* is used when the semantic structure of the source language strongly implies a particular sketchy script; "*Police apprehended John Doe*" is analysed to produce a POLICE_APPREHENSION event which is deemed as centrally occurring in the $ARREST script.

Charniak & Goldman suggest a probabilistic model of plan activation based on the set of observations being passed to the activator; the central problem with statistical methods lies in the difficulty of setting and making consistent the probabilities associated with the observations. The defence for such an outlook in (Charniak & McDermott) is that this *decision theoretic* approach is not for plan *construction*, but plan *evaluation*. However, this simply moves the problem of assessing the probabilities back a stage; the criticism remains valid.

Charniak has alternatively suggested algorithms similar to DeJong's. He addresses the problems of indexing the frame structures comprehensively (a separate structure required to search the frames, and the number of *ad hoc* decisions required). His solution is to internalise all the information necessary, sidestepping the problem of frame-indexing, but introducing the problem of a combinatorial explosion in deductions, including large numbers of false deductions. Suitably integrated frames should combine to suggest the correct candidate, as in "*the man sawed the woman in half.*" The three elements (SAW ACTION1 MAN1 WOMAN1), (MAN MAN1) and (WOMAN WOMAN1) combine to suggest the MAGIC frame. This approach tends to ignore the context of the language, as highlighted by the text fragment "*the man sawed the box in two. there was a woman in it*". Neither action or actor strongly suggests a magic trick being performed, but the inference is clear from the two sentences combined. Charniak acknowledges that a system based around current frames should be able to produce a series of possible meanings which can be gradually disambiguated as more information is supplied.

Another formalism which was proposed was *plot units* (Lehnert), structures which attempted to explain a cluster of events in terms of *affect states*, the motivations of actors - for instance, a helpful act on the part of one actor may help to explain any future reciprocated assistance. In some ways, this methodology was an implementation of Bales's notion (mentioned above) that an action should be viewed "as a response to, the last act of the last other, or as anticipation of the next act of the next other". The scheme attempted to impose structure on stereotypical events, on the level of emotion and motivation than actions themselves, and is subsequently very complex. The results of a study of the ability of plot units to explain the structural influences behind recall in story summarisations was published (Lehnert, Black & Reiser). They showed that plot units are at least as good a model of memory and recall as Story Grammars, despite being a fundamentally different approach. Brewer found something similar. A test of recall of a series of videos of a goal-directed action was done, which showed a high degree of correlation with the tests discussed above. Brewer claimed that this correlation was not in fact due to the story's structure, but in fact due to the underlying plan schema.

Marker-passing inference (Norvig, Quillian, Raphael) is another method which utilises semantic networks. Quillian's TLC (Teachable Language Comprehender) found connections between concepts and from these constructed the relationship which existed between those concepts. Hence the phrase '*the lawyer for the client*' constructed a structure representing the meaning as '*the lawyer is employed by the client for legal matters*'. The system is very simple and doesn't take into account the semantic structure of the sentence. Thus, the system has problems distinguishing between '*the wife of the lawyer*', and '*the enemy of the lawyer*'. Quillian's Semantic Memory system used semantic nets to disambiguate the possible relationship between 2 contexts input by the user :

QUESTION: CRY, COMFORT

A. Intersect: SAD

(1) CRY2 MEANS AMONG OTHER THINGS TO MAKE A SAD SOUND.

(2) TO COMFORT3 CAN MEAN2 TO MAKE2 SOMETHING LESS SAD.

This inference is made by following links between concepts until a marker is reached from both start-points. The meaning of the link is then deduced by examining the paths of links leading to the intersection.

Raphael's SIR system similarly attempted to represent semantic knowledge by use of word-association models and mathematical logic to related concepts together. Trees of semantic concepts are constructed, and rules then applied to describe which concepts can be related to each other. Thus, aeroplanes cannot talk, humans cannot fly, and so on. A semantic network is used to resolve ambiguity in sentences like '*X has Y*'.

Norvig attempts to demonstrate that a marker-passing system can perform the same tasks as script- and plan-based inference models, and can overcome the problems faced by Quillian. '*For*' and similar locally meaningless words are marked as full-blown concepts, and meaning is derived from the path taken to *collide* with another concept.

Given the sentence '*John was eating at a restaurant with Mary*', SAM (Cullingford) fired the *restaurant* script from the '*restaurant*' token and fills in inference from there. Norvig's system (called FAUSTUS) makes inferences by passing markers.

i) Passing markers from the 'EATING' and 'WITH' concepts produces a marker *collision* inferring that '*with Mary*' is supplying a meal *Companion*.

ii) 'EATING' and 'AT RESTAURANT' infers that an *Eat-At-Restaurant* event is occurring.

Norvig's system can also deal with planning-type situations. Given the following story,

i) *John was lost.*

ii) *He pulled over to a farmer.*

iii) *He asked where he was.*

PAM (Wilensky) would infer as follows :

i) John wanted to *know* where he was $\rightarrow$ goal state.

ii) John wants to use the farmer, and therefore be near him.

iii) To *Ask* is a plan for knowing and therefore a connection exists between sentences i) and iii).

FAUSTUS infers :

i) Nothing from this sentence.

ii) the pronominal reference as no other actor exists.

iii) Nearness is a precondition of asking something; asking is a plan for knowing.

Various problems are inherent to approaches such as these. Mandler & Johnson mention the difficulty of identifying a goal from an utterance without the specific desire of the actor to communicate that goal. If no goal is explicitly mentioned, they contend that a listener has to analyse the following actions to determine if a relevant goal exists. Jacobs & Rau note that the difficulty in finding the motivations of actors in character-driven stories has given way somewhat to the study of more transparent and declarative texts, such as newspaper stories, in which rationality and causal flow is easier to find. This is partly because of the desire for researchers to actually put their tools to work on large bodies of text, and inevitably compromises need to be made. Charniak's Ms. Malaprop system ignored the problem of frame recognition altogether, and relied on the explicit statement of goals and plans in order to select the correct structure to explain the behaviour of an actor. Suchman notes that the necessary formalisation of background or commonsense knowledge for use in models presupposes that at some point, the researcher is prepared to say that some intelligence is simply implicit, because this process of formalisation can continue indefinitely. The researcher has to decide for him/herself at what point information can be said to be *basic*, by Abell's definition.

CHAPTER TWO :
THE THEORY OF
COMPARATIVE NARRATIVES

Abell's theory of Comparative Narratives

1. Introduction

In order to construct a theory about a social event, its precursors and its outcomes, it is necessary to assume that the social world (ie the space in which we interact with other people) can be viewed as a web of inter-related actions and forebearances. Assuming also that it is in some way possible to identify the causal connections, it becomes feasible to trace backwards through this web and make a decision concerning the forerunners of the event of interest.

Repeated performance of this operation will lead to the construction of a *narrative*, which will describe a set of actions and their relations with each other, in terms of consequences and antecedents.

To explain an event which holds some social significance to us, it is necessary to first explain the events that brought it about, and subsequently to examine *these* events for their origins. Assuming that at some point we are able to pinpoint no further significant precursors (or more realistically have a facility enabling the process to stop meaningfully), we are then in a position to explain the generation of the social event. By comparing many such social explanations of apparently identical / different events, the process makes possible the comparison of multiple narratives; thus *Comparative Narratives*.

The material presented in this chapter is based exclusively on two works by Abell, '*The Syntax of Social Life : The Theory and Method of Comparative Narratives*', and '*The Theory and Method of Comparative Narratives*'.

1.1 Basic actions

The nature of cause and effect and our imperfect knowledge of the world mean that any action that we may take is likely to have any number of unforeseen and possibly undesirable outcomes. Regardless of whether the actor in question actually intended any of them, it is essentially true to say that the actor did "perform" all of them.

The same effect works in a slightly different way going backwards in time. The apparently holistic nature of the social world means that it is very difficult to say with any degree of certainty whether all the precursors of a particular event have been found. The further back in time we go the harder this procedure becomes, as even written accounts of an event are open to interpretation and revision.

In order to stop this potential information overload swamping any attempts at narrative construction, it is necessary to set a criteria by which the recursive process of finding precedents will terminate. *Basic* acts, as defined by Abell, are actions that can be said to have been performed without any previous significant action having taken place. By the same token, *non-basic* actions require some form of preparation, taking the form of (chronologically speaking) earlier actions.

It is possible in virtually all cases to describe an action in a variety of different ways. For instance, "X switched on the light" is simply a more succinct phrasing of "X flicked the switch, causing current to flow which illuminated the bulb". In terms of the story being told in the first sentence, the act of turning on the light appears to be basic, but on a more general level, a light being illuminated would be caused (and therefore not be basic) by darkness. The first action in the second story (flicking the switch) is also basic, whilst the two consequences clearly are not.

1.2 Notation

Abell uses the following notation to categorise outcomes of an individual's actions and forebearances, which take place within the larger situational context *C* - that is, the conditions and context under which human action takes place.

$X \lfloor Y$: Event X leads to event Y

αI_o : α intended o

αD_{Io} : α intentionally did o

αD_o : α brought about o

αP_{Io} : α intentionally prevented o

αP_o : α brought about the prevention of o

Thus it can also be said that

$$\alpha D_{Io} \equiv \alpha I_o + o$$

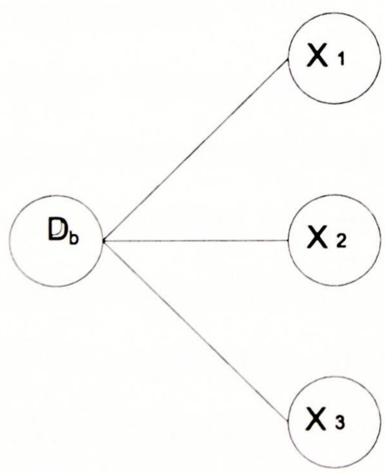
and equivalently,

$\alpha P_{Io} \equiv \alpha I_o$ + _o, where _o is the non-occurrence of o.

A similar notation can be used to account for forebearances, using _D and _P.

This notation is also of assistance in comparing the degree to which events are basic. αD_o is more basic if the effect this action introduces causes the effect introduced by $\alpha D_o'$. Although it is true that the degree of basicness decreases along a causal path, it is impossible to compare the basicness of a group of events which are all outcomes of a single event.

1. Assume αD_{Io} is true.
2. Assume αD_{Io} entails αD_b, where αD_b is a basic act.
3. Let $\alpha D x_i$, $x_i \in X$, where X is all states for which αD_b is a sufficient condition.
4. Causal relations will define the structure on the set of actions $(X \cup b \cup o)$.
5. The resultant digraph (if any) will be weakly connected and acyclic.
6. If X1 ⌊ X2, X2 ⌊ X3, αD_{X3} is less basic than αD_{X2} is less basic than αD_{X1}.
7. Basicness decreases along a path.
8. It is not possible to compare basicness of each set out of αD_b.



It is not possible, given this structure denoting the causality of a narrative, to say whether X_1 is more or less basic than X_2 or X_3.

A central problem to be addressed in identifying this flow of causality is in distinguishing actions from a continual stream of heavily interdependent activity, not readily divisible into useful pseudo-independent actions.

1.3 Dependency

Abell highlights three kinds of dependency between actions.

1) *Preparatory*. αD_{Io2} implies that αD_{Io} was previously successful. For instance, eating a sandwich is dependent upon successfully making it.

2) *Normatively dependent*. A series of actions can be redescribed as one complex action. The Chinese Tea Ceremony is an instance of this type of relation.

3) *Causal*. α performs an action which later influences them without being molecular in structure (ie part of a normatively dependent series of actions) or consciously preparatory.

2. Local explanation of actions

In situation *C* α *intended* that o;

In situation *C* A *believed* that if $\alpha D_I x$ then o would result;

therefore in *C* α *intended* x;

and $\alpha D_I x$;

and $\alpha D_I o$;

and αD_y (where y is the set of events causally related to x).

2.1 Generalisation

If the above scheme is accepted as the reason for α doing x, then generalisation is not a necessary component of the local explanation of the event. The explanation of this action is therefore not bound to a generalisation, except in terms of a sequence of actions that appear to be causally related. If two elements are causally related, the explanation of this causal link will require mention of a generalisation. The explanation of what happened is still valid. If the outcome is unexpected, the fault lies in imperfect knowledge of the system the actor is immersed in. For instance, if α attempts to bring outcome o about via action X and o doesn't result, it could be because α was incorrect in the belief that X causes o, rather than because the scheme is in some way flawed.

2.2 Intention

Intention implies action, not the other way around; it is perfectly feasible to perform an action without meaning to. All actions (intentional and otherwise) can be said to be consequential to a deliberate action. Intention can be divided into two parts;

1) *Deliberative intention.* There is no need for this to be tied to a specific action. An actor may deliberately intend to do something, and then fail to do it for some other reason.

2) *Intention intrinsic to action.* This form of intention is related to actions that could be considered habitual, although not reflex - "find yourself doing something". The action is essentially self-justifying, as part of a larger, possibly lifestyle, pattern.

The point to note is that it is important to distinguish between a goal-directed intentional action and an unintended consequence of some action. Although the effect with regards to causality is the same, from the point of view of a single actor the meaning may well be different. In a complex multi-actor narrative, the effect is blurred somewhat, as a high degree of control is wrested from the hands of the actor and they are at the mercy of others' intentional actions and unintended consequences.

Forbearance has an interesting bearing on the subject of intentional actions, because if such forbearance is intentional, an action is deliberately not being taken.

When α takes an action, there are two statements that can be made:

1) α *de facto* forbears to take all other actions in α's power.

2) α intentionally forbears to take all other actions α believes α could take.

Intentional forbearance is important to a narrative's structure, because it is explicitly an action. Other actors will (forbear to) perform actions because of someone else has not done something. Inactivity is, in a multi-actor scenario, an activity.

The utility approach to decision-making means that a (rational) actor will consider their whole series of possible actions and choose that which yields the highest utility on their terms. When an actor is acting in a manner which could be considered habitual, it is difficult to say whether they have considered all of their potential options, ie is such forbearance strictly intentional?

It is not possible to explain why α forbore some actions because α may well be operating with imperfect information and therefore be unaware of all of their options. To explain such forbearance requires an explanation as to why α was in a position of imperfect information to start with. Any explanation of the choice that α made is not logically equivalent to an explanation of the action itself.

2.3 Cultural assumptions

Narratives under analysis will generally consist of complex natural language descriptions of a particular environment. Such an environment will contain sets of descriptions and assumptions that are *locally meaningful*, causing actors to recognise features of the environment in question as integral to the process of decision making. *Social interaction* is a clear example of this - the actions of other actors in a particular scenario will cause α to behave in ways that are perhaps not generalisable to other situations.

Assumptions that are applied unconsciously every day are similar in effect to habitual forbearance, in that they are only brought considered on a conscious level when justification for their existence is required. They assist in applying context to the undertaken action and its consequences. For this reason it is possible for an analyst to expand on the contextual information available at the time to the actors involved in order to explain behaviour which appears to be in some way aberrant or misconceived.

On a slightly more analytical level, an action can be evaluated locally in an attempt to discover any of the following:

1) *Why actor α intended outcome o to start with.* In order to explain an action, it is potentially useful to ask why α valued a particular state o over all other potential outcome states that α is aware of.

2) *Why α believed action X would lead to o*, whether in reality it does or not. Because actors are likely to be dealing with imperfect information, the accounts they will give will be partial and therefore it is important to consider their *pattern of reasoning*. This is definitely a non-trivial task. In a real-world situation, quantitative methods may be of less use because of the difficulty in highlighting correlations

from relatively nebulous concepts like ideas, goals and beliefs.

3) The *rationale behind choosing the direction represented by X*, over any other existing alternative actions. Decision theory has been applied in many forms to the problem of ranking a series of alternatives in terms of their utility. Subjectivity, or more accurately irrationality induced by imperfect information, makes such procedures very frail, because of their implicit (and explicit) assumptions, and the form of data required for the mechanics of their operations.

3. Social interactions

It is a truism to say that actors do not act in isolation; they wield influence over each other via the things they say, do, or forbear from. If α brings about an action by β, β's action is said to exist in the *field* of α's, possibly as a step on the way to some unspecified objective of α. This is not true in all cases, as α may have had some *unintentional influence* over β, which α may not be aware of and would therefore be unable to make such contingency plans for.

3.1 Compact model

$\alpha D_I[\ \beta D_I[\ \gamma D_{Io}\ ]\ ]$ means $[\alpha D_{IX}] \rightarrow [\beta D_{IY}] \rightarrow [\gamma D_{Io}]$

where $\rightarrow$ means 'leads to'.

In this model, it is important that α performs X; this conjunction of actor and action leads to β performing Y, not just X occurring. It is possible for multiple actions to be required in order for a particular action to take place;

$[\alpha D_{Ix}]$ and $[\beta D_{Ii}]$ and $[\ [\alpha D_{Ix}]$ and $[\beta D_{Iy}]\ ] \rightarrow [\gamma D_{Io}]$

Note that β's compliance could occur because of unintentional side-effects, even though α intended to procure β's help.

So-called *molecular actions* exhibit the same behaviour. They are a set of single actions that are linked in some way; each could have a local explanation, but the global motivation behind each one is the same motivation. As a whole, they exhibit a form of social interaction between themselves, even if the interactions are wholly unintentional.

What the compact model fails to outline is why one event leads to another at all. Actions and consequences are a large part of the contextual information used to form C. In general, β does / forbears from an action because of action X, where X is either α's action or some consequence of an action by α. The conjunction of the situation C and action X is enough to influence β. From this, α has three methods of influencing β's behaviour in such a way as to assist α:

1) Altering β's *objectives* (values). The easiest way to change (or maintain in the face of potential change) β's objectives is to alter their pattern of *values*. β has a perceived set of objectives, ranked under some criteria. At this point, α intervenes to change β's perception of the desirability of these outcomes. Such *affective socialisation* can happen during sequences of interactions - a whole series may be required by α in order to influence β to change.

The theory of Comparative Narratives itself is interested in value changes *only as far as they alter actions* and their associated outcomes. The partial and dynamic nature of application of the theory doesn't encourage the study of attributes and their correlations.

2) Altering β's *perceived beliefs*. Such alterations will tend to fall into one of three general categories.

2.1) α may have *power* over β. β feels threatened and acts to prevent α exercising their perceived ability to act with negative utility to β. α's ability to threaten is in contrast with the use of force - α could physically stop β doing anything (by killing β) or reduce the range of potential actions that β can take. In order for α to successfully exercise power over β, β must *recognise the threat*, otherwise the subsequent actions of β will continue as if α never intended to influence β in this way at all. β may anticipate the threat before it becomes explicit and comply with α's wishes.

2.2) α may make the *promise* of sanctions of positive utility to β. In most respects, promises and threats are very similar, differing only in β's perception of α's initial intentional action.

2.3) α may *influence* β by making β believe that if β performs X, Y will naturally follow, where α has no control over Y. The reverse of this particular relation is *manipulation*, where α knows that Y will not in fact follow from X.

If α's ability to exercise power or influence over β is seen as legitimate then threats become superfluous. α has the right to control β's actions.

3) *Causally* prompting β.

4. Accounts of actions

Potential sources of input data on which to run the theory are numerous - documentaries, interviews, observational data. The creation of a coherent narrative is in the hands of the analyst, although care is required to preserve the actor's views. It is impossible to say whether an actor is telling the truth about their own actions or those of any other actors involved. Because of this, it should be borne in mind that the process of forming a coherent account (or indeed not forming any account at all) is an action in itself, on the part of the analyst who is basically giving an account of an account.

An account of an action comprises a sequence of events that the analyst is required to put down on paper, *true in spirit* to the events as specifically related. Elicitation of the true meaning of an account is generally a problematic task, because there can be no such thing as a definitive interpretation - each retelling of an account may dilute the original material whilst introducing new information from the narrator's point of view.

There is no such thing as social life without inferences in the form of *shared meanings*. Since accounts of an action are essentially inferential representations of that action, they are basically no less valid as a form of data than any other. Since every account is treated as value-free and self-justifying, they are never strictly false. Having said that, the relationship between an account and an action is based on conjecture, and therefore it is important that testing takes place under many different circumstances, taking in as many viewpoints as is feasible.

5. Narratives

The theory seeks to explain an event in terms of the generations of events that culminate in its existence. A narrative is generated accounting for the outcome's occurrence. There are two versions of a narrative.

5.1 Compact narratives

A compact narrative comprises a *finite set of actions A* leading to the local explanation of the outcome, a set of *actors*, *weak time ordering* (permitting concurrency) and a *mapping* of actors onto actions. The narrative contains a generalised asymmetric non-cyclic 'leads to' relation on A $\times$ A. The meaning contained within the narrative takes two forms:

1) *Local* meaning in *each element* of A,

2) *Global* meaning in terms of *connectivity* generated by $\lfloor$.

5.2 Extended narratives

Extended narratives distinguish between:

1) *Intentional premises* (set I),

2) *Cognitive premises* (set B),

3) *Conditions* of action (set C),

4) *Consequences* of action (set O).

These four distinct types of action replace A in the compact model above, although their union can be referred to as A. An extended narrative contains a *directed graph* of A and an *A* $\times$*A matrix* formed from 2 copies of the vector A. The elements of A form the nodes of the digraph, the arcs are formed from 'leads to' nodes. The matrix entries are defined as follows:

$M(i, j) = 1$ if $A(i) \lfloor A(j)$,

$M(i, j) = 0$ otherwise.

The extended model also includes a *narrative graph* to cover the temporal aspect of the actions. Each actor is represented by an individual row, and *time* provides the columns. A particular actor at a particular time is then represented on a graph. The points in the table abstractly represent the actors' actions.

5.3 Notes on constructing tables

1) A *narrative table* is constructed from extensive interviews (as an example

of suitable input data) with the actors involved. The analyst combines information from the input data with more general *background reasoning* to fill in any potential holes in logic. The constructed narrative is therefore hypothetical and open to revision if more information comes to light.

2) The *level of abstraction* desired by the analyst is adjustable at this point. If more local detail is required, then multiple interviews can be carried out with members of each organisation in question.

3) It is open to question which actions are pertinent to the narrative. In the case of conflict, the analyst should generally proceed with whichever event *causes the next action*.

4a) Direct quotes from actors are placed in quotation marks.

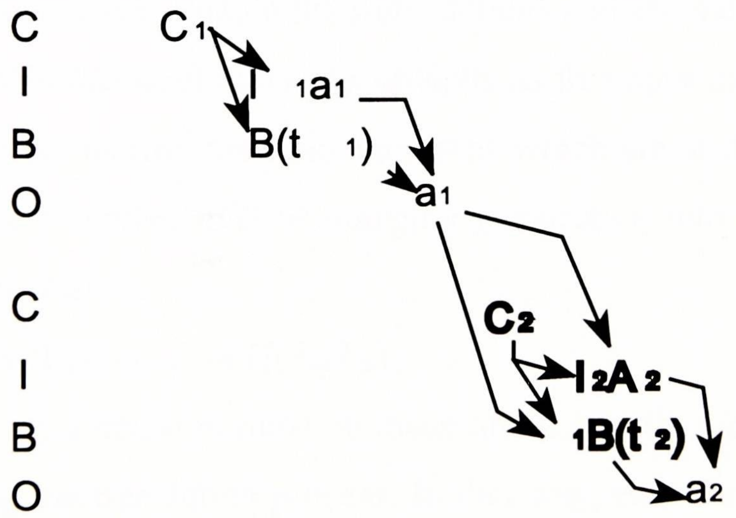
4b) The analyst's hypothetical surmises are placed in brackets.

4c) In the face of *competing accounts*, both should be placed in the table.

5) In the event of actors working both in groups and alone, *joint actions* should be portrayed as if each actor performed it.

6) Each actor is depicted with the sets C, I, B, O being displayed individually.

An example;



C_i : Conditions of action at time t_i

$B(t_i)$: Beliefs at time t_i

Having reached this stage, it is important to attempt to *simplify the narratives*, by attempting to group actions with the same objective, commonly called *molecular actions*.

6. Aspects of structure

1) *The level of abstraction is variable.* As it increases, the level of local detail declines, the limit to this relationship being the point at which all actions are said to be basic. If this point is reached, the narrative is said to be *maximally fine*.

2) The elements forming the structure should be *weakly connected*, which is to say that a path of connectivity should exist between all pairs of points, no matter how circuitous. If the structure does not fulfil this condition, it represents more than one narrative.

3) One-actor narratives (*biographies*) are possible, and are treated exactly the same. 4) The structure *cannot contain 'OR' relations.*

5) If the grounds for any action (in the extended model) are purely contextual (that is, belonging to set C) *this action is the start point* of the narrative.

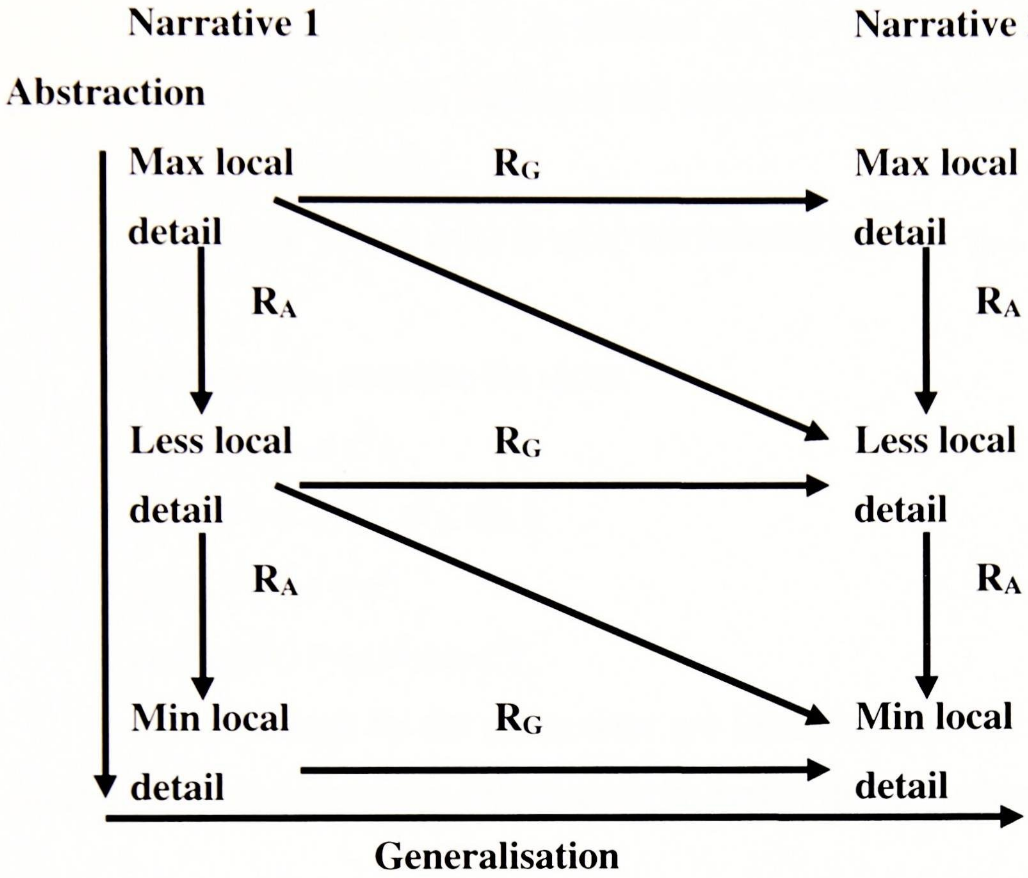
7. Generalisation and abstraction

Comparative Narratives enables the questions "Is [event] O a recurring social event? Is O generated by similar actions?" to be asked. In order to answer these questions, a theory of the *similarity of narratives* is needed.

To be able to say that two narratives are identical, two conditions must be fulfilled. First each narrative must contain the *same actions* and second, these actions must be *inter-connected in identical ways.* As obvious as this appears, what is also required is some way of comparing multiple narratives which are at different levels of abstraction. Therefore a method (G) of mapping a narrative into more abstract narratives is needed, such that

$G(A_b:L_b) \rightarrow G(A_1:L_1) \rightarrow .... \rightarrow G(A_n:L_n)$

When generalising, a decision must be made about exactly what is kept and what is discarded during the translation process. In this way, two narratives can be abstracted to generate the equivalence which fulfils the conditions needed to be able to say the narratives are now identical. Diagrammatically this can be represented as the following :



7.1 Translation of narratives

In order to formulate a series of rules to translate narratives, the following information is required.

1) Assume two narratives $G(A{:}L_1)$, $G(C{:}L_2)$ exist.

2) Assume C is a *partition* of A, a set of mutually exclusive and exhaustive subsets of A.

3) An *identified pair* is a pair of actions in the same abstract equivalence class in C.

4) Assume two actions a_i and a_j map into the same equivalence class in C. Then *all actions on all paths between a_i and a_j map into C* as well.

7.2 One actor partial homomorphisms

1) $(a^k_i * a^k_j) = (a^k_j * a^k_i)$ = *the set of all actions by k between a_i and a_j inclusive*, on all paths.

2) $(a^k_i * a^l_j) = (a^l_j * a^k_i) = a_i, a_j$ *iff $\exists$ at least one path from a_i to a_j*. If $k \neq l$, the operation does not include any actions on paths between a^k_i and a^l_j.

3) $(a_i * a_j) = (a_j * a_i) = 0$ *iff no path exists between a_i and a_j or vice versa.*

A similar operation $\oplus$ is defined on C. Therefore we have the multi-groupoids $g(A^*)$ and $g(c\oplus)$, leading to the partial homomorphism :

$\psi : g(A^*) \rightarrow g(c\oplus)$,

where $\psi(a_i * a_j) \subseteq \psi(a_i) \oplus \psi(a_j)$ are indexed by $\alpha, \beta, \gamma, \ldots$

for example, consider the series

$a^{\alpha}_1 \rightarrow a^{\alpha}_2 \rightarrow a^{\alpha}_3$.

$a^{\alpha}_1 * a^{\alpha}_3 = \{ a^{\alpha}_1, a^{\alpha}_2, a^{\alpha}_3 \}$

$\psi(a^{\alpha}_1 * a^{\alpha}_3) = c^{\alpha}$,

and $\psi(a^{\alpha}_1) * \psi(a^{\alpha}_3) = c^{\alpha}$.

If two actions by the same actor are *identified* (termed as being part of the same thing), *all intervening actions* must be part of the same thing.

7.3 Interactive homomorphisms

In order to deal with narratives that multiple actor scenarios are going to produce, it is necessary to extend the abstraction process to include sequences of events in which more than one actor is involved.

$G(A{:}L_1)$, $G(C{:}L_2)$ define a binary operation on A such that

1) $(a_i \bullet a_j) = (a_j \bullet a_i)$, where $a_i \bullet a_j$ represents the set of all actions by all actors between a_i and a_j.

2) A parallel definition exists on C, represented by $\otimes$.

3) 1 and 2 produce the following homomorphism :

$\varepsilon : g(A\bullet) \rightarrow g(C\otimes)$,

where $\varepsilon(a_i \bullet a_j) = \varepsilon(a_i) \otimes \varepsilon(a_j)$.

If any two action by the same actor, a_i and a_j, are put into the same equivalence class, then all actions performed by any actor on all paths between a_i and a_j also map into the same class.

For example the series

$a^{\alpha}_1 \rightarrow a^{\beta}_2 \rightarrow a^{\alpha}_3 \rightarrow a^{\beta}_4 \rightarrow a^{\alpha}_5 \rightarrow a^{\beta}_6 \rightarrow a^{\alpha}_7 \rightarrow a^{\beta}_8$

can feasibly reduce to

1) $a^{\alpha}_{1}, a^{\alpha}_{3}, a^{\alpha}_{5}, a^{\alpha}_{7} \equiv c^{\alpha}$

$a^{\beta}_{2}, a^{\beta}_{4}, a^{\beta}_{6}, a^{\beta}_{8} \equiv c^{\beta}$

where $c^{\alpha} \rightarrow c^{\beta}$.

2) $a^{\alpha}_{1}, a^{\beta}_{2}, a^{\alpha}_{3}, a^{\beta}_{4}, a^{\alpha}_{5}, a^{\beta}_{6}, a^{\alpha}_{7}, a^{\beta}_{8} \equiv c^{1}$

3) $a^{\alpha}_{1}, a^{\beta}_{2}, a^{\alpha}_{3} \equiv c^{1}$

$a^{\beta}_{4}, a^{\alpha}_{5}, a^{\beta}_{6} \equiv c^{2}$

$a^{\alpha}_{7}, a^{\beta}_{8} \equiv c^{3}$.

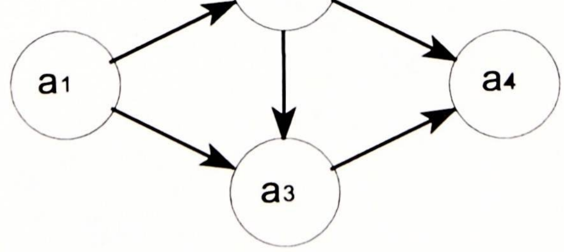
Note that he final example abstracts across individual actor boundaries. An example of a general narrative structure that example 3) could represent would be

Meet $\rightarrow$ Negotiate $\rightarrow$ Split up.

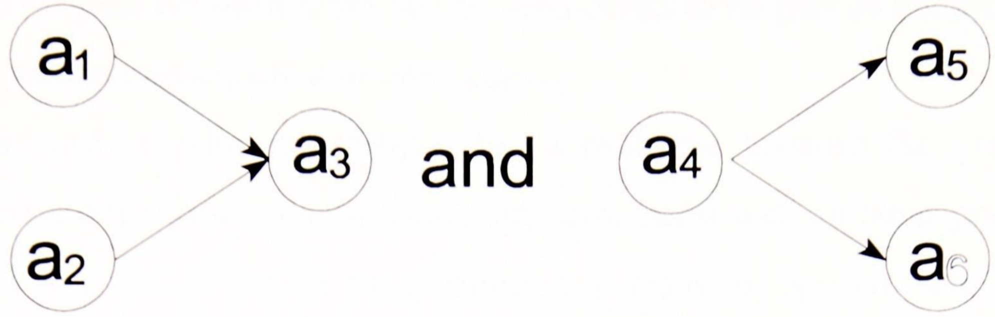
7.4 Notes on generalising

In order to activate the abstraction mechanism, the actions to be abstracted need to fulfil one condition, that the subgraph representing a_i and a_j and all events on all paths in between must be *all path closed* - the set of actions in the subgraph must contain all actions comprising all pairs of products. In more general terms, this means a path must exist between all pairs of points contained within the subgraph.

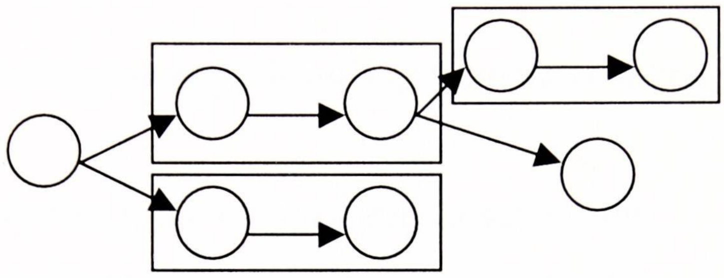
For example, the following subgraph is structurally capable of reduction to a single point, is the analyst feels that a suitable summarisation of the events in question can be produced.



If for example a_2 did not lead to a_3, the structure would not reduce, because there would be neither a direct nor indirect route between a_2 and a_3. Hence,

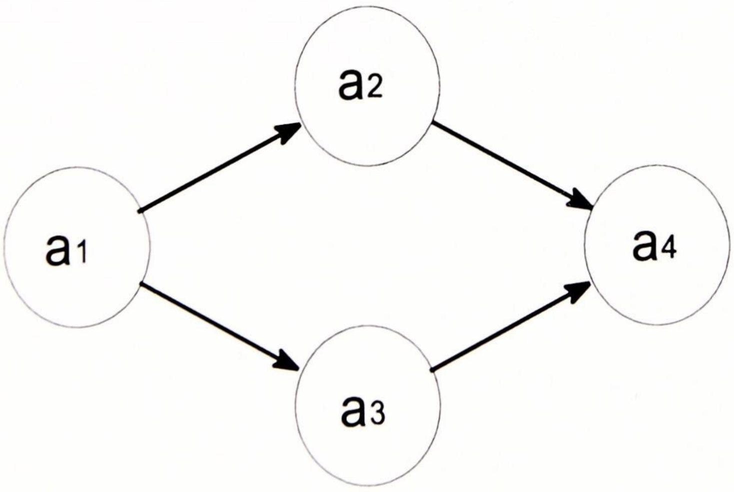


are irreducible. The following structure is irreducible any further if the indicated abstraction has been performed.



7.5 Further partial homomorphisms

A possible problem arises from structures which are *almost* all-path closed, with perhaps only one link missing, within which the analyst wishes to identify all the nodes as part of the same greater structure and therefore reduce them to a single point. Given the present homomorphisms this is not possible if no extra link can be justifed.



Such a structure is displayed above. To cover such an eventuality, a p-homomorphism θ is defined such that

1) θ : ($a_i\ \theta\ a_j$) is the set of all semi-parts (joined, ignoring direction); that is

two actions with no path between them can be considered to be part of the same set *if they share a common consequence or antecedent.*

However, such a p-homomorphism has a problem, because the graph is weakly connected and therefore all points on it are semi-paths with all other points.

Therefore θ requires a degree of redefinition, possibly by applying one of the following restraining conditions :

1) Use only the *shortest* semi-path between a_i and a_j.

2) Include only those semi-paths involving *temporally antecedent* actions to both a_i and a_j.

3) Include only those semi-paths involving *temporally subsequent* actions to both a_i and a_j.

4) Use only the *shortest temporally antecedent* semi-paths.

5) Use only the *shortest temporally subsequent* semi-paths.

Having concluded the description of Abell's theory itself, the following chapter deals with the immediate concerns raised by attempting to computerise the theory of Comparative Narratives.

CHAPTER THREE :
EVENT REPRESENTATION AND CAUSAL LINKING

Event Representation and Causal Linking

1. Introduction

This chapter addresses two crucial areas : i) the disambiguation of a piece of text and its subsequent representation, and ii) the grounds on which causal links are made.

In order to explain in more detail how event representations are created and how ambiguity is reduced and eventually resolved, the debug output from one of the example texts in the results chapter (beginning on page 96) will be explained. The full text is :

"*the body of a boy has been found in a gutted garage. he had been sleeping rough with friends as an outdoor adventure. the boy is believed to have died from smoke inhalation after a fire started in the garage where the boys were using candles. the boy had told his parents that he was staying with a friend.*"

2. Creating event representations

A natural-language description of an event throws up the problem not only of working out what the narrator is talking about, but how to resolve the issue if more than one interpretation of an utterance is possible. A system is required that will help the model to make this decision. First, a brief outline of the structures used, such as event representations and plans, is required.

2.1 Event Representations

These event representations owe much to those in Schank's *Conceptual Dependency*, and other case grammar systems in that they define their own internal structures according to their needs, rather than having a structure imposed on them. The created event is called upon to analyse itself, to fill in roles from the information supplied in the clause. This self-analysis is done in one of two ways.

i) The set of 'canned' (that is, internal) event representations are called upon to construct their own internal structure.

ii) The conditions that help to define user-specified event representations are created at the same time as the event representaion itself, and stored with it.

2.2 Plans

Almost all human beings are sophisticated enough to be able to adapt and deal with unknown situations by breaking them down into perhaps more familiar units that can be tackled individually. They can be said to be *planning* their behaviour to bring about a goal they are interested in.

Plans address the central drawback of scripts (that of excessive mundanity) by applying the principle behind scripts (inferring causality from typical patterns of behaviour) on a more general plane. Plans try to allow for longer-term planning on the parts of the actor, thereby coping with apparent leaps in logic which may in reality be a step in an obscure plan to execute some goal or achieve some state of the world the actor deems desirable to themselves. While they still cover commonplace situations, they should not be as linear or as tied to individual sequences of actions as scripts.

2.2.1 Plan structure

Because of the similarity between the functions of scripts and plans (the difference is only on the level of abstraction), they traditionally exhibit superficially comparable structures. The implemented versions are different because of the divergence of the scripts from a standard structure, but the plans in the model have remained traditional. All plans, regardless of type, have three slots and one globally important marker. The slots are:

A list of *Events* that form the nodes in the plan, along with a list of the *Terminators* that represent those nodes at which the plan can be said to have been completed. For instance, the *ProposePlan* has only one terminator, that of a proposition actually being made.

The third and final universal slot is a matrix, *Conns*, containing explicit causality information between the nodes. The flag *Marker* represents the node in the plan that has most recently been reached.

As events are created by the analyser step of the model, they are passed through all the activated plans. A plan that is activated for the first time also requires that its conditions are set up to reflect the circumstances of this particular instantiation. The practical effect of this is confined to filling roles in the nodes within the plan. The marker flag is used to test a series of conditions (in very similar vein to Augmented Transition Networks) associated with the arcs leading away from the current node.

The plans are specialised (and therefore differentiated from each other) by the setting of roles. An *AgreePlan* contains those separate parties who are trying to resolve an argument, a *WantPlan* identifies the actor who desires something, the owner of that item, and the object of desire itself. Using these actors, plan roles and subsequently conditions can be fixed.

The *connectivity information* is stored inherently in the very skeleton of the plan. The only extra requirement is that the route taken through the plan is recorded, a simple task. When the time comes, the causal route through the plan is taken, and the links transferred into the nascent narrative graph.

2.2.2 Plan Activation

A central problem in the implementation of any planning mechanism is the process by which plans are recognised and activated. Many approaches have been suggested, as described on page 10pp. The approach taken in this project is based on Mandler & Johnson's observation that unless an actor is trying to explicitly communicate his planning behaviour to the observer, active plans are best inferred from the actions that the actor does unambiguously perform. DeJong's FRUMP program approached the problem from a similar perspective, activating so-called *sketchy scripts* on 3 grounds : *keywords*, *implicit reference*, and *event-induced activation*. Keywords were not used in the model because of their inflexibility, event-induced activation was the central method of activating plans and implicit reference was used if an event failed to induce an activation.

A plan is activated based on the following algorithm :

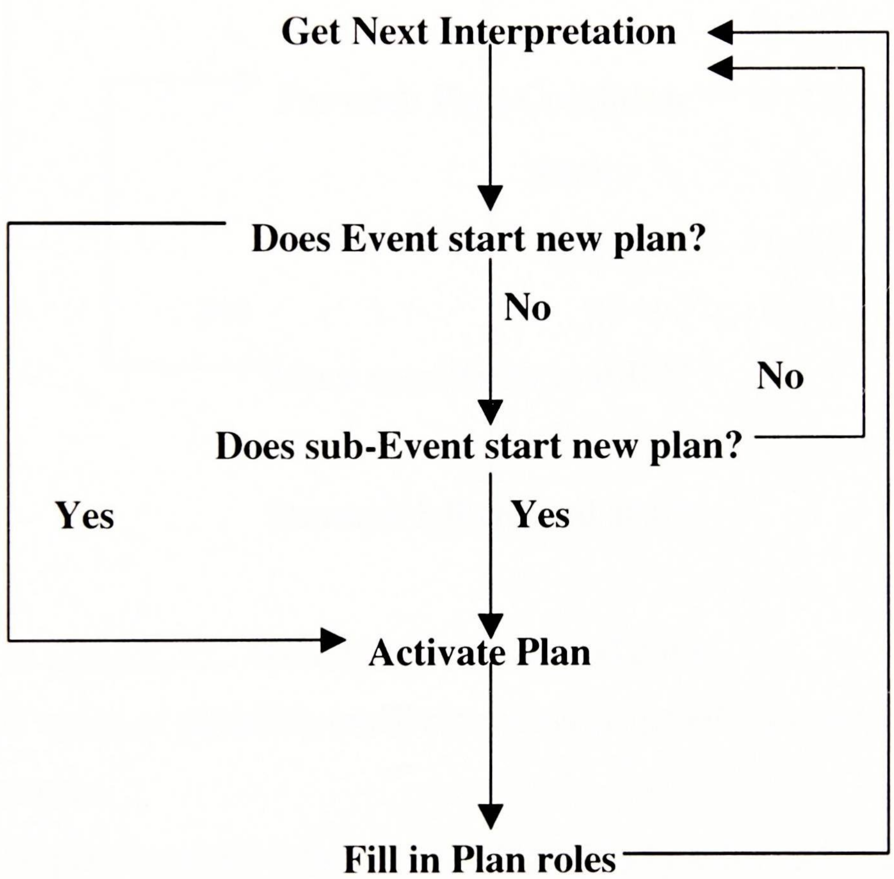
i) The planning procedure is called, and passed each event representation in turn.

ii) For each ambiguous interpretation of the text (represented by an event representation) that exists, each plan is called.

iii) If the action is a legitimate starting point for the plan, the plan is activated, and added to the active plans structure.

iv) If the action is deemed to be related to a sub-event that is likely to initiate a new plan, the process repeats from step ii) using the sub-event.

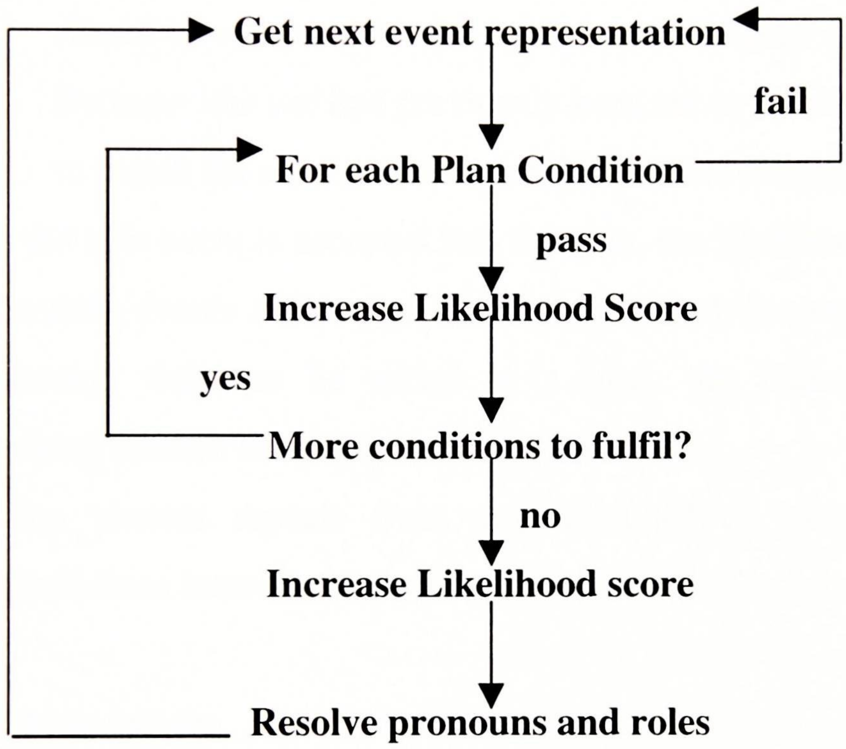
v) Any relationships which exist between actions and actors that are derivable from this initial event are created. For instance, a particular actor may play a role in other events; that actor's role is filled in during this step.



For example, consider the text fragment '*a taxi driver was charged with dangerous driving when a passenger was killed...*'. The action of charging the taxi driver comes at the end of a chain of events describing how the charge came about. The action of a passenger being killed, or the taxi being driven in a dangerous manner may well lie at the start, therefore whenever a *ChargeEvent* is created the model passes the charge itself to the planner. Assuming in the above example that a *Car Accident* plan is created, certain roles will be set within the plan after this initial activation to restrict the events subsequently deemed to be part of it. For instance, the taxi driver must be the driver of the vehicle involved, the taxi driver must be the actor who gets arrested, and so on.

2.2.3 Adding an event to the planner

As well as possibly activating a new series of plans, each event representation is also passed to the planning mechanism to see if any currently active plans appear to provide the required context for the new event. The algorithm is as follows :



i) The series of plausible event structures is passed into each currently active plan.

ii) The plan supplies information which fills in actor and event roles, and disambiguates pronoun references.

iii) The plan checks its own internal set of conditions (Bares *et al*, Vilain, Kautz) against the adjusted event representation. The more conditions an event has to satisfy to be accepted, the higher the likelihood score then associated with the event.

iv) If all the relevant conditions are satisfied, the event is added to the plan structure. A step towards the further disambiguation of the series of possible event representations takes place; an event which is accepted to be part of an active plan is given a large increase to its likelihood score.

v) The plan checks to see whether the new event supplies any information which in some way invalidates the premise of the plan. If this is the case, the plan is marked as invalid, and deleted by the planner.

For example, consider the text fragment "*the vet was jostled as he left the court*". The conjoined sentence is analysed first, and an *Attack Plan* (amongst others) is created. The first sentence is then analysed and the plan supplies the role information that the Object of the *Attack Event* must be the same as the Agent of the *Move Event*. Because '*the vet*' had previously been set as female, the pronoun fails to match the actor, and the plan is therefore terminated.

vi) If the new event is accepted into the plan, the likelihood scores of all previous events in the plan are also increased, because the more information that can be added to a plan, the more likely the underlying context the plan provides is correct.

vi) The process repeats from step ii) until all possible event representations have been interpreted.

2.2.4 Plan failure

If an event occurs which puts a plan at risk of remaining unfulfilled, the model consults the plan to see whether any alternative actions on the part of an actor are to be expected, and whether the successful analysis of any such plan tends to indicate the reinstatement of the original plan, or is in fact a substitute method for achieving the same goal.

i) Events that fail to add to a plan are passed to the failure module.

ii) If the event is deemed to cause the temporary or permanent failure of a plan

i) The plan suggests possible remedial steps. If necessary, the failed plan is removed from active consideration.

ii) Any relevant plans are activated, and analysis continues.

2.3 Scripts

A very large percentage of the causal connections that exist within a text are explainable only in terms of the context within which the actions take place. The social world is not made up of actors constantly indulging in random behaviour. They are often motivated by some larger goal they are acting in pursuit of, or can at least be presumed to be behaving rationally. Skvoretz mentions this in passing, while discussing well-formed narratives; his example being a narrative of someone walking - a left-leg movement cannot follow a left-leg movement. This argument is somewhat nullified, however, if rationality of the actors is presumed.

Both scripts and plans (Allen, Schank, Schank & Reisbeck, Wilensky) (and similar structures such as frames [Lehnert]) are an attempt to deal with this phenomenon and they exhibit similar internal structures, but they behave in different ways, and consequently they will be explained separately. This section explains the theory behind scripts and plans, the practical implementations of them, and their implications for the object-oriented framework of the model.

The descriptions being given of scripts are of their more traditional structure in the realms of text analysis; the section on scripts as they are implemented details the differences between tradition and use, and the reasons for this.

2.3.1 Script structure

Scripts are an attempt to situate an action in terms of the contexts of actions that are going on around it in quite a specific way. They pre-suppose that actors will not perform in totally random ways, and that some form of coherency will characterise a sequence of actions as could be spelled out by a narrative. Scripts organise knowledge to aid the process of understanding. They reduce the space of rational behaviour to those events that are likely to take place given a variety of contextual information which the script defines. Scripts could be said to be providing the true causal information of the two methodologies. They tend to eschew the connectivity associated with the way in which something is done, and dwell on the more philosophical question of why it is done.

When stories are being related they are, on one level, an incomplete account of the actions that actually took place. Narrators leave out information that is inconsequential to the outcome of the discourse, in order to be concise and not appear boring. Human beings apply subconscious scripts to formalise the events that they are explicitly told into a coherent causal chain of actions. So long as the correct script has been activated, the causal links in the stock-situation the script covers should become clear. Actors who previously went unmentioned are assumed to be present because of the experience of the listener in dealing with this domain of story. For a similar reason, the listener is capable of making assumptions about the events that took place in between those that are being explicitly related.

Although structures concerned with providing local causal information (making them functionally similar to scripts) were created, they were dramatically different in structure.

All scripts as implemented have four slots that require filling. Specialised (derived) scripts include other slots; these and other issues will be dealt with in the section on scripts and object orientation, below. The slots common to all scripts are as follows :

KeyEvents and *Events* are lists of events that form the nodes within a script. The difference between them is to do with the idea that the information in a script requires ranking to highlight the most important parts (DeJong). When a script is checked for activation, KeyEvents are checked first, Events afterwards. KeyEvents are often set so that they can only be fired once; if such an event was to fire again, the implication would be that a separate case of the same script was necessary. For instance, a *MeetingScript* restricts the number of meetings it can describe to one; if another meeting is referred to in a text, the assumption made is that it is another independent meeting.

A list of the *People* who have taken part (explicitly or implicitly) in the events describing the script is kept, to protect against fitting too much into a single instantiation.

Finally, a slot representing the *summary of events* within the script is also kept. This is obviously empty to start with, and is only filled at the stage in which the model finds a list of events suited to abstraction. Strictly speaking, no such slot is necessary because given the same information a script will always find the same summary representation. For simplicity's sake it was included.

2.3.2 Scripts' role in abstraction

Scripts also provide a second function in the model, that of attempting to *create abstracted representations* of the events contained within them. Under these circumstances, the script is instantiated simply as a container. Because each script is supposed to be representative of a particular kind of event, the events it contains within it (the *Events* slot is used) are supplied with different summarising information. Therefore, two overlapping sets of events may yield completely different summaries despite their common elements, because of the specialised connection information in each script.

The script itself contains *roles* which represent actors and events that figure prominently. For instance, the *TerrorScript* deals with incidents perceived as terrorist-inspired, and therefore contains a slot called *Outrage* representing the atrocity carried out. Any local summary derived from this script is almost certain to be based around this one incident.

Some justification for this approach to creating event summaries can be found in the works of Propp (see Silverman) and Halliday. Propp's thesis was that there are very few distinct stories; they are a variant on one of a few types. This postulation leads to the structuralist comparison between the appearance of detail and complexity and the simple, repeatedly applied underlying structure of reality. To follow Silverman's explanation, the theory can be illustrated by fairy tales. Actors in stories do not derive any importance for who they are, but for the role they play within the scope of the story itself.

Fairy tales across cultures share themes, which can be broken into their constituent parts, which can be replaced by other actors equally suited to perform their function. Silverman demonstrates this by using the text that "*A dragon kidnaps the King's daughter.*"

Element	Function	Replacement
Dragon	Evil	Witch
King	Ruler	Chief
Daughter	Loved one	Wife
Kidnap	Disappearance	Vanish

By accepting that the role rather than the actor is important, it is possible to preserve the underlying structure of a story while changing the actors and indeed some of the events! By using the tabulated information above, the original fairy tale could be rewritten as "*The witch made the Chief's wife disappear.*"

In order to change the elements in the story while preserving its structure, it is necessary to be able to situate the function of the themes in terms of that story. Consider the model example above, concerning the stabbing of the policeman. In summarising this story, it is necessary to identify the themes contained within it. Five are readily identifiable, the *Criminal* (the man), the *Guardian* (the forces of law and order), the *Victim* (the policeman), the *Crime* (the stabbing), and the *Retribution* (the arrest).

The script fills in the Crime, Criminal, Victim and Arrest slots explicitly. The Guardian theme is somewhat implicit in the story; the statement pertaining to appearing in court presumes a judge is there. Since the script deals with the summary along the lines of "*Retribution* by *Guardian* for *Criminal* committing *Crime* on *Victim*", the story can be reconstructed to have a different meaning with the same underlying structure. Dealing with only the themes that are explicitly stated above, the roles central to the meaning of the story can be replaced thus :

Element	Function	Replacement
the man	Criminal	the bandit
stabbing	Crime	cattle rustling
policeman	Victim	townspeople
arrest	Retribution	run out of town

Given these replacement roles, the underlying story structure could equally, if somewhat fancifully, be representing :

"*The lynch mob ran the bandit out of town because the bandit had rustled their cattle*".

Given a robust script-firing mechanism, the events that formed this story in the first place are equally suited to the same *Arrest* script. The importance of roles over actors is amply demonstrated. Therefore, the crucial role in creating summaries is to identify the roles active within subsections of the story, a job for which script-like mechanisms are perfectly suited.

Interestingly, Heise raises this point as a drawback to Abell's theory - the underlying structure of the story is abstract to the point of containing no semantic information at all, giving the analyst *carte blanche* to rewrite the story how they like. Scripts could potentially have this criticism applied to them as well, but as explained above, they appear to provide the perfect vehicle for creating summaries of events.

3. The example text

What follows is the output of the computer model with all the debug flags turned on. It is interspersed with commentary explaining what the output means, and also diagrams and text explaining how the model uses the structures described above to structure the text in a form suitable for the results procedures.

Now starting analysis of clauses connected with : found in

The code outputs the main verb in the clause that it is presently analysing. In this case, the model knows of only one case in which the verb form "*found in*" is used. Until such a time as more cases are added, an event representing 'the discovery of an object in place x' will be created when it is found.

Now finishing analysis of clauses connected with : found in

Because the analysis of clauses can recurse and therefore become complicated, the model displays the verb form associated with the clause it is examining at such a time as the analysis terminates.

Now starting analysis of clauses connected with : sleeping rough

Noun phrase...2

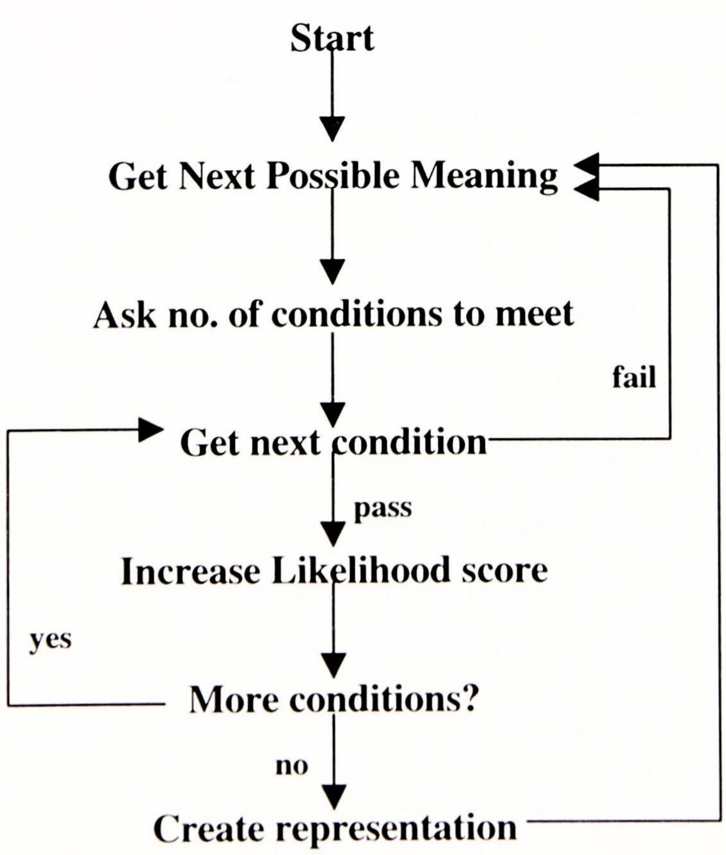
an outdoor adventure

Is a new dummy event required?

(Y/N) ?n

The parsing algorithm constructs a simple noun phrase from the phrase "*an outdoor adventure*". However, no record of an event structure representing this phrase has been found, so the user is prompted as to whether they wish to be taken through the steps used to create a new event type for it, assign roles to the resultant event structure, and so on.

The parser next attempts to construct some meaning for the entire clause, in this case "*he had been sleeping rough with friends as an outdoor adventure.*" The basic procedure for this is :



The run-time output is as follows :

Beginning semantic decoding...

1 context conditions to fulfil to create event.

The first piece of information displayed is the number of conditions that the program is aware of that must be fulfilled in order for the event representation under consideration to be permitted.

Context condition is :

ignored and default meaning assumed...

In this case, no 'real' condition is present. As the database of event 'meanings' is built up, conditions may be added, to differentiate between these distinct meanings. This default condition simply says, "Here is the default meaning I believe to be correct".

Result is passed. Associated weight is : 1

The code next informs the user that the result of analysis of this condition has been successful. The Associated Weight is an attempt to measure the differential power of the condition. As clauses are analysed, the possible events they represent are decided between on the basis of a series of suggestive factors (see page 59 for the complete discussion of this). Clearly, a default 'catch-all' condition does not differentiate much, so little discriminatory power is associated with the event on the basis of this condition.

All conditions fulfilled (0 inconclusively). Creating possible event.

Event representation created is :

slept rough

The user is told that all the conditions associated with a potential meaning have been fulfilled. The user is also told how many of these were inconclusive; that is, how many didn't fail, but insufficient evidence was available to say that they had passed. The event representation created is then displayed.

i) Every possible alternative based on the source text is created. For instance, "*lost*" currently causes the instantiation of two events; if the Object role is filled by a part of the body (as in '*the man lost an arm*'), a *Health* event may be applicable and is therefore created (following the notion of *surface semantics* [Cullingford & Pazzani] or *syntagmatic restrictions* [Quillian]), or a *Loss-of-possession ('the man lost the dog')*, event may be required instead (following, for lack of further information, the notion of meaning dependent on syntax). A possible event representation that fails any initial condition is immediately removed from consideration; those that pass are collected in a set and passed to the remainder of the planning process.

ii) The potential new event structures are called upon to analyse themselves and the clauses that brought about their creation as far as they can.

The result of this self-analysis is displayed to the user :

slept rough in the friends

Incrementing likelihood score by : 1

The likelihood of this particular event representation being the correct one is increased, to reflect the fact that the generic 'sleep rough' event object has managed to use some of the data in the clause. If it had used none, the possibility exists that this failure was caused by the unsuitability of the created event representation. The more information used at this stage, the more the likelihood score is increased.

iii) The remaining alternatives are passed consecutively through the currently active plans in the model to attempt to disambiguate the desired meaning of the sentence (as in Cullingford & Pazzani's concept of *contextual knowledge*). Following the plan activation criteria explained above, new plans are also activated by the latest information. Starting a plan or being added to a plan (and therefore improving confidence in that plan as a reasonable explanation of events) conditions will add more to the representation's likelihood score.

The debug appears as follows :

Incrementing plan activator event.

slept rough in the friends

Incrementing likelihood score by : 10

A plan (as will be seen, the Accidental Death plan) is activated by the potential event representation under analysis. The likelihood score is incremented by a score of 10, to reflect the fact that the event representation appears to have some potential planning significance.

Activating accidental death plan...

Plan number : 0

The new plan is then initialised using information contained within the event representation that caused it to be activated. Plans contain information pertaining to how actors operate within them, which is helpful in providing some context for an event. For instance, "*The waiter asked the man what he wanted for dinner*" is likely to part of some kind of restaurant-based plan. "*The man asked the waiter what he wanted for dinner*" is quite likely *not* to be, despite the bald fact that essentially the same event is being described.

i) Attempts are made at resolving pronoun references by searching role information contained within each plan. If suitable candidates can be found given the state of the plan's activation, the relevant substitutions are made, and the event is added to the plan's explanation of the unfolding action. Other inferences regarding roles within the new piece of information are also performed at this time.

Now seeking to match pronoun he

to : the boy

Does actor match pronoun ?

(Y/N) ? y

In this case, the plan attempts to resolve the roles the actor represented in the clause by '*he*' will play. This necessitates resolution of the pronoun itself. The first suitable candidate found by the parser is the actor known only as '*the boy*'; the user confirms that this is correct.

ii) If pronouns remain unresolved at the termination of a particular plan's analysis, the analysis continues when the next active plan is called until candidates for all such pronouns are found or all active plans are exhausted.

iii) If no active plan satisfies the requirements of the input text, then the remaining references are resolved by asking the user which actor fits the pronoun. The model maintains a *history list* of actors (Seely Brown & Burton). The model moves backwards through this list until a possible candidate meets those criteria that do exist, and prompts the user to confirm or refute the choice. This new information is added to the event representation, which is then re-submitted to the planning mechanism to start a new wave of planning analysis.

iv) A tally is kept, marking the number of hits made by each possible event representation; the higher the tally, the more likely the current interpretation is of being considered the correct one.

v) Each event representation is passed to the planning mechanism again in an attempt to instantiate any new plans that may apply. A successful instantiation is again recorded in terms of the event's tally, but given less weight than hits in step iv), on the grounds that this step is more generalised.

vi) If more than one alternative still exists, further more general checks are made, to see whether a *domain* specific interpretation exists that can resolve the ambiguity. Domains are the general area a story falls into; for instance, crime, in this case accidental death, and so on. Although very much an inexact science, as actors and events are created, weights are added to various domains in the model, indicating what style of story the model believes the text is. So, for instance, the presence of a police officer may indicate a crime story. In this case, the event of the boy sleeping rough adds weight to the accidental death story.

vii) Event representations are disambiguated by the addition of future events to the planning process - the more one particular plan is fired, providing the context of the narrative, the more confidence is placed in the event representations contained within it.

viii) Once the entire text is read, ambiguous event structures are resolved finally:

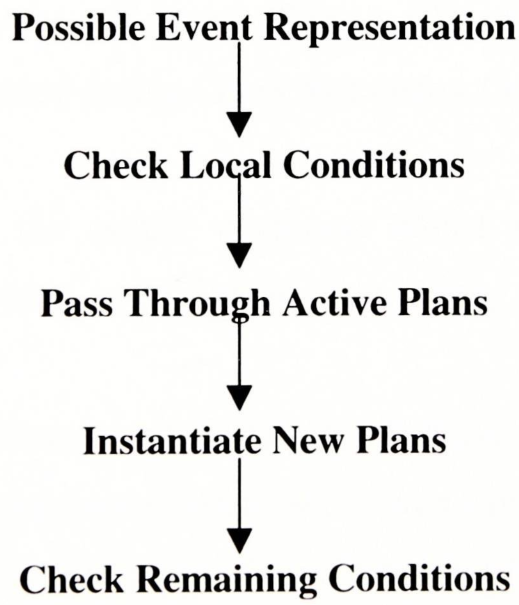
i) For each series of ambiguous event representations, the maximum scoring event is selected.

ii) Planning information involving the redundant event representations is removed.

iii) The redundant event representations are deleted.

3.1 Choosing amongst Event Representations

The likelihood that a particular event representation is in fact the intended one is done by combining the results of a series of heuristics, variously weighted. Graphically, this is represented as :



The four metrics used are :

i) A measure is taken of the degree to which the event representation makes use of the source text by counting the number of fields set within the event by its self-analysis step. This step is subsumed within the *Check Local Conditions* requirement of event representation creation.

ii) Counting the number of times that the event representation causes the planning mechanism to fire its currently active plans. The selection procedure is weighted such that event representations are heavily favoured if this heuristic is fired, to reflect the importance that context plays in resolving textual ambiguity.

iii) As plans unfold to explain more and more event representations, the weights of the events within them are increased, to reflect this increased confidence in their being the desired meaning of the narrator. Both this step and the last are performed during *Pass through Active Plans*.

iv) Event representations that cause the activation of a plan have their weights increased, during the *Instantiate New Plans* step.

v) Counting the number of conditions that are fulfilled by an event representation in order to be accepted by a plan. The conditions themselves have varying weights attached to them. For instance, an event which suggests the text lies in a domain thought to be unlikely by the model to be correct adds relatively little to an event's likelihood score. A specific actor filling a role adds relatively more. This step is performed during *Check Remaining Conditions*.

The analysis of the output displayed whilst the model is being run demonstrates this behaviour andis shown from here, dealing with the second sentence in the text.

Now starting analysis of clauses connected with : believed

Now starting analysis of clauses connected with : died from

The parser moves onto the next clause in the text (in fact, an entire sentence, "*the boy is believed to have died from smoke inhalation after a fire started in the garage where the boys were using candles*") , and begins to work down through its subordinate clauses.

Event representation created is :

the choking

In this case, the parser decides that the phrase "*smoke inhalation*" is representative of a 'choking' event. The planning algorithm is passed the 'choking' event to see if it fits anywhere. It fits in the Accidental Death plan, as the cause of the accidental death itself. Thus its likelihood score is increased.

Now starting analysis of clauses connected with : started

Beginning semantic decoding...

The first clause associated with *started* to be examined is the noun phrase "*a fire*". There are a variety of meanings for this, and three events are created as possibilities:

Beginning semantic decoding...

1 context conditions to fulfil to create event.

Context condition is :

ignored and default meaning assumed...

Result is : passed. Associated weight is : 1

All conditions fulfilled (0 inconclusively). Creating possible event.

Event representation created is :

the controlled burning

In exactly the same way, two other events are also created (the output is edited to save space) :

Event representation created is :

the uncontrolled burning

and

Event representation created is :

bombing

As before, the plan module is called, to see how any or all of these event representations fit in with what has gone before.

Incrementing likelihood in active plans...

the boy had slept rough in the friends

Incrementing likelihood score by : 20

If a slot is found in an already active plan, then the extra credibility lent to the plan by the addition of further information is reflected by increasing the scores of those events already in the plan. In this case, the likelihood of the 'sleep rough' event is increased by a score of 20. The algorithm then moves to the event newly added to the plan.

the uncontrolled burning

Incrementing likelihood score by : 200

Finished incrementing likelihood in active plans...

The newly added event is given a big increase in its likelihood score, because not only does it help explain the previous events in the plan (by providing more contextual evidence that they were correctly deduced), but the new event comes after the others, meaning the range of events that could be added after them is that much smaller.

The parser then moves to the verb itself, "*started*".

Beginning semantic decoding...

1 context conditions to fulfil to create event.

Context condition is :

that Syntactic subject has an associated action.

Result is : passed. Associated weight is : 1

This meaning of "*started*" is that of some other event starting; the condition required for fulfilment of this meaning is that the subject of the verb has an event associated with it. In this case the subject is "*the fire*", and it has three potential events associated with it, so the condition is successfully passed.

All conditions fulfilled (0 inconclusively). Creating possible event.

Event representation created is :

Abstract Begin Event.

the controlled burning

Now finished analysis of clauses connected with : started

The event representation is that of some event starting, but as yet, this has not been decided. The 'work-in-progress' form of the event has three possible meanings, as constructed above (controlled and uncontrolled fires and a bombing event). The first ("*the controlled burning*") is shown until such a time as one of the three is selected as most likely.

The recursion of the clause structure starts to unwind, and the parser moves back to deal with "*died from*" again.

Beginning semantic decoding...

1 context conditions to fulfil to create event.

Context condition is :

domain 6 is inferred as likely.

The only real ambiguity with the verb form in question is whether the death was in some way violent. In order to judge this, the parser consults the domain module to see whether the only violent death domain it is currently aware of is to any extent active.

Result is : passed. Associated weight is : 13

The evidence of the text so far suggests that the text could fall in to the (admittedly very broad) category of Accidental Death. The weight currently associated with the domain is added to the score associated with the new event.

Event representation created is :

Kill Event

The form of the clause "*...died from smoke inhalation*" leads the parser to believe that the event represented by "*smoke inhalation*" is the cause of the event represented by "*died*". The *Reason* slot of the 'kill' event is therefore filled by the 'choking' event. This causes an increase in the likelihood score of the 'kill' event (because of the use of clausal information to confirm that the meaning deduced is correct).

Once again, the plan module is called and the new kill event is added to one of the plans currently active, the Accidental Death plan.

Plan has potentially terminated!

Incrementing likelihood scores in active plans...

With the addition of the new event, the plan module deduces that enough elements are explicitly present in the text to say that the accidental death plan is sufficiently well-formed to have potentially finished - the death itself, the cause of death, and an explanation of how the cause itself arose. This bumps up the likelihood score of the events within the plan once again.

Once the entire text has been dealt with in this way, and all possible information about the events has been filled in from the plans and other contextual information, then the ambiguous meanings are excised, by choosing the highest scoring event representations (as defined by their likelihood scores). Most are fairly unambiguous, because of the explicit nature of the text, but the following is selected.

Searching for highest scoring event...

the uncontrolled burning

Likelihood score for the uncontrolled burning

is : 801

Maximum score so far.

bombing

Likelihood score for bombing is : 21

the controlled burning

is : 2

The maximum scoring event is therefore that representing *the uncontrolled burning* and it is chosen. Plans involving either of the rejected events are removed, as is the information they contain.

CHAPTER FOUR :
COMPARISON METRICS

Comparison Metrics

1. Introduction

This section gives details about how two aspects of Abell's theory (comparing the texts and finding causal links within them) were performed.

2. Results Procedures

It has been noted (Abbott, Michaelson-Kanfer) that Abell's theory does not include a detailed comparison procedure beyond the conditions for identity structures mentioned before. Because it is certain that each of two accounts will include information and opinions unique to them, they will almost certainly present different underlying structures. A method is required which takes Abell's theory for creating structures and gives some measure of the degree of similarity between them, other than a simple test for identity.

At least two methods of extending Abell's theory to include a robust results step have been suggested, and these were implemented in the computer model to one extent or another. Although they did not play a significant part in the final results method implemented, the code for them still exists and could be reintroduced. This section briefly outlines the two alternate results methodologies implemented prior to the final system chosen. They fall into two categories, *sequence comparison* (Abbot, Kruskal *et al*) and *pattern matching* (Kosaka).

Sequence comparison is perhaps most well-known for its use in attempts to unravel the structure of DNA strings and its work in the field of speech processing. Pattern matching in the form is takes here is a method of reconfiguring a problem and applying statistical tests to derive conclusions.

The methods themselves and their relative advantages and disadvantages are briefly discussed, and the reasons these methods were not in the end exploited fully are outlined. The final analysis method and its suitability to the theory of Comparative Narratives is explained.

2.1 Sequence comparison

Sequence comparison can be considered an attempt to answer two questions:

1) Are there common patterns amongst a set of sequences?

2) If such patterns exist, how are they produced?

Question 1) reflects interest in the individual units that make up the sequences in question, and question 2) pertains to issues dealing with the generation of such sequences; that is, what underlying rules exist for the creation of these sequences? This question is very similar to that of Abell, which asks "How characteristic is an event for a generation of similar events?"

The next section details a typical sequence comparison algorithm, drawn from Kruskal and Sankoff:

2.1.1 A sequence comparison algorithm

Sequence comparison is a methodology by which a measure of the distance between two directed networks is measured. A directed network is defined by :

1) A *directed graph*, with no directed cycles.

2) A *successor* to a node is a node that can be reached in one step along an arrow.

3) A *predecessor* is the reverse of the above.

4) A *source* has no predecessor.

5) A *sink* has no successor.

6) A *full sequence* is a path from source to sink.

7) An *initial sequence* means paths starting at a source.

Then a *supersource* is connected to each source in each network, meaning that each node in a network has at least one predecessor (exactly one in the case of initial and full sequences), and each full and initial sequence starts with a supersource.

Assume that two networks exists, a and b. Every alignment between them $a_0 \rightarrow b_0$ has an associated weight Ø.

d_{ij} represents the smallest distance between an initial sequence in one network terminating at a_i, and an initial sequence in b, terminating at b_j.

$d_{ØØ}$ = Ø.

a_{i*} = all predecessors of a_i.

b_{j^*} = all predecessors of b_j.

$$d_{ij} = \min \begin{cases} w(a_i, \varnothing) + \min_{i^*} d_{i^*j} \\ w(a_i, b_j) + \min_{i^*} \min_{j^*} d_{i^*j^*} \\ w(\varnothing, b_j) + \min_{j^*} d_{ij^*} \end{cases}$$

Let a_i and b_j run through all sinks in the networks a and b.

$d(a, b) = \min_i \min_j d_{ij}$.

Sequence comparison has been suggested as a suitable method for undertaking the comparison of the underlying text structures produced by Comparative Narratives, specifically in the form of an extension to sequence comparison theory called *Optimal Matching*.

2.1.2 Optimal matching

Optimal matching provides a measure of the similarity between sequences that contain chains of elements taken from a small population of elements. The process is widely known in the natural sciences. Optimal matching produces measures of likeness between intervals of elements within sequences, rather than more specifically about sequence patterns.

Optimal matching applies a series of weighted operations to convert one sequence into the other, the resultant distance being known as *Levenshtein distance*. Bearing in mind that each operation type has its own individual weight formula, it is approximately true to say that the "closer" two sequences are to each other the fewer operations will be required to finish the conversion.

Depending on the operations implemented in the model, there may be a number of ways of converting one sequence into each other. The weights associated with each operation type then come into force, and the transformation process that incurs the lowest operational expenditure is taken to be the minimum distance between the sequences.

2.1.3 Points to note

The relative lengths of the sequences are important when the number of operations incurred is considered. For instance, comparatively many substitution operations in the conversion of a short sequence is more likely to merit a rethink at the comparison operations implemented than an equal number of substitutions in a sequence ten times as long. In order to normalise these ratios, the distance (number of operations) between the sequences is divided by the length of the longer sequence.

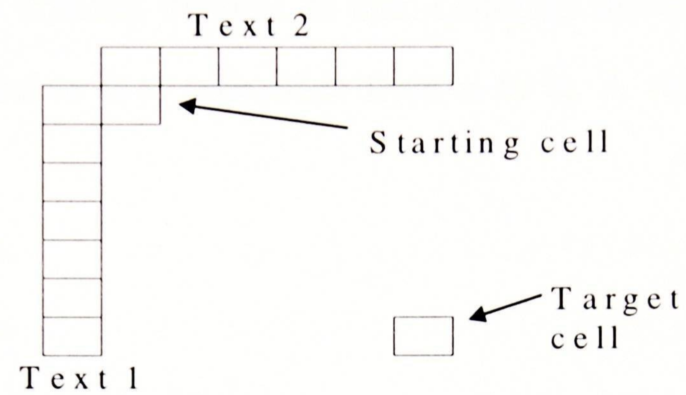
The weights attached to the operations are some function of the data and context from which the data came. This formulation is true for all the basic operations generally associated with optimal matching. It is obviously true for substitutions - the greater the difference between the elements, the greater the cost involved in changing between them must be. It is also true for insertion and deletion operations although this is harder to see in non-contextual situations.

Consider Abbott and Hrycak's example of a sequence of numbers reflecting the movement of actors up a hierarchical ladder of ten career steps over time - the higher the number, the further up the organisational structure the actor has climbed. Then, assume one of the sequences contains a ten, reflecting an actor's achievement in reaching the summit of the organisation. Because of the relative rarity of this post (as compared to many actors at level 1 or 2), the cost of inserting or deleting such an element could be argued to be higher.

Setting the weights for each operation is something of a lottery, and therefore adjustments are made once the algorithms are being run.

2.1.4 Optimal matching in practice

Optimal matching introduces a series of operations by which one sequence is turned into another, the basic set being *insertion*, *deletion* and *substitution*. Each of these operations has an associated *weight* or cost, either a fixed value or a function of the two elements being processed at any time. Computationally, a matrix is formed from the two vectors, and a path "through" this matrix, from the origin to the opposite diagonal corner, is measured.



The vectors marked texts 1 and 2 are the event representations from each of the texts.

Each matrix cell is indexed to show the costs associated with moving through it via each of the associate operations. Thus, continuing Abbott's example, each cell could look like:

Substitution Cost	**Deletion Cost**
Insertion Cost	**Minimum Cost to Cell**

Once all of these values are calculated, a simple recursive search through the matrix will find the least costly path through the matrix.

2.1.5 Advantage

Because sequence comparisons are based around vectors of numbers, constructing a consistent form of measuring the distance between an element in each network is simple. Defining operations to minimise the distance between the networks would not be difficult because of this. Consider the basic insertion («), deletion (») and substitution (↔) operations from a_i to b_j. A valid scheme could consist quite simply of;

Cost (« (a_i)) = a_i,

Cost (» (a_i)) = a_i,

Cost (↔ (a_i, b_j)) = $| (a_i - b_j) |$

depending obviously on the context of the information.

2.1.6 Disadvantages

Some form of sequence comparison appears to be highly suited to producing a measure of the distance between two networks representing the underlying structures of the input texts. However, such an algorithm would only work on the shape of the structures themselves, and would necessarily have to discard the semantic information each node represents unless a scheme can be assembled that reliably produces a gauge of the distance between two qualitative objects, those being the events contained within the elements of the graphs.

No attempt was made to create such a substitution scheme, for two reasons. First, the meaninglessness of any number purporting to be the distance between, for instance, a speech act and a walk act. Second, assuming some notional meaning could be attached to the numerical distance between any two event types, the job of maintaining the scheme's consistency would be herculean, if not entirely impossible.

If it had proved possible to create some kind of hierarchical event structure relating actions to those semantically similar, such a measure of closeness or distance may well have been possible, for instance by counting the number of steps required to get from one to the other. As the events are structured in the model, the only axiom that can safely be stated is that they are all instances of an abstract event, therefore a sequence comparison will fail to distinguish the degree of difference between virtually all pairs of events; that is, all the scheme will reveal is that two events are the same or different.

If sequence comparison was applied to the job of converting one series into another by assuming that no two event representations were any closer or further away to each other than any other event representation, an optimal matching model could be (and was) applied. The resultant measure of closeness or distance would still mean very little in the context of comparing the texts' underlying structures, rendering the algorithm rather pointless for analytic purposes.

Sequence independence is required, which in the context of partial views from inter-related actors seems potentially unlikely. Abbott and Hrycak contend that "optimal matching is not a substitute for, but a complement to, stochastic analysis". Comparative Narratives attempts to assert some order on a sequence of actions by linking them causally. While the model does allow for unintended outcomes of intentional actions, the actors are also presumed to have a degree of rationality, meaning that causal events are not wholly independent; an outcome event is being caused by its precursors, and therefore is not independent of them.

2.1.7 Optimal Matching implemented

The matrix used to transform one vector into the other is created dynamically whenever the Optimal Matching code is called. The following step is to fill in the operation costs in each cell. Insertion and Deletion costs were fixed at 10 and 1, respectively, the logic behind these values being that it is better to remove information which is explicitly in one narrative than to insert extra information which may or may not legitimately belong there into the other.

Substitution costs were rather more problematic. A scheme was worked out to measure the differences between event representations (so that a substitution between two identical events scores 0), but it was based around the (later abandoned) attempt at creating a hierarchical semantic structure of events mentioned before, and effectively did little more than distinguish between events on the same branch of the tree and those on other branches, and then between different leaves on the tree, and so on. Substitution costs were deliberately made high (in the regions of 100s, and over 1000 if two events were deemed to be opposites - agreement and disagreement, for instance) for similar reasons that insertion operations were discouraged. The crux of Abell's theory rests on the assertion that local detail is erased to compare narratives - in this light, adding information (which both insertion and substitution operations represent) seems fairly ludicrous! Their presence remains necessary however; an OM scheme consisting only of Deletion operations will simply erase the vectors to nothing unless they are exactly the same to begin with.

These flaws were felt to be tied fundamentally enough to the basic doctrine of sequence comparison that any scheme along these lines was deemed unworkable in any meaningful sense.

2.2 Bales's categories

Another method of comparing structures produced by Abell's theory that has been suggested is some method of pattern matching (Kosaka). Actions are recategorised into more abstract classes, representing general sentiments as opposed to specific actions. Bales' categories are an attempt to do this, re-aligning events into one element of six pairs of categories, those categories being;

1) *Show solidarity / antagonism*

2) *Shows tension release / tension*

3) *Agrees / disagrees*

4) *Gives suggestion / asks for suggestion*

5) *Gives opinion / asks for opinion*

6) *Gives orientation / asks for orientation.*

The Bales system views "each act as a response to, the last act of the last other, or as anticipation of the next act of the next other".

Various differences exist between Abell's and Bales' view of the event. While Abell's actions are explicit, the Bales categories are far more general. Abell permits groups of actors to perform an action, while Bales does not. Abell's method has a degree of formality about it, whereas Bales' method is experimental. Coupling the two schemes is not a trivial task, because of the amount of contextual information that may be required to situate an action into one of Bales' categories. Even with this information, it may be difficult to decide whether an action is *giving opinion* or *giving orientation.* Such a distinction could be made in terms of Abell's influence relations, implying that a high degree of expertise with the domain being studied would be needed to recognise its subtleties.

2.2.1 Using the categories

Once the underlying structure of a text has been revealed, the Bales' categories are applied, producing a number of new graphs representing such relations as actor inter-relations and event-type "followed by" relations. A number of statistical tests are also possible, seeking correlations amongst the relative frequencies of occurrences of the various categories.

2.2.2 Results and conclusions

The question to ask at this point is "What new information do these graphs tell us?" Frequently this is not clear, and the graphs are therefore a form of displaying the information for analysis in some other way, rather than being analytic themselves. Kosaka implements certain statistical tests on the frequency with which certain categories crop up without drawing out any particularly meaningful results from them. No firm deductions are made from the graphs either.

Because of the statistical basis for some of these tests, it is possible that unless a case study is huge, the sample size of interactions will be too small to draw out any worthwhile conclusions. If this is in fact true, the whole method of reclassifying actions into smaller groups (like Bales' categories) and checking the resultant patterns may well be invalidated in terms of Abell's theory. Since the theory is formalised, but lacks a suitably specific results stage, some external results step is required that will take as its starting point the structures produced by Abell's theory. Any method that cannot fulfil this assumption is not suitable for use in extending Abell's theory.

Categorising schemes were eventually rejected because one of the basic tenets of the computer model is the attempt to avoid statistical procedures as far as possible, and remain on a qualitative footing. Any potentially useful information from such a scheme as that outlined above would be derived from these statistical tests, thereby introducing a degree of incompatibility with the philosophy behind the model, and more prosaically, the remainder of the results process.

Such methods would be suitable for some form of back-propagating neural network, trained to find the characteristic features of the text's reconfigured underlying structure. However, it is debatable exactly what the neural network would be expected to produce. Some theory about what makes a number of texts different would be required, which would be a far from trivial job. Such a problem appears to be symptomatic of these methods in many ways, because the results will either confirm what is already known, or the texts will need to reach some form of "conclusion", enabling a pattern-matching type system to look at generative issues concerning the origins of that predetermined conclusion.

An optimal matching scheme was implemented in the model, and the structures associated with Kosaka's pattern-matching were also produced; because of the drawbacks

outlined above, neither scheme played a further significant role in any other stage of analysis. Kosaka's structures were later removed from the model, but the optimal matching algorithm remains implemented. The results produced by the optimal matching have not since been exploited.

Having discussed the options considered that were ultimately deemed unsuitable, it is necessary now to outline the actual structural analysis steps that were used.

3. The comparison procedures implemented

The methodology used was chosen because it bears in mind the use to which the theory will be put. The goal behind comparing a number of texts is to deduce the elements they hold in common; the comparisons above are, in a sense, taking a step away to look at more general patterns.

When dealing with qualitative data, it was thought to be a good idea to avoid quantification as much as possible. This seems to be especially true when dealing with textual accounts; it would be extremely foolhardy to attempt to produce some measure of distance between qualitative reports of an event. This would involve attempting to measure the difference between two separate actions say a speech event and a movement event. The most that can be said without controversy is that they are the same or different, therefore applying Abell's comparison method on a micro rather than a macro level; within the text, rather than between texts.

The central tenet of the comparison method used is to identify the elements which occurred in all texts. Clearly there are other criteria that could be used, and the circumstances under which they would be used are to do with the content and purpose of the text itself. For instance, newspaper stories are a collection of facts describing an event. The comparison step can simply take the form of deciding which events have occurred in some or all of the texts. The mission statements of a group of co-operatives will outline their goals, perhaps with a measure of how well the goals are being fulfilled. Comparison here is more complex, because the "purpose" of each text is to measure the fulfilment of these goals. Inter-text comparison could take the form either of comparing the extent of this goal satisfaction, or of identifying the goals the co-operatives share. In this way, the analyst is once again almost becoming just another actor in the process of applying the theory, his or her action being the creation of the data itself.

It is important to remember that Abell distinguishes between *abstraction* and *generalisation* of narratives. The difference between them manifests itself in terms of the dimension the data is "smeared" across. Abstraction is the process of summarising pieces of text internally, whereas generalisation looks beyond the boundaries of individual texts to search for structures occurring in more than one text. Both methods are looking for similarities, but with different targets.

In order to replicate this two-pronged approach to comparison in the computer model, two different methods of comparing structures produced by the model have been implemented, and each is explained below.

3.1 Direct comparison

It is important not to overlook the obvious when considering how the structures produced by the theory are to be contrasted; the very name Comparative Narratives implies some form of direct comparison, and this is indeed undergone. Because the computer model produces a sequence of discrete event representations from each input text, rather than a seamless flow of action, it is an uncomplicated matter to search one list to see how many of another list match with it. This comparison process is not quite as simple as it may at first appear to be, because there are two levels at which comparisons can usefully be made :

i) two entire complex event structures can be compared to see if they match in every respect. Performance of this test is simplicity itself; and slight deviation and the events are immediately declared different. For example the following two sentences yield an exact match in their resultant event structures.

"*tehran responded by claiming that britain had planted a bug in its embassy*".

"*iran claimed britain had planted listening devices in iran's embassy*".

ii) The second level at which comparisons can be made is considerably more subtle. Suppose the second sentence above had been "*the dispute began when britain accused iran of links with the ira*".

On the face of it, the events represented by these sentences are now different. Accusations of spying and of links with terrorist organisations cannot be said to represent the same thing on a micro level. But by what may be considered an abstraction of the comparison procedure, it is possible to say that an accusation of *something* has taken place in each case. The abstraction procedure built into the model has the potential of summarising this event into its more general case, but this is by no means a certainty. It is therefore important to search for comparisons on this more general level.

The procedure for the comparison of event representations is :

i) Find a matching pair of events (one from each text) under the strict criteria.

ii) Search along the causal link matrices to find a subsequent pair of strictly matching events.

iii) If a match is made, move to the end of the relevant causal links, and repeat from step ii).

iv) If no subsequent match can be made and no 'holes' are permitted, repeat the procedure from step i) with a new pair of matching events.

v) If no match is made but 'holes' are permitted, insert a 'hole' marker in one list, move to the end of the causal link in the other matrix, and repeat the procedure from step ii).

This idea is enlarged upon by supplying the model with information concerning more general event patterns (somewhat in the style of Bales' categories above, although more specific) via which two ostensibly different events can be matched. Thus a *KillEvent* and a *BombEvent* are deemed to both be a *Violent Act.* In this case the algorithm is as before except

i) Find a matching pair of events (one from each text) under the relaxed criteria.

ii) Search along the causal link matrices to find a subsequent pair of matching events.

3.2 Filtering the Sequential Result stage

A problem that can tend to occur, particularly where holes are permitted in the sequential comparison, is one of an explosion in the number of sequences that match. While they are all correct in that they highlight correlation between the texts, any 'real' information that exists may be lost in a welter of noisy data. Once the sequence comparison has concluded, the model activates a series of heuristics designed to remove information which is either trivial or displaced by another piece of data. The model measures the relative size of a text by the number of distinct events that are constructed during the analysis of a text.

The model achieves this filtration by manipulating a series of variables which affect the run-time status of the model. These variables are open to manipulation by the user before any run of the code; they are manipulated under the model's control only when analysing the results produced from a run.

The procedure for filtering the results is as follows :

i) Execute the sequential comparison step.

ii) Apply heuristic to determine whether a superfluous number of sequences have been found.

iii) If yes :

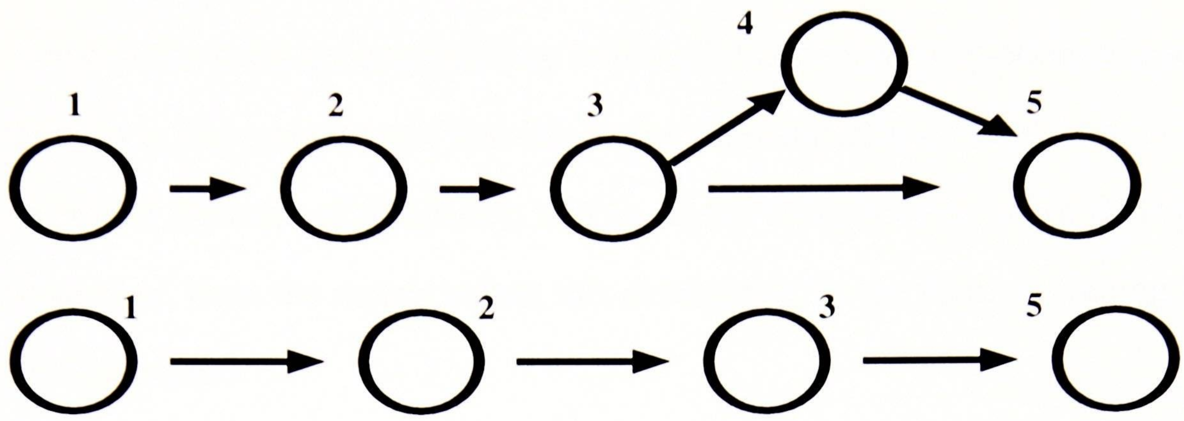
    i) Assist user in adjusting minimum sequence length permitted.

    ii) Assist user in adjusting the number of 'holes' permitted in a sequence.

iv) Resubmit texts to sequential comparison with adjusted model parameters.

3.3 Example

Consider the following partial networks, assuming that the nodes with identical labels are the same;



If sequences of three nodes and longer are to be considered, only three exist - $1 \to 2 \to 3$, $2 \to 3 \to 5$, $1 \to 2 \to 3 \to 5$.

The sequential comparison process can be configured to look for sequences as described above, but with a user definable number of "holes" (non-matching elements) in either of the sequences. For instance the sequences described above will match in a four-element sequence if a single hole is permitted. The lower sequence $1 \to 2 \to 3 \to 5$ will match into the upper sequence with the obvious direct match $1 \to 2 \to 3 \to 5$ and also via the "hole" allowance, $1 \to 2 \to 3 \to$ *<HOLE>* $\to 5$. All shorter subsequences will also be found, unless disabled via the minimum sequence length mechanism. The degree to which these holes alter the semantic meaning of the event sequences they occur in is left to the user to decide.

The information gleaned from this could technically be used in the standard story structures as described in the following section; the program as it runs could be used to build a database of likely causal information that could be used to make the standard story structures evolve. This mechanism has not as yet been built into the model.

Referring once again to Abell's Abstraction / Generalisation distinction, direct comparison of narrative structures as performed in the procedures described above is the process of searching along the inter-text dimension, and is, *ergo*, comparison by generalisation.

4 Testing

Because the texts are compared by looking for congruent sequences of events within them, it is important to test that this sequence comparison metric does in fact provide a robust measure of similarity; that is, as the texts become more divergent from one another, does the metric reflect this divergence, in the form of finding less congruent sequences?

4.1 The definition of the metric

Mathematically, the metric can be defined as follows :

for all texts x, y, there exists $n(x, y)$, which is the number of matching sequences between x and y. The distance d between x and y can then be represented as :

$d(x, y) = 1 - 2n(x, y) / [n(x, x) + n(y, y)]$, in the range 0-1. (I)

Then if $x = y$,

$d(x, x) = 1 - 2n(x, x) / 2n(x, x) = 0.$

$d(x, y) = d(y, x).$

$d(x, y) >= 0.$

<u>Proof</u>

Let $A = n(x, y)$, $B = n(x, x)$, $C = n(y, y)$.

Let $A > B$ and $A > C$; i.e. $B = A - \alpha$, $C = A - \beta$.

Then I above becomes :

$d(x, y) = 1 - (2A / (A - \alpha) + (A - \beta))$

$d(x, y) = 1 - (2A / (2A - \gamma))$, where $\gamma = \alpha + \beta$.

Therefore, $d(x, y) < 0$, which is not permissible.

In order to test the behaviour of the model, it is important to determine what the qualitative properties of a metric are, and how they may best be examined. It is felt desirable that the performance of an aspect of a system is graded in some way. Somerville defines a predictor metric as "measurements of a product attribute which can be used to predict an assorted product quality."

According to Kitchenham, there are three assumptions that such predictor metrics are based on :

"i) We can accurately measure some property...

ii) A relationship exists between what we can measure and what we would like to know about the product's behavioural attributes.

iii) This relationship is understood, has been validated and can be expressed in terms of a formula or model."

It is on these grounds that the metric to compare texts has been chosen.

4.2 An initial discussion

The product quality that requires measuring is the similarity of texts, and it is posited that a reliable metric of this is the number of similar sequences found between texts purporting to be about the same event or events. Whether this property can be measured accurately depends on how well the metric matches Kitchenham's assumptions :

i) It is without doubt that the property chosen by the metric (the number of congruent sequences between a pair of texts) can be measured accurately. It is a simple matter of counting them.

ii) It is suggested that a relationship exists between this number and the texts' relative closeness. That relationship manifests itself as the number of sequences found being directly proportional to the closeness of the texts; the closer the texts are to one another, the more sequences will be found. Tests have been carried out in order to try to confirm this.

iii) Algorithms have been implemented which claim to represent the metric, and the algorithms have been subject to tests to validate them. Users have been asked whether the results as measured by the metric produced transparent and meaningful results.

4.3 Designing an experiment to confirm the metric

Kitchenham's assumptions were based around the validity of software metrics, but they are equally applicable wherever a numerical measure of a property is required. An experiment is required which will either confirm or refute the metric for task of robustly comparing text representation produced by the model.

As a control group, a series of naive human users will be presented with the same data (newspaper stories) as the model, and asked to go through the same steps as the model (finding event representations and the causal links between them). The results of this follow the discussion on the algorithms involved.

Some of the data will be artificially changed with a desired goal in mind (to make stories both closer together and further apart), and the metric will be retested, to see if it behaves as expected.

Finally, the parameters governing the metric will be altered to test its sensitivity and robustness.

4.4 Testing the metric

In order to see whether the text comparison metric varies as would be expected with the similarity of the texts in question, two tests were carried out.

i) First, the results of the metric on two unaltered stories were compared with the results of a pair of texts that had had detail removed to make them more similar. The test was then repeated with a pair of stories, one of which had a crucial detail changed (making the stories dissimilar), being compared with the original text. Finally, two nominally unrelated stories were compared.

ii) Secondly, unaltered texts were compared by adjusting the metric's performance parameters to simulate the level of noise tolerated by the model. These results are explained in section 3.2 below.

iii) Human users were then presented with the same data and the results compared with the metric's results.

4.4.1 Testing by data adjustment

Before texts were compared, changes were made to test the sensitivity of the metric to changes in the event structures produced. Clearly if the metric is operating correctly, as the texts move further apart, progressively fewer matching sequences should be found. The changes took the following forms :

i) Extraneous detail was removed from the relatively more detailed of the two accounts. Thus the texts moved closer together in semantic terms, and if the metric is working correctly, more matching sequences should be found. The texts compared in this manner were as follows :

> "*a driver was jailed for attacking a woman after she asked him why he was holding up the traffic. the driver threw the woman into a parked car and kicked her. she suffered a collapsed lung.*"

and

> "*a driver who attacked a woman after she asked him to stop holding up the traffic was jailed. the driver threw the woman onto a parked car and kicked her. she was found to have a collapsed lung.*"

The full text of the second story can be found in the results chapter.

ii) A crucial detail was changed in one of the texts, moving the texts further apart semantically. It is to be expected that the number of sequences found will decrease, particularly when causal holes are limited or not permitted at all. The texts compared here are as follows :

> "*the body of a boy has been found in a gutted garage. he had been sleeping rough with friends as an outdoor adventure. the boy is believed to have died from smoke inhalation after a fire started in the garage where the boys were using candles. the boy had told his parents that he was staying with a friend.*"

and

> "*a boy was injured in a fire at a garage after telling his parents that he was staying with a friend. the boy was found in the gutted building.*"

Originally, the second text read "*a boy has died....*".

The following results were obtained. An analysis of the results takes place after the table.

Changes made to texts	Sequences found before changes	Sequences found after changes	d(x, y) (before change)
Crucial detail changed (moving texts apart)	25 (2 holes)	3	0.977 (0.808)
	6 (1 hole)	0	1 (0.846)
	0 (0 holes)	0	1 (1)
Extraneous detail removed (Moving texts closer)	22 (2 holes)	10	0.623 (0.887)
	12 (1 hole)	6	0.586 (0.831)
	4 (0 holes)	4	0.529 (0.826)
Unrelated stories Compared	0 (2 holes)	0	1
	0 (1 hole)	0	1
	0 (0 holes)	0	1

i) The first test compared two stories, one of which was artificially changed to move the texts further apart. As the table shows, there was a dramatic decrease (from 25 to 3) in the number of sequences found. This is to be expected; if the stories are made less similar, it is reasonable to assume that the algorithm will find less congruent sequences. The metric values also indicate the stories are less similar, rising to close to 1 and 1, from 0.808 and and 0.846 respectively.

ii) The second test involved removing extraneous detail from one of the texts. Although the number of sequences found did fall (from 22 to 10), this can be explained by the pattern of results. The same sequences were found with 0, 1 or 2 causal holes permitted. The texts matched so closely that there was no room for causal holes. Therefore to judge the algorithm, it is important to compare where there are fewer causal holes; the model finds as many congruent sequences despite working with less data, comfirming the hypothesis. The metric values once again lend confirmation, with the values falling from the range 0.826-0.887 to the range 0.529-0.623, indicating less difference between the texts.

iii) Finally, the model was re-run comparing one story with another totally unrelated to it, to check the metric does not mistakenly find matches where none exist. The model found no congruent sequences. The metric returned the maximum possible difference value of 1.

4.4.2 Adjusting the noise tolerated by the model

In order to test the model whilst not altering the input data, 'noise' was simulated by changing the number of holes permitted in the comparisons. If the metric is working correctly, the number of sequences found will vary in direct relation to the amount of noise permitted.

Story	Min. Sequence size	No. Gaps allowed	Sequences found	$d(x, y)$
Road rage	3+	2 or less	22	0.887
	3+	1 or less	12	0.831
	3+	0	4	0.826
Taxi stories	3+	2 or less	9	0.723
	3+	1 or less	5	0.677
	3+	0	1	0.230
Garage stories	3+	2 or less	25	0.808
	3+	1 or less	6	0.846
	3+	0	0	1

The table shows that as the permitted number of holes is decreased, the number of sequences found also decreases, matching the behaviour that is expected for the sequence matching algorithm, and fulfilling Kitchenham's second assumption. The metric numbers are possibly slightly contradictory here. The Road Rage stories appear to be quite different (metric values of 0.826-0.887); however this is due to one story having more information in it, and therefore many causal paths through it, affecting the value of the metric. The Taxi stories exhibit the same sort of behaviour, although demonstrate it rather better. As the amount of artificially induced noise is decreased, the metric measurement of the difference falls rapidly, from 0.723 to 0.230. The results of the Garage stories mirror those of the Road Rage stories; one story has much more information in it, and therefore many more causal links and paths through it, causing a disproportionate number of possible sequences, and adversely affecting the metric's behaviour.

The metric is also concerned with the length of the sequences that are found. It is clearly to be expected that as the minimum sequence length that is of interest is increased, the number of sequences found should in turn decrease. The following table demonstrates the results found when this hypothesis was tested by altering the minimum length the model looked for and counting the resultant sequences :

Story	Min. sequence size	No. Gaps allowed	Sequences found	d(x, y)
Road Rage	3+	2 or less	22	0.887
	4+	2 or less	3	0.974
Taxi stories	3+	2 or less	9	0.723
	4+	2 or less	0	1
Garage stories	3+	2 or less	25	0.808
	4+	2 or less	3	0.857

As can be seen, increasing the minimum relevant sequence length has an immediate and drastic effect on the number of sequences found, as is to be expected. The metric values confirm this; by increasing the minimum length of sequence required, fewer matches will be found, and the stories deemed less similar.

These results confirm the belief that the implemented metrics will behave as expected, finding less similarity between stories as less tolerance of 'noise' is allowed.

4.4.3 Adjustments to the metric itself

While metric I appears to accurately reflect trends (ie it decreases as texts get closer together, and *vice versa*), it is relatively insensitive to the number of sequences involved in its calculations and therefore, two modified versions of the metric were also tested.

The following changes to the metric have been implemented :

i) $d(x, y) = n(x, x) + n(y, y) - 2n(x, y)$. eg as before, but multiplied through by the denominator. This becomes metric II.

ii) $d(x, y) = (1 - 2n(x, y) / [n(x, x) + n(y, y)]) * \_(x, y)$, where $\_(x, y)$ represents the number of sequences that fail to match from one text to the next. This becomes metric III.

Metric II requires no proof, as it is simply an open-ended re-arrangement of metric I. The proof of the suitability of metric III follows.

4.5 Proof of Metric III

Let x and y be texts. $d(x, y)$ represents the distance between texts.

$d(x, x) = 0$.

Proof

If $x = y$, metric III can be re-written :

$d(x, x) = [1 - 2n(x, x) / 2n(x, x)] * \_(x, x) = 0$.

$d(x, y) >= 0$.

Proof

Let $d(x, y) = 0$ but $x \neq y$.

Since $x \neq y$, $\exists$ sequence S such that $S \in x$, $S \notin y$.

Therefore $\_(x, y) \neq 0$. The proof for the rest of the metric is as metric I.

Therefore $d(x, y) > 0$.

The comparison of the results of all three metrics follows.

Story	Metric I	Metric II	Metric III
Original story	0.808 (2 holes)	105	102.616
	0.846 (1 hole)	31	33.84
	1 (0 holes)	19	16
Moved apart	0.977 (2 holes)	149	145.573
	1 (1 hole)	43	40
	1 (0 holes)	19	16
Original story	0.887 (2 holes)	346	299.806
	0.831 (1 hole)	118	96.396
	0.826 (0 holes)	38	28.084
Moved closer	0.623 (2 holes)	33	12.46
	0.586 (1 hole)	17	4.688
	0.529 (0 holes)	9	2.116

4.6 Comparing the metrics

As the above results show, all three metrics appear to give reliable *relative* measures of the text differences. Two cases were considered for each metric :

i) Two stories artificially adjusted to be semantically further apart,

ii) Two stories artificially changed to be semantically closer together.

4.6.1 Artificially separated texts

Consider first the stories made more 'different'. Both metrics II and III appear to be saying that as the number of causal holes permissible is reduced, the texts are getting closer together. This is obviously not in fact true, but a function of the fewer sequences that will match if less slack is permitted in the system. When comparing metric II across the texts, it is clear that the semantic widening has been reflected in the weights returned by the metric : a difference of 105 becomes 149, a difference of 31 becomes 43. Because metric I set the absolute maximum distance of 1 between the texts where no causal holes are permitted, both metrics II and III reflect no further divergence between the texts; values of 19 and 16 respectively remain constant. This is because no congruent sequences were found in either case, and metrics based on the number of sequences will therefore return the same values.

4.6.2 Converged texts

Consider now the case of the artificially converged texts. The relative differences in the values of metrics II and III are huge and indicate that the texts are semantically closer together, as reflected in the number of sequences found. Metric II returns a value of 33 on the altered stories, whereas the difference between the original stories was valued at 346. Likewise, metric III originally scored very highly (299.806) prior to the texts being changed, the resultant value being 12.46, indicating much more similar structures.

It needs to be borne in mind that there is something of a combinatorial explosion taking place as more causal holes are permitted, and therefore the values of metrics II and III where 2 holes are permitted could be viewed as being implausibly high. However, the advantage of this is that metrics II and III are much more sensitive as the number of causal holes is reduced than metric I, and therefore could be considered to be better measures of semantic difference between the texts.

4.7 Does the metric fulfil the necessary criteria?

Consider first the mathematical definition of a metric. The distance function d is representative of the similarity of the texts.

i) $d(x, y) >= 0$. If the distance between the texts is greater than 0 (ie the texts are said to be different), a sequence of events should exist in one text which does not exist in the other. All three metrics represent this.

ii) $d(x, y) = 0$ iff x = y. In order to fulfil this condition, the texts must be the same. Sequences will exist in which one text is wholly contained within the causal graph of the other text. If a sequence that exists within one text does not exist within the other, all three metrics will reflect a difference of greater than 0.

iii) $d(x, y) = d(y, x)$. The order in which the texts are passed to the algorithm representing the metric is unimportant.

Consider next Kitchenham's assumptions. The first assumption is easily fulfilled - a property deemed to represent the texts' similarity (the number of congruent sequences) can easily be measured, simply by counting them.

The results as displayed above indicate that if the texts are moved artifically closer together, the number of sequences found in both texts increases (or at least finds as many sequences from less data). Thus, the measured property (the number of sequences) does indeed act as we would expect if the relationship between that property and what we wish to know (how similar are the texts?) is as stated in point ii) above.

The relationship was validated by questioning a series of naive users as to its usefulness and transparency. It was deemed to be both a useful measure and clear enough to understand simply. The above results were produced from computer models of the expected relationship; the fact that they produce the results we would expect leads to the fulfilment of Kitchenham's third assumption.

5. Conclusion

It can be concluded that the original metric is a suitable measure of the closeness of two texts. It fulfils both Giles' definition and Kitchenham's assumptions and has borne out well when compared with the results produced by naive users.

Further, the adjustments to metric I (as described in metrics II and III) make metric I more responsive to changes in the model parameters, and make differences between the texts more evident. Although the numbers produced by the metric have meaning only with respect to each other, they do appear to give an accurate measure of the relative 'closeness' of a pair of texts.

Therefore, while all three metrics are apparently suitable for use in measuring text differences, metrics II and III are superior to metric I.

6. Standard stories

If direct comparison is analogous to Abell's concept of generalising across texts, then the use of standard stories is an attempt to put the abstraction process to further use. Abell speaks of applying his theory to ask "How similar is this event for a generation of similar events?". In order to answer this question, it is necessary to guard against the possibility (however small) that our input data set (ie the group of narratives) is consistently skewed in some way, particularly in the case of multiple interviews coming from the same source. Whilst the data is treated on a value-free basis, it is only possible to answer Abell's question by building up a history of the sort of actions which traditionally transpire in this type of story.

The procedure for doing this is conceptually very simple :i) The domain of the input data is decided. This step is independent of the results stage, and is done during the analysis of the text.

> ii) The events identified from the input text are compared to a series of events deemed typical of the domain the story is in. For instance, a *Crime* story is likely to include an eventual *Arrest*, an international incident may often contain allegations of *Spying*, and so on.

In this way, a degree of the typicality of the story in its domain can be deduced. Such an approach is initially open to the perturbations caused by input data of an extreme nature as any other method, but if a successful method of evolving the standard scripts can be performed, then statistically speaking as more data is added in each domain (ie the sample size increases), the patterns of events to be found in the database will come more closely to reflect a typical story in this domain. Speaking statistically once again, as the sample size (the number of texts entered in a particular domain) increases, the standard error of the data decreases, and the mean (pattern, in this case) becomes an unbiased estimator for the population mean (true domain-typical events).

Standard story comparison is set up only to operate at the most general level of looking at patterns of behaviour, because of the raw material it works with - increasingly abstract (ie less specific) accounts of an action. Although an abstract text can discard superfluous detail it cannot change its theme in the broadest sense, hence it is only necessary to look at this most general level.

CHAPTER FIVE :
RESULTS

Example runs of the model

1. Introduction

This chapter displays some example runs-through of the model and the results of the naive user test. The purpose of this chapter is threefold :

i) to illustrate the types of input and output that the user is expected to deal with,

ii) to demonstrate the usefulness of the results procedures discussed in chapter 4,

iii) to give examples of the kinds of texts the model is currently dealing with.

Model-generated output is shown in *italics* and user input in *`italicised courier`*.

2. Example 1 - Text 1

"a boy has died in a fire at a garage after telling his parents that he was staying with a friend. the boy was found in the gutted building."

2.1 Event Representations created

1 : the boy tells the parents past movement of the friend

2 : the uncontrolled burning in the garage

3 : the boy was killed because of the uncontrolled burning in the garage

4 : the finding of the boy in the building

2.2 Setting of text's domain

The story fails to fire any domain currently in the model, and the user is taken through the steps required to set up a new domain for this type of story.

No domain!

Enter a suitable domain name (TERM to terminate) : *`accidental death`*

Adding new domain labelled accidental death to set!

Setting up standard text elements...

1 : the boy tells the boy's parents the boy had the boy moved to the friend

2 : uncontrolled fire

3 : the boy was killed because of uncontrolled fire

4 : the finding of the boy in the building

Enter standard element number (-1 to terminate) : 2

Adding element to new domain!

This process is repeated until the user decides that all events representative of the new domain have been added to it's structure.

2.3 Example 1 - Text 2

"the body of a boy has been found in a gutted garage. he had been sleeping rough with friends as an outdoor adventure. the boy is believed to have died from smoke inhalation after a fire started in the garage where the boys were using candles. the boy had told his parents that he was staying with a friend."

2.4 Event Representations created

1 : the boy had the boy tells the parents past future movement of the friend

2 : the boy had slept rough in the friends

3 : the uncontrolled burning

4 : past the choking

5 : the boy was killed because of past the choking

6 : the finding of the boy in the garage

7 : the suspicions that the boy was killed because of past the choking

2.5 Setting the text's domain

Because of the information supplied by the user during the model's run through the last text, the model correctly deduces the domain of the present text :

This story appears to be in the accidental death...

domain.Is this ok?

(Y/N) ?y

The model then identifies the elements contained within the text which are deemed typical of it's domain.

Following matches to element 2 in standard script :

past the choking

Following matches to element 3 in standard script :

the finding of the body in the garage

2.6 Comparing the two representations

First, the event representations that match directly are sought :

Setting match at pos : 1 and 1!

Following event occurs in all texts :

the boy tells the parents past movement of the friend

Setting match at pos : 2 and 3!

Following event occurs in all texts :

the uncontrolled burning in the garage

Setting match at pos : 3 and 5!

Following event occurs in all texts :

the boy was killed because of the uncontrolled burning in the garage

Setting match at pos : 4 and 6!

Following event occurs in all texts :

the finding of the boy in the building

Once these matches are found, sequential comparisons are done via the causal links made between event representations. The results are printed from the longest possible sequence found, to the shortest. The negative numbers are 'holes' (as explained on page 81) in the matching sequences :

-99 1 -2 3 -4 5 6 corresponds with event series

the boy had the boy tells the parents past future movement of the friend

the uncontrolled burning

the boy was killed because of past the choking

the finding of the boy in the garage

This (the longest sequential match) matches all of the event representations in the first story with corresponding elements in the second story. Therefore, the second story is a superset of the first story. The model also lists any other sequences it finds; they have been omitted here.

3. Example 2 - Text 1

"*police are searching for a student who disappeared after leaving a nightclub. the girl failed to return to her room at a hotel where she works as a waitress. the police want to trace a man who picked her up when she flagged him down. her parents have been contacted.*"

3.1 Event Representations created

1 : the student moved from the nightclub

2 : the girl didn't moved to the room

3 : the girl flagged down the man

4 : the man gave a lift the girl because of the girl the flagging down of the man

5 : the police wants the police finds the man

6 : the police searched the student

7 : the police tells the parents

3.2 Example 2 - Text 2

"the police are trying to trace a student who failed to return after leaving a disco. she telephoned the hotel where she had been staying and asked for an alarm call. the police have appealed for her to contact relatives. she is believed to have flagged down a taxi before getting into a car with a young man."

3.3 Event Representations created

1 : the student moved from the disco

2 : the student went missing

3 : the student calls the hotel

4 : the student asks the student is woken up

5 : the student moved to the car

6 : the police wants the police finds the student

7 : the police tells the student future to discuss the relatives

8 : the suspicions that the student flagged down the taxi

3.4 Comparing the event representations

The following isolated event matches are found :

Setting match at pos : 1 and 1!

Following event occurs in all texts :

the student moved from the nightclub

Setting match at pos : 2 and 2!

Following event occurs in all texts :

the girl didn't moved to the room

Setting match at pos : 7 and 3!

Following event occurs in all texts :

the police tells the parents

Setting match at pos : 5 and 6!

Following event occurs in all texts :

the police wants the police finds the man

The sequential matcher turns up the following information :

-99 6 -7 7 corresponds with event series

the police wants the police finds the student

the police tells the student future to discuss the relatives

-99 6 7 corresponds with event series

the police wants the police finds the student

the police tells the student future to discuss the relatives

4. Example 3 - Text 1

"*a taxi driver was charged with dangerous driving after a passenger was killed when his cab plunged into a river.*"

4.1 Event Representations created

1 : the cab moved to the river

2 : the passenger was killed because of the cab movement to the river

3 : the taxi's driver drives recklessly

4 : the taxi's driver was charged with the taxi's driver drives recklessly

4.2 Setting the text's domain

Once again, the text fails to trigger any domain currently existent in the model, and the user is taken through the steps to create a new domain suited to the text.

It is necessary to manually set the domain, if any is suitable.

1 : International relations

2 : Crime

3 : Strikes

4 : Pursuit

Enter choice (-1 to terminate) : `-1`

No domain!

Enter a suitable domain name (TERM to terminate) : car accident

Adding new domain labelled car accident to set!

Setting up standard text elements...

1 : the driver drives recklessly

2 : the driver's cab moved to the river

3 : the passenger was killed because of the driver's cab movement to the river

4 : charged the driver with the driver drives recklessly

Enter standard element number (-1 to terminate) : `1`

Adding element to new domain!

4.3 Example 3 - Text 2

"*a taxi driver was charged with dangerous driving after a passenger was killed when the taxi plunged into a river. the driver escaped with shock. the passenger died after being trapped in the submerged taxi.*"

4.4 Event Representations created

1 : the taxi moved to the river

2 : the entrapment of the passenger

3 : the driver escaped

4 : the passenger was killed because of the taxi movement to the river

5 : the taxi's driver drives recklessly

6 : the taxi's driver was charged with the taxi's driver drives recklessly

4.5 Setting the text's domain

This story appears to be in the car accident...

domain.Is this ok?

(Y/N) ?y

The model finds the following elements in the text to be typical of stories in its domain :

Following matches to element 2 in standard script :

the passenger was killed because of the taxi movement to the river

Following matches to element 1 in standard script :

the taxi's driver drives recklessly

Following matches to element 3 in standard script :

the taxi's driver was charged with the taxi's driver drives recklessly

4.6 Comparing the event representations

Once again, the isolated matches are searched for first :

Setting match at pos : 1 and 1!

Following event occurs in all texts :

the cab moved to the river

Setting match at pos : 2 and 4!

Following event occurs in all texts :

the passenger was killed because of the cab movement to the river

Setting match at pos : 3 and 5!

Following event occurs in all texts :

the taxi's driver drives recklessly

Setting match at pos : 4 and 6!

Following event occurs in all texts :

the taxi's driver was charged wwith the taxi's driver drives recklessly

Then the sequential check is carried out. Event though all the elements of the first story are included in the second story, as the above output shows, the different causal links found mean that no sequence is found in the second story which fully includes the first story:

-99 1 4 -5 6 corresponds with event series

the taxi moved to the river

the passenger was killed because of the taxi movement to the river

the taxi's driver was charged with the taxi's driver drives recklessly

-99 1 4 6 corresponds with event series

the taxi moved to the river

the passenger was killed because of the taxi movement to the river

the taxi's driver was charged with the taxi's driver drives recklessly

5. Example 4 - Text 1

"a woman who nearly died after taking an ecstasy tablet has been injured in an accident. officers recovered a quantity of white powder and two tablets from the car. she fractured a pelvis when her car collided with a car in cambridge. after the accident the woman was taken to the hospital where she provided a negative breath test."

5.1 Event Representations created

1 : the woman eats the tablet

2 : the woman was hurt because of the woman the eating the tablet

3 : the crash

4 : the woman had health state : -1

5 : movement of the woman to the hospital

6 : breath test

7 : the officers finds the drugs

8 : the officers finds the tablets

5.2 Setting the text's domain

This story appears to be in the car accident...

domain.Is this ok?

(Y/N) ?y

Following matches to element 2 in standard script :

the woman was killed because of the woman the eating the tablet

5.3 Example 5 - Text 2

"a woman who nearly died after taking an ecstasy tablet has been hurt in an accident. two pills and some white powder have been sent for analysis. she suffered a fractured pelvis when the car she was driving was involved in a collision. the driver of the other car suffered a broken foot. the woman gave a negative breath test."

5.4 Event Representations created

1 : the woman eats the tablet

2 : the woman was killed because of the woman the eating the tablet

3 : reckless driving in the car

4 : the crash

5 : the woman had health state : -1

6 : breath test

7 : the foot had health state : -1

8 : to discuss the pills analysis

5.5 Setting the text's domain

The model again correctly predicts the domain of the story from amongst those that it is already aware of.

This story appears to be in the car accident...

domain.Is this ok?

(Y/N) ?y

Following matches to element 2 in standard script :

the woman was killed because of the woman the eating the tablet

Following matches to element 1 in standard script :

reckless driving in the car

5.6 Comparing the two Event Representations

The direct comparison of the texts is done in isolation initially.

Setting match at pos : 1 and 1!

Following event occurs in all texts :

the woman eats the tablet

Setting match at pos : 2 and 2!

Following event occurs in all texts :

the woman was killed because of the woman the eating the tablet

Setting match at pos : 3 and 4!

Following event occurs in all texts :

the crash

Setting match at pos : 4 and 5!

Following event occurs in all texts :

the woman had health state : -1

Setting match at pos : 6 and 6!

Following event occurs in all texts :

breath test

The sequential comparison procedure then produces the following results :

-99 1 2 corresponds with event series

the woman eats the tablet

the woman was killed because of the woman the eating the tablet

-99 4 5 6 corresponds with event series

the crash

the woman had health state : -1

breath test

-99 4 5 corresponds with event series

the crash

the woman had health state : -1

-99 5 6 corresponds with event series

the woman had health state : -1

breath test

6. Naive user tests

A series of tests of the model with unfamiliar users was carried out, in order to ascertain the following :

i) The degree to which a new user of the model can quickly get useful information from its execution,

ii) The suitability and clarity of the results procedures.

These tests took the following forms :

i) An execution of the model in which all new information was pre-supplied to the user.

ii) A test of the ease by which a series of user-defined structures were created.

6.1 A test run of the model

The users were presented with two texts which they were required to run through the model, answer any questions it may present them with, and comment on the transparency of the results. The following sections displays the results as obtained by an informed user of the model. The naive users were presented with all the necessary structures and these texts, and were asked to execute and comment on the model.

6.1.1 Text 1

"a driver was jailed for six years for attacking a woman after she asked him why he was holding up the traffic. the driver threw the woman into a parked car and kicked her. she suffered a collapsed lung."

6.1.1.1 Event Representations produced

1 : the woman asks the driver had delay driving

2 : the driver moved the woman to the car

3 : the driver fights the woman

4 : the woman had health state : -1

5 : the jail term the driver because

the driver fights the woman

6.1.1.2 Setting the text's domain

The model selects a previous domain as created by the user, which is accepted.

This story appears to be in the assault...

domain.Is this ok?

(Y/N) ?y

Following matches to element 1 in standard script :

the driver fights the woman

6.1.2 Text 2

"a driver who attacked a woman after she asked him to stop holding up the traffic was jailed. the driver threw the woman onto a parked car and kicked her before driving off. she was taken to the hospital where she was found to have a collapsed lung."

6.1.2.1 Event Representations created

1 : the driver delays driving

2 : the woman asks the driver the driver ends delay driving

3 : the driver moved the woman to the car because of the request to the driver ends delay driving

4 : the driver fights the woman because of the request to the driver ends delay driving

5 : movement of the woman to the hospital

6 : the finding of the woman the woman had health state : -1

7 : the driver went missing

8 : the jail term the driver

6.1.2.2 Setting the text's domain

This story appears to be in the assault...

domain.Is this ok?

(Y/N) ?y

Following matches to element 1 in standard script :

the driver fights the woman because of the request to the driver ends delay driving

6.1.3 Direct Comparison results

Initially, the texts are compared by looking at their isolated elements :

Setting match at pos : 1 and 2!

Following event occurs in all texts :

the woman asks the driver had delay driving

Setting match at pos : 2 and 3!

Following event occurs in all texts :

the driver moved the woman to the car

Setting match at pos : 3 and 4!

Following event occurs in all texts :

the driver fights the woman

Setting match at pos : 5 and 8!

Following event occurs in all texts :

the jail term the driver because

the driver fights the woman

The model is then asked to list any sequences with more than 3 elements that are to be found via the causal links within the texts :

-99 2 3 4 -5 -6 8 corresponds with event series

the woman asks the driver the driver ends delay driving

the driver moved the woman to the car because of the request to the driver ends delay driving

the driver fights the woman because of the request to the driver ends delay driving

the jail term the driver

-99 2 3 4 -5 -6 8 corresponds with event series

the woman asks the driver the driver ends delay driving

the driver moved the woman to the car because of the request to the driver ends delay driving

the driver fights the woman because of the request to the driver ends delay driving

the jail term the driver

-99 2 3 4 -5 -6 8 corresponds with event series

the woman asks the driver the driver ends delay driving

the driver moved the woman to the car because of the request to the driver ends delay driving

the driver fights the woman because of the request to the driver ends delay driving

the jail term the driver

-99 2 3 4 -5 -6 8 corresponds with event series

the woman asks the driver the driver ends delay driving

the driver moved the woman to the car because of the request to the driver ends *delay driving*

the driver fights the woman because of the request to the driver ends delay driving

the jail term the driver

What these results tell us is that there are 4 matching paths to be found if the minimum permitted sequence is restricted to 4 places, and 2 'holes' in the sequence are allowed. The apparent repetition of the result is due to the model finding alternate ways to match the 4 true events by placing the 'holes' in different positions during the analysis.

6.2 User comments

The naive users encountered problems in a variety of areas of the model. Some of these problems were foreseeable, some were not. Some of the users had difficulty understanding why such a method exists or needs to be computerised; these questions have been addressed elsewhere in the thesis. The more relevant areas of concern that the new users identified are detailed in the following sections.

6.2.1 User interface

Although there is no formal user interface, the model prompts the user at various points to accept or refute various deductions that it makes. The users didn't find any serious problems with the types of questions being asked (almost all of which are 'yes/no' questions), but did have some problems with the data presented to help make this decision. Partly this was because of incomplete information (see point 6.2.2 below), but the lack of a language generator in the model contributed largely to this. The model simply tries to display a general idea of the structures which have been created, with no consideration of the well-formedness of the language produced.

This is clearly a serious problem, as it hinders the communications between the model and the user. However, the creation of a language generator is not by any means an insurmountable problem; it was not implemented because other more important issues were addressed, and a very crude but basically usable alternative was found.

6.2.2 Incomplete Information

Because the model compares event references as they are created, the user is sometimes presented with unfinished information. For instance, the following example created problems for some of the users :

New event :

the driver fights the female represented by the following pronoun :

her

Old event :

the driver fights the woman because of the request to the male represented by the following pronoun : he had the male represented by the following pronoun : he delays driving

The pronoun hasn't been resolved at this point, hence the slightly unwieldy output which replaces the correct actor. The output seemed clearer to them on re-reading, but clearly the syntax used by the program requires either some rethinking or more serious user-exposure.

The users did not have problems with the general form of the questions *per se*. They were told in advance (and during the test) of the type of questions that they would be expected to answer, and that these questions would be 'yes/no'. Their complaints were largely concerned with Imperfect Information (see previous section).

6.2.3 'Expert' Information

Some information that was not deemed as being expert turned out to be so. For instance, not all the test subjects were aware of the clauses of a sentence and their relations to one another. When creating a new event, for example, they had difficulty in assigning the relevant conditions to it because of their lack of knowledge of the source text's relevant parts. The solution to this problem is in two parts :

1) Information concerning the relevant parts of the text is available and, given a suitable user interface, could be presented to the user without difficulty. Because of the lack of a window-based environment, this can presently only be done in an unwieldy manner.

2) It is not unreasonable to assume that someone using the model to analyse textual data will have some familiarity with common structures contained within their texts. This is especially true when using a specific tool, such as Abell's Comparative Narratives. Some of the problems that the users faced were undoubtedly because of their unfamiliarity with the theory. Even though the Results stage presented no difficulty to the users, some of them had a problem comprehending why anyone would want to analyse texts in this way.

6.2.4 Creating new structures

Although the creation of new structures was felt to be easy, some of the users could not see the relationship between these new structures and the rest of the model. Some of the users had problems when applying conditions to new structures because they were unaware of the constituents of a clause. Others felt that the formalism used in the creation of new plans, actors and events would not be complex enough to deal with large-scale real world problems. There is an element of truth in this. However, the expandability of which these algorithms are a part was added at a relatively late stage of the project and is therefore less well-developed than other aspects of the model. The relative simplicity of the structures in question does not undermine the basic model.

6.2.5 Results Procedures

The output produced by the results procedures was generally felt to be self-explanatory. Once again, the notation was initially slightly confusing, and has been revised in light of this. For instance, the initial sequence of numbers in the following example output are basically superfluous, the information content of the output being carried in the text :

-99 2 3 4 -5 -6 8 corresponds with event series

the woman asks the driver the driver ends delay driving

the driver moved the woman to the car because of the request to the driver ends delay driving the driver fights the woman because of the request to the driver ends delay driving

the jail term the driver

The direct matching algorithm output presented no problem in comprehension.

The users felt that the information that the results were producing was transparent enough to be quickly digested, although some explanation of the mechanisms available to filter and reduce this information was necessary. Information such as this could be easily presented in a user manual, which has not been written to date.

6.3 Discussion

Many legitimate concerns were identified during the user test. Some of these can be put down to the inevitable unfamiliarity that occurs whenever a new piece of software is run for the first time. Others are more pressing, and fall into two broad categories :

i) *Notation and output.* The user interface, such as it is, is not user-friendly. This is largely because the program runs from the command line, and not a windowing environment. However, given its status as a prototype program, the author does not propose to do anything about this point. Similarly, the notation can be altered and simplified; this task has begun. There will nevertheless remain a certain amount of notation that is unavoidable, that a user will become familiar with by running the program regularly.

The question of a language generator is not so glibly answered however, and presents an obstacle in the understanding of the output produced by the program. Work has now begun on a simple language generator.

ii) *Expert Knowledge*. The users' remarks on the usefulness of both the theory and implementation of Comparative Narratives are not to be taken lightly, but the users were drawn from random backgrounds, possibly totally unconnected even with the use of computers. Therefore, their comments have no direct bearing on the question at hand : how easy is it to use the computer model?

Similarly, the users' lack of knowledge concerning the structure of a piece of text is perhaps not of central concern, as a user who wished to analyse some texts would presumably do so from a point of having at least a little knowledge in a relevant field.

Finally, the expert knowledge needed to devise new structures has shown that they are easy to create and subsequently manipulate. The anxiety over whether they are sufficiently complex may have some grounding to it, but as was explained above, adding more conditions and slots is not a major programming task, and this chore could be removed completely by implementing a parser allowing the user to configure the setup themselves. No work has been done on this. By forcing the user to accept a simple methodology, the model has no need to become bogged down dealing with extremely case-specific conditions that may never occur again.

7. Conclusions

The example runs dissected at the start of the chapter have demonstrated that the adjusted version of Abell's theory discussed in chapter three has been successfully implemented. Algorithms have been implemented to find causal links in a piece of text, and produce easily interpreted but meaningful results. The model has been made expandable.

The naive user test has shown that, with a little initial tuition, the model can be used on simple examples by a previously uninformed user. Needless to say, regular use of the model would improve the user's ability to get results relatively quickly. There are a number of problems that require attention before it can be considered a usable and useful tool, not least of which is the language generator object. The feedback from the user test suggests that the implementation of this single algorithm would aid comprehension of the model enormously. The naive user test also provided support for the suitability of the implemented comparison procedures.

CHAPTER SIX : CONCLUSIONS

Conclusions

1. Introduction

This work has detailed the implementation of a metric for the comparison of representations of text. To provide the raw material on which the metric works, a prototype computer model to perform Abell's Comparative Narratives has been implemented.

This work has outlined the extent of this computerisation and the adjustments and enhancements made to the theory required to bring the project to fruition. The theoretical and practical backgrounds of the tools used in the implementation has been explained.

The results of these event- and causal link- finding algorithms have been compared to the results obtained by human analysts, and have been found to be very similar, lending support to the behaviour of these algorithms.

The comparison metric has been evaluated by varying both the input texts and the parameters it operates under during the model execution and it has been found that the metric behaves as an analyst would qualitatively expect.

The model has been run by a series of untutored 'naive' users to test its ease of use, and with a number of qualifications has passed.

2. General points

The model produced as a result of this research is quite different in approach to the majority of computational tools for use in qualitative research. The emphasis in these other methodologies lies very much on assisting the user to perform the task. The model detailed in this work was made with the maximum useful automation in mind, whilst prompting the user to settle every actual decision that the model deems necessary to take. Whereas other qualitative schemes may tend to assist the user in deciding for themselves exactly what these decisions are, this Comparative Narrative model attempts to enclose the performance of the original theory in a "black box"; no particular knowledge of Abell's work is essential in operating the model, although obviously having some conception of the general principles it is attempting to espouse would be useful.

It is possible to easily, if somewhat crudely, add more information to the system in order to expand its knowledge of stereotypical situations. Tuning the system then becomes theoretically possible - where it is consistently making errors in meaning, more exacting examples can be supplied. This will cause the 'reasoning' behind the model to calculate differently the next time a particular story is passed through it, picking up on the extra detail available to it. Again, the computer model itself is supporting Abell's contention that the analyst has a role to play in shaping the narrative (although this is perhaps rather disingenuous; the analyst is forced to enrich the narrative until such a time as it can be processed successfully). In this way, the nature of the model is perhaps forcing the analyst to perform a more traditional job, that of removing 'noise' from the data before beginning the analysis.

The structural comparison step of the model produces results that currently require some interpretation; this is because of the format they are presented in, which is based around the computational structures they are derived from. The information that they are attempting to put across is in fact very transparent. Clearly there wasn't sufficient time in the course of the project to spend any time on the presentation of the code (by way of a graphical user interface or similar). However, a small investment of time here would work wonders for the general 'user-friendliness' of the code. This would also clearly assist the untutored, who faced not only the problem of interpreting the terse output, but also the rather larger mental hurdle of understanding the motivation and achievements of Abell's theory in the first place.

The comparison algorithms have provided a more flexible way of comparing the structures of texts than Abell's very rigorous maxim that 'each narrative must contain the *same actions* and second, these actions must be *inter-connected in identical ways'*. They allow for the somewhat rigid nature of computer models by permitting gaps in the data sets being compared, and they have been shown to fulfil Abell's dual concepts of *generalisation* and *abstraction*. They can be used to provide reasonable measures of the relative distances between pairs of texts both in terms of the structure of the texts compared to each other (abstraction), and also to the concept of a (admittedly user-defined and therefore hugely subjective) 'standard story' (generalisation - how typical is this story for a generation of similar stories).

The metrics that were presented to supply an element of objectivity to the model seem to behave as expected. Clearly this is hard to prove without resorting to the somewhat artificial techniques used in manipulating the texts - the only 'natural' way to do this would be to take a text that was a less-detailed version of a text already available, which is exactly what the program is trying to prove! However this potential source of doubt aside, the metrics appear to be both robust and meaningful. They do not attempt to 'score' a text in absolute terms, a goal which would appear to be a pipe-dream in anything other than a highly specialised tool, in terms of subject matter and scoring criteria. Rather, the aim is to be able to say, "does the underlying structure (as defined by Abell's work) of texts *v* and *w* suggest that they are closer together than texts *x* and *y*?

3. Further work

It is a truism to say that more work can always be done on a model, particularly a model in which the domain of the input information is significant on some level; this particular prototype is no exception to this rule.

The script and plan-based structures can always be expanded and made more comprehensive, as can the dictionary and semantic information used to create event structures. It is true that they could never be said to be finished because of the domain-driven nature of the model.

Perhaps most importantly, a method of causing the standard scripts to "evolve" as more data is added is required, along with a macro language to generalise the semantic interpretation of sentences by the parser. The results procedures so far implemented dovetail most usefully with this goal; the direct comparison procedure configured to run on sequences of events could conceivably be made to provide a database of causal information which could be analysed for events and patterns that occur with unusual frequency. However, caution will be required when undertaking this task because of the likelihood that some degree of (elementary) statistical analysis will be required - just how unusual is "unusual frequency"? Another possible analysis of this causal information could be a form of machine learning. The end of each sequence is considered to be its "goal state" and patterns are deduced from the sequence of precursors that lead to each goal state.

4. Conclusions

This thesis has shown that it is possible to produce a computational version of Abell's theory of Comparative Narratives. The computer model has the advantages of repeatability and robustness over the theoretical model, due to the higher specification various aspects of the theory underwent.

The original theory has received a number of changes to those areas in which it was not highly specified, either because such processes come naturally to human beings or because the original theory was vague in certain respects. The behaviour of the algorithms used to represent these changes has compared favourably with the results found by human analysts.

A series of heuristics to find causality amongst event representations were designed and implemented. This set encompasses both domain-dependent heuristics and more general routines usable on potentially any piece of action-driven text. The domain-dependent heuristics are simple enough in structure to mean that their counterparts in other domains can be easily set up where required by the user.

A pair of procedures to compare the event representations drawn from each text were created, enabling texts to be compared whether or not they come from the same or different domains, or have no domain associated with them at all. The procedures produce easy to understand results, with no reliance on complex and abstract mathematics. Once again, these procedures are created under the auspices of the model, and comparison structures for texts in new domains are easily and quickly formulated. Metrics were implemented which gave relative measures of the 'distance' between two texts. In tests, the results produced by the metrics were found to be in line with the expectations of the user.

Due in part to the object-oriented nature of the model, a high degree of expandability was built into the model, meaning that the changes made to Abell's theory to make it more potentially useful can be exported to other as yet unknown domains. The method by which these procedures are implemented is simple and easy for an inexperienced user to follow, and in many cases completely automatic.

The implemented model does demonstrate that, with certain attached conditions and modifications as explained above, Abell's theory of Comparative Narratives is very much suited to a computational transformation, and that with the application of results metrics, it can be used to meaningfully compare texts. The model also shows that such a transformation can be made by a combination of standard text-analytic tools and new task-specific algorithms, and that both old and new methodologies can make a successful and natural metamorphosis into an object-oriented framework.

This project has shown that, by adding robust results and causal linking procedures, Abell's theory can be used to produce meaningful and transparent comparisons of a series of text-representative structures, and that it lends itself well to the implementation of metrics designed to test the robustness of this behaviour.

REFERENCES

References

Abbott, A - 'Measure for Measure - Abell's Narrative Methods', *The Journal of Mathematical Sociology*, 18 (2-3), 203 - 214, Ed. Patrick, Gordon & Breach (1993).

Abbott, A and Hrycak, A - 'Measuring resemblance in Sequence Data: An Optimal Matching Analysis of Musicians' Careers', *American Journal of Sociology*, 96 (1), 144 - 185, University of Chicago Press (July 1990).

Abell, P - An Example text, *Connections*, 12 (3), Ed. Susan Greenbaum, INSNA (winter 1989).

Abell, P - 'Narrative Method - A reply', *The Journal of Mathematical Sociology*, 18 (2-3), 253 -, Ed. Patrick Doriean, Gordon & Breach (1993).

Abell, P - 'Some Aspects of Narrative Method', *The Journal of Mathematical Sociology*, 18 (2-3), 93 - 134, Ed. Patrick Doriean, Gordon & Breach (1993).

Abell, P - '*The Syntax of Social Life : The Theory and Method of Comparative Narratives*', Oxford University Press (1987).

Abell, P - '*The Theory and Method of Comparative Narratives*'.

Abraham, C - 'Seeing the Connection in Lay Causal Comprehension : A Return to Heider', *Contemporary Science and Natural Explanation : Commonsense Conceptions of Causality*, Ed. D.J Hilton, 145 - 174, Harvester, (1988).

Allen, J - 'Knowledge Representation', *Natural Language Understanding*, Benjamin-Cummings (1987).

Allen, J - 'Using World Knowledge', *Natural Language Understanding*, Benjamin-Cummings (1987).

Allen, J F - 'Actions and events in interval temporal logic', *Journal of Logic and Computation*, 4 (5), 531 - 581, (1994).

Allen, J F, Frisch, A M and Litman, D J - 'ARGOT : The Rochester Dialogue System', *Proceedings of the National Conference on Artificial Intelligence (AAAI 82)*, 67 - 70 (1982).

Alterman, R - 'An Adaptive Planner', *Proceedings of the National Conference on Artificial Intelligence*, (1986).

Antaki, C - 'Structures of Belief and Justification', *Analysing Lay Explanation : A Casebook of Methods*, Ed. C. Antaki, 60 - 73, (1988).

Antaki, C and Naji, S - 'Events Explained in Conversational "because" Statements', *British Journal of Social Psychology*, 26, 119 - 126, (1987).

Bandini, S, Carniel, B and Pomello, L - 'Narrative Context Model for Representing Deep Knowledge', *Cybernetics and Systems '88*, 869 - 876, Ed. R. Trappl (1988).

Bares, M, Canamero, D, Delannoy, J F and Kodratoff, Y - 'XPlans: case-based reasoning for Plan Recognition', *Applied Artificial Intelligence*, 8 (4), 617 - 643, 1994.

Bates, M - 'The theory and practice of Augmented Transition Network grammars', *Lecture notes in Computer Science 63 : Natural Language Communication with Computers*, Ed. Leonard Bolc, Springer-Verlag (1978).

Blum, A - '*Neural Networks in C++; an Object-Oriented Framework for Building Connectionist Systems*', John Wiley & Sons (1992).

Brewer, W F - 'Plan Understanding, Narrative Comprehension and Story Schemas', *Proceedings of the 1st National Conference on Artificial Intelligence (AAAI 82)*, 262 - 264 (1982).

Bruce, B - '*Case Systems for Natural Language,* Artificial Intelligence, 6, 327 - 360, (1975).

Carley, K - 'Content Analysis', *The Encyclopedia of Language and Linguistics*, Ed. R.E Asher et al. (1990).

Chaib-draa, B, Mandiau, R and Millot, P - 'Distributed Artificial Intelligence': An Annotated Bibliography', *SIGART Bulletin*, 3 (3), 20 - 37, (1992).

Chandrasekaran, B - 'Functional Representation : a historical perspective', *Applied Artificial Intelligence*, 8 (2), 173 - 197, (1994).

Charniak, E - 'On the Use of Framed Knowledge in Language Comprehension', *Artificial Intelligence*, 11, 225 - 265, (1978).

Charniak, E and Goldman, R P - 'Bayesian model for plan recognition', *Artificial*

Intelligence, 64 (1), 53 - 79, (1993).

Charniak, E and McDermott, D - '*Introduction to Artificial Intelligence*', Addison-Wesley, (1985).

Clarke, D D and Crossland, J - '*Action Systems*', Methuen, (1985).

Cody, M J and McLaughlin, M L - 'Accounts on Trial : Oral Arguments in Traffic Court', *Analysing Everyday Explanation : A Casebook of Methods*, Ed. C. Antaki, 113 - 126, (1988).

Conyers, T R - '*Object-Oriented Design and Programming*', BSc final year project, University of Greenwich (1992).

Cullingford, R E and Pazzani, M J - 'Word-meaning Selection in MultiProcess Language Understanding Programs', *IEEE Transactions on Pattern Matching and Machine Intelligence*, 493 - 509, (1984).

Dayhoff, J - '*Neural Network architectures, an introduction*', Van Nostrand Reinhold (1990).

Dehn, N - 'Story Generation after TALESPIN', *7th International Joint Conference on Artificial Intelligence '81*, 1, 16 - 18 (1981).

DeJong, G - 'An overview of FRUMP', *Strategies for Natural Language Processing*, Ed. Wendy Lehnert and Martin Ringle, Erlbaum Associates (1982).

Dobbins, R and Eberhart, R - 'Software Tools', *Neural Network PC Tools*, Ed. Russell Eberhart and Roy Dobbins, Academic Press Limited (1990).

Dunnette, M D - '*Handbook of Industrial and Organizational Psychology*', Rand Corp, (1976).

Dyer, M G - 'In-Depth Understanding: A Computer Model of Integrated Processing for Narrative Comprehension', MIT Press, (1983).

Dyer, M G, Wolf T C and Korsin M - 'BORIS - An In-Depth Understander of Narratives', *Proceedings of the 7th International Joint Conference on Artificial Intelligence* (1981).

Fararo, T - 'Generating Narrative Forms', *The Journal of Mathematical Sociology*, 18 (2-3), 153 - 182, Ed. Patrick Doriean, Gordon & Breach (1993).

Finin, T and Morris, G - 'Abductive Reasoning in multiple fault diagnosis', *Artificial*

Intelligence Review, 3 (2-3), 129 - 158, (1989).

Forbus, K - 'Qualitative Process Theory', *Artificial Intelligence*, 24, 178 - 219.

Forbus, K - 'The Qualitative Process Engine', *Readings from Qualitative Reasoning about Physical Systems*, 220 - 235, Ed. Daniel S. Weld and Johan de Kleer, Morgan Kaufman (1991).

Garvey, C and Caramazza, A - 'Implicit Causality in Verbs', *Linguistic Inquiry*, 5, 459 - 464, (1974).

Gergen, M - 'Narrative Structures in Social Explanation', *Analysing Lay Explanation : A Casebook of Methods*, Ed. C. Antaki, 94 - 112, (1988).

Gilb, T - '*Software Metrics*', Winthrop (1977).

Giles, J R - '*Introduction to the analysis of metric spaces*', Cambridge University Press (1987).

Grosz, B J and Sidner, C - 'Attention, Intention, and the Structure of Discourse', *Computational Linguistics*, 12 (3), (1986).

Halliday, M A K - '*Introduction to Functional Grammar*', Edward Arnold (1985).

Harré, R - Agentive Discourse, *Discursive Psychology in Practice*, 120 - 136, Ed. R. Harré and P. Stearns, Sage (London), (1995).

Harré, R and Stearns, P - `Psychology as Discourse Analysis', *Discursive Psychology in Practice*, 120 - 136, Ed. R. Harré and P. Stearns, Sage (London), (1995).

Harris, M - `*Introduction to Natural Language Processing*', Reston / Prentice Hall, (1985).

Haun, U, Schacht, S and Broeker, N - `Concurrent, Object-Oriented natural language parsing : The *ParseTalk* model', *International Journal of Human / Computer Studies*, 41 (1-2), 179 - 222, (1995).

Heise, D R - `Narratives Without Meaning?', *The Journal of Mathematical Sociology*, 18 (2-3), 183 - 190, Ed. Patrick Doriean Gordon & Breach (1993).

Heise, D R - `*Causal Analysis*', Wiley-Interscience, (1975).

Heise, D R - `*Understanding Events, Affect and the Construction of Social Action*', Cambridge University Press, (1979).

Heise, D R and Durig, A - 'A Frame for Organizational Actions and Macroactions', Forthcoming in the *Journal of Mathematical Sociology*, (1996).

Hirschman, L and Story, G - 'Representing Implicit and Explicit Time Relations in Narrative', *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, 1, 289 - 295 (1981).

Hirst, G - 'Existence assumptions in knowledge representation', *Knowledge Representation*, 199 - 243, Ed. Ronald J. Brachman, Hector J. Levesque and Raymond Reiter, Elsevier Science Publishers (1991).

Hockey, S - '*A Guide to Computer Applications in the Humanities*', Gerald Duckworth & Co, (1980).

Huyck, C R and Lytinen, S L - 'Efficient heuristic natural language parsing', *Proceedings of the 11th National Conference on Artificial Intelligence*, 386 - 391, (1993).

Iwasaki, Y, Vescovi, M, Fikes, R and Chandrasekaran, B - 'Causal Functional Representation with behaviour-based semantics', *Applied Artificial Intelligence*, 9 (1), 5 - 31, (1995).

Jacobs, P S and Rau, L F - 'Innovations in Text Interpretation', *Artificial Intelligence*, 63 (1- 2), 143 - 191, (1993).

Kass, A M - 'Adaption-Based Explanation : Extending Script / Frame Theory to Handle Novel Input', *Proceedings of the International Joint Conference on Artificial Intelligence '89*, 141 - 147, (1989).

Kautz, H - 'Generalised Plan Recognition', *Proceedings of the 5th National Confererence on Artificial Intelligence*, 32 - 37, (1986).

Kintsch, W and Van Dijk, T A - 'Toward a Model of Text Comprehension and Production', *Psychological Review*, 85, 363 - 394, (1978).

Kintsch, W and Van Dijk, T A - 'Cognitive Psychology and Discourse : Recalling and Summarizing Stories', *Current Trends in TextLinguistics*, 61 - 80, (1978).

Kitchenham, B - 'Software Metrics', *Software Reliability Handbook*, Ed. P. Rook, Elsevier (1990).

Kolodner, J L - '*Retrieval and Organisational Strategies in Conceptual Memory: A*

Computer Model', Lawrence Erlbaum Associates, (1984).

Kosaka, K - 'Towards a Further Analysis of Narratives', *The Journal of Mathematical Sociology*, 18 (2-3), 141 - 152, Ed. Patrick Doriean, Gordon & Breach (1993).

Kruskal, J B and Sankoff, D - 'An Anthology of Algorithms and Concepts for Sequence Comparison', *Time Warps, String Edits and Macromolecules*, Ed. D. Sankoff and J.B. Kruskal, Addison-Wesley (1983).

Laskowski & Hofman - 'Script-based reasoning for situation monitoring', *Proceedings of the Xth National Conference on Artificial Intelligence*, 819 - 823, (1987).

Lebowitz, M - 'Creating a Story-Telling Universe', *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, 63 - 65, Morgan Kaufman, (1983).

Lebowitz, M - 'Memory-Based Parsing', *Artificial Intelligence*, 363 - 404, (1983).

Leech, G - '*Meaning and the English verb*", Longman (1971).

Lehnert, W - 'Plot Units : a Narrative Summarisation strategy', *Strategies for Natural Language Processing*, Ed. Wendy Lehnert and Martin Ringle, Erlbaum Associates (1982).

Lehnert, W, Dyer, M G, Johnson, P, Yang, C and Harley, S - 'BORIS : an Experiment in In-Depth Understanding of Narratives', *Artificial Intelligence*, 20, 15 - 62, (1983).

Lehnert, W G, Black, J B, Reiser, B J - 'Summarising Narratives', *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, 1, 184 - 189, (1981).

Luh, C-J - 'Modelling and simulation in an object-oriented environment', *Journal of Information and Software Technology*', 36 (6), 343 - 352 (1994).

Luria, M - 'Dividing Up the Question Answering Process', *Proceedings of the 1st National Conference on Artificial Intelligence (AAAI 82)*, 71 - 74 (1982).

Mandler, J M and Johnson, J S - 'Remembrance of Things Parsed: Story Structure and Recall', *Cognitive Psychology*, 9, 111 - 151, (1977).

McArthur, L Z - 'The How and What of Why : Some Determinants and Consequences of Causal Attributions', *Journal of Personality and Social Psychology*, 22, 171 - 188, (1972).

Meehan, J R - 'TALE-SPIN' (pp. 197-226) and 'Micro TALE-SPIN' (pp. 227-258), *Inside Computer Understanding: Five Programs plus Miniatures*, Lawrence Erlbaum

Associates, (1981).

Meehan, J R - 'TALE-SPIN: an Interactive Program that Writes Stories', *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, Morgan Kaufman, (1977).

Mellish, C S - '*Computer Interpretation of Natural Language Descriptions*', Ellis Horwood (1985).

Mueller, E T and Dyer, M G - 'Daydreaming in Humans and Computers', *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, 278 - 280, (1985).

Michaelson-Kanfer, A - 'Some Comments on Some Aspects of Comparative Narratives', *The Journal of Mathematical Sociology*, 18 (2-3), 215 - 216, Ed. Patrick Doriean, Gordon & Breach (1993).

Noble, H M and Hugh, M - '*Natural Language Processing*', Blackwell Scientific (1988).

Norvig, P - 'Inference in Text Understanding', *AAAI-87*, 561 - 565, (1987).

Pereira, F and Warren, D - 'Definite clause grammar for language analysis', *Artificial Intelligence*, 13 (3), 231 - 278, (1980).

Pomerantz, A - 'Attributions of Responsibility : Blamings', *Sociology*, 12, 115 - 121, (1978).

Popping, R and Roberts, C W - '*Content and Text Analysis*', SSIT 94 workshop paper, Amsterdam (December 1994).

Potter, J and Edwards, D - 'Nigel Lawson's Tent : Discourse Analysis and the Social Psychology of Fact', *European Journal of Social Psychology*, 20 (5), 405 - 424, (1990).

Quillian, M R - 'Semantic Memory', *Semantic Information Processing*, Ed. Marvin Minsky, MIT Press (1968).

Quillian, M R - 'The Teachable Language Comprehender : A Simulation Program and Theory of Language', *Communications of the ACM*, 12 (8), 459 - 476, (August 1969).

Ram, A - '{AQUA} : Asking Questions and Understanding Answers', *AAAI-87*, 313 - 316 (1987).

Raphael, B - 'SIR - Semantic Information Retrieval', *Semantic Information Processing*, Ed. Marvin Minsky, MIT Press (1968).

Rau, L F - 'Information Retrieval from Never-Ending Stories', *Proceedings of the 6th National Conference on Artificial Intelligence (AAAI 87)*, 317 - 321 (1987).

Reeves, J F - 'Ethical Understanding : Recognising and Using Belief Conflict in Narrative Processing', *Proceedings of the 7th National Conference on Artificial Intelligence (AAAI 88)*, 227 - 232 (1988).

Reiser, B J - 'Character Tracking and the Understanding of Narratives', *Proceedings of the 7th International Joint Conference on Artificial Intelligence* (1981).

Rumelhart, D E - 'Notes on a Schema for Stories', *Representation and Understanding: Studies in Cognitive Science*, 211 - 236, Ed. D.G Bobrow & A. Collins, Academic Press, (1975).

Sabah, G - 'Knowledge Representation and Natural Language Understanding', *AICOM*, 6 (3- 4), 155 - 186, (1993).

Schank, R - '*Explanation Patterns; Understanding Mechanically and Creatively*', Lawrence Erlbaum Associates (1986).

Schank, R and Riesbeck, C - '*Inside Case-Based Reasoning*', Lawrence Erlbaum Associates, (1989).

Schank, R and Riesbeck, C - '*Inside Computer Understanding*', Lawrence Erlbaum Associates, (1981).

Schank, R C - '*Dynamic Memory : a Theory of Reminding and Learning in Computers and People*', Cambridge University Press, (1982).

Schiffrin, D - '*Approaches to Discourse*', Blackwell, (1994).

Schild, U J and Kerner, Y - 'Multiple Explanation Patterns', *Proceedings of the European Workshop on Case-Based Reasoning '93*, Ed. Gook, 93 - 163, Springer Verlag, (1994).

Schildt, H - '*Turbo C/C++ the Complete Reference*', McGraw-Hill (1990).

Schmid, L A - 'Parsing word graphs using a linguistic grammar and a statistical language model', *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2, 41 - 44, (1994).

Schuster, B, Fursterlung, F and Weiner, B - 'Perceiving the Causes of Success and Failure: A Cross-Cultural Examination of Attributional Concepts', *Attribution : Perceiving the Causes of Failure*, Ed. E. E. Jones, (1989).

Seely Brown, J and Burton, R R - 'Multiple Representations of Knowledge for Tutorial Reasoning', *Representation and Understanding: Studies in Cognitive Science*, 311 - 349, Ed. D.G Bobrow & A. Collins, Academic Press, (1975).

Semin, G and Fiedler, K - 'Relocating Attributional Phenomena with a Language-Cognition Interface : The Case of Actors' and Observers' Perspectives', *European Journal of Social Psychology*, 19 (6), 491 - 508, (1989).

Sexton, C - '*C++ pocket book*', Butterworth-Heinemann (1993).

Shweder, R A - 'Likeness and Likelihood in Everyday Thought : Magical Thinking and Everyday Judgements about Personality', *Thinking : Readings in Cognitive Science*, 446 - 467 ed. P.N Johnson-Laird and P.C Wason, Cambridge University Press, (1978).

Silverman, B G - 'Survey of Expert Critiquing systems: Practical and Theoretical Frontiers', *Communications of the ACM*, 35 (4), 106 - 127, (1992).

Silverman, D - '*Interpreting qualitative data, methods for analysing talk, text and interaction*', Sage (1993).

Simmons, R F and Correira, A - 'Rule Forms for Verse, Sentences and Story Trees', *Associative Networks: The Representation and Use of Knowledge by Computers*', 363 - 392, Ed. N.V. Findler, Academic Press, (1979).

Skvoretz, J - 'Generating Narratives from Simple Action Structures', *The Journal of Mathematical Sociology*', 18 (2-3), 135 - 140, Ed. Patrick Doriean Gordon & Breach (1993).

Smeaton, A F - 'Progress in the Application of Natural Language Processing to Information Retrieval Systems', *The Computer Journal*, 35 (3), 268 - 277, Ed. P. Hammersley, Oxford University Press (1992).

Somerville, I - '*Software Engineering*', Addison-Wesley (1992).

Song, F and Cohen, R - 'The Interpretation of Temporal Relations in Narrative', *Proceedings of the 7th National Conference on Artificial Intelligence (AAAI 88)*, 745 -

750 (1988).

Suchman, L - '*Plans and Situated Actions: The Problem of Human - Machine Interaction*', Cambridge University Press, (1987).

Tennant, H - '*Natural Language Processing, an introduction to an emerging technology*', PBI (1981).

Thorndyke, P W - 'Cognitive Structures in Comprehension and Memory of Narrative Discourse', *Cognitive Psychology*, 9, 77 - 110, (1977).

Thorne, J P, Bratley, P and Dewar, H - 'The syntactic analysis of English by machine', *Machine Intelligence 3*, Ed. Donald Michie, American Elsevier 281 - 309 (1968).

Turnbull, W and Slugowski, B R - 'Conversational and Linguistic Processes in Causal Attribution', *Contemporary Science and Natural Explanation : Commonsense Conceptions of Causality*, Ed. D.J Hilton, 66 - 93, Harvester, (1988).

Van Dijk, T A - '*Handbook of Discourse Analysis*', Academic Press, (1985).

Vasconcellos M - 'Machine Translation', *Byte*, McGraw-Hill (January 1993).

Vilain, M - 'Getting serious about parsing plans', *Proceedings of the eighth National Conference on Artificial Intelligence*, 190 - 197, (1990).

White, L - Interviews concerning cooperatives in Latin America, unpublished.

Wilensky, R - '*Planning and Understanding, a computational approach to human reasoning*', Addison-Wesley (1983).

Willer, D - 'A Critique of Abell's 'Paths of Social Determination', *The Journal of Mathematical Sociology*', 18 (2-3), 191 - 202, Ed. Patrick Doriean Gordon & Breach (1993).

Winograd, T - '*Language as a Cognitive Process, volume 1 : Syntax*', Addison-Wesley (1983).

Winograd, T - '*Understanding Natural Language*', Edinburgh University Press (1972).

Wooffitt, R - '*Telling Tales of the Unexpected : The Organization of Factual Discourse*', Harvester Wheatsheaf, (1992).

Yazdani, M - 'Computational Storywriting', *Computers and Writing : Models and Tools*, 125 - 147, Ed. N. Williams and P. Holt, Blackwell, (1989).

Zhang, S - 'Story parsing grammar', *Journal of Computer Science and Technology*, 9 (3), 215 - 228, (1994).

Zhang, S - 'Weak precedence story parsing grammar', *Journal of Computer Science and Technology*, 10 (1), 53 - 64, (1995).

Newspaper stories were taken from four London-based broadsheet daily newspapers, the *Times*, the *Guardian*, the *Independent*, and the *Daily Telegraph*.