Automatic Item Generation via Frame Semantics:

Natural Language Generation of Math Word Problems

Paul Deane

Educational Testing Service

Princeton, NJ 08541

Kathleen Sheehan

Educational Testing Service

Princeton, NJ 08541

1. The Role of Natural Language Processing in Automatic Item Generation

1.1. The Scope of Automatic Item Generation

There is an increasing interest in the use of Automatic Item Generation (AIG) in educational assessment, concomitant with the development of technologies which have brought delivery of test content by computers into the mainstream. Early work (e.g. Bejar 1986, 1993, 1996, 2002; Bejar & Yocom 1991, Hively, Patterson & Page 1968, Irvine, Dunn & Anderson 1990, Laduca et al. 1986, Meisner, Luecht & Reckase 1993) has led to a flowering of AIG research (see, e.g., Irvine & Kyllonen 2002 for an overview.)

Automatic item generation -- the practice of creating assessment items algorithmically -- can be motivated in part by a number of obvious practical advantages (cf. Bennett in press). AIG can speed or even partially automate the development of new items, it can provide similar items at the same level of difficulty, thus improving test security, and it can support adaptive testing by providing similar items that vary systematically in difficulty. Given the attractiveness of these goals, much recent research has focused upon evaluating the extent to which these promises can be met (cf. Bejar et al. 2002, Enright, Morley & Sheehan 1999, Enright & Sheehan 2002, Hombo & Dresher 2001, Wright 2001).

However, automatic item generation obviously and critically depends upon well-founded models of the cognitive abilities underlying performance. Where such models are lacking, generation algorithms can have heuristic usefulness only. Much of the recent progress in AIG has come in areas where there are well-founded cognitive theories to support the development of AIG algorithms, e.g., matrix completion (Embretson 1993, 1998, 1999, 2001) or analytical reasoning (Newstead et al. 2002). It is thus important to approach AIG in the light of theories of test construction which ground item design upon an evidentiary basis, such as Evidence Centered Design (ECD), cf. Mislevy et al. 2002.

1.2. Generation of Verbal Items: Templates and Natural Language Generation

It is not particularly surprising to note that applications of AIG have been concentrated in areas where the underlying cognitive domain is restricted in content and highly structured (e.g., various forms of abstract reasoning and mathematics.) Progress in automatic generation of verbal

item types has been much more limited, due to the openness of content and the considerable complexity of natural language. For passage-based verbal items, there is a strong preference for developing naturalistic items based upon actually published materials, and the most productive approaches have focused upon providing techniques to support test developers by supporting more efficient source selection and evaluation (cf. Sheehan, Deane and Kostin, in this forum.)

Where more constrained item types have required natural language generation, the treatment of verbal materials has been very straightforward and minimal, involving generation from verbal templates, cf. Dennis et al. 2002. The practical advantages of template-based generation are obvious: implementation is straightforward, development time is minimal, and item generation can be based directly on existing items. However, automatic item generation from simple templates of this sort has clear limitations, in both theory and practice, as it is based on pure string manipulation without making use of any linguistic knowledge. This paper will explore the usefulness of an alternative method: Natural Language Generation (NLG), i.e., the use of computational linguistics techniques to automate text generation, cf. Reiter & Dale 1997, Cahill & Reape 1999, Paiva 1998.

Natural language generation systems typically have the following structure (cf. Reiter 1995, Cahill et al. 1999):

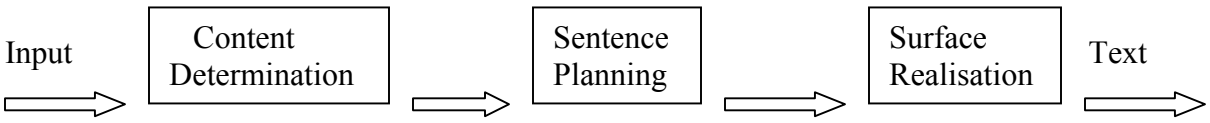

Figure 1

Reiter 1995 outlines several ways in which Natural Language Generation (NLG) systems have potential to achieve superior performance over simple template systems:

1. Maintainability.

Generating text from templates requires storing and manipulating large numbers of lists,

carefully adjusted to be appropriate for a specific generation task. A NLG system can be used to separate knowledge about language in general from knowledge about specific tasks and thus to increase the ease with which a system can be maintained and modified to suit changing needs and demands.

2. Flexibility of Output

As the number of templates in a template-based system grows, it becomes more and more probable that the variety of templates disguises the systematic combination of a much smaller set of variables. An NLG system can be used to reduce this variety, thus enabling much more flexible text generation.

3. Quality of Output

Sooner or later a template-based system comes up against one of the many context-dependencies inherent in human language, such as subject-verb agreement (*1 man sings, 2 men sing*), selection restrictions (you **drive** a **car**, but **fly** an **airplane**), definite vs. indefinite (*a student* … later, *the student*) and the like. A well-developed NLG system contains modules that handle such phenomena automatically. In a template based system, by contrast, it is necessary to handle such variations over and over again on an ad hoc basis.

4. Potential for Multilingual Output

In a template based system, text generation in multiple languages only happens at a price: namely, separate translation of every single template from the source into the target language. By contrast, in an NLG system low-level language-specific details are segregated in language-specific modules, making it possible to transpose high level knowledge without a complete system redesign.

It should be noted that the distinction between template-based and true natural language generation systems (also termed plan-based systems) is one of degree. I.e., there exist hybrid systems which use templates for high level document structure, but use NLG to automate the expression of details (cf. Pianta & Tovena 1999).

The considerations adduced thus far suggest that there is considerable room for synergy between automatic item generation within assessment theory, and natural language generation within computational linguistics. Both are concerned with automatic generation of verbal material, and in complementary ways. While automatic item generation focuses on controlling construct-relevant content, natural language generation focuses on lower-level document details: given a content domain, the strength of an NLG system is that it can provide general algorithm for expressing messages within that domain, eliminating massive maintenance of item-specific text generation facilities.

1.3. Automatic Item Generation, Natural Language Generation, and Evidence-Centered Design

Another way to address the potential contribution of NLG to automatic item generation is to consider the radical/incidental distinction discussed in Kyllonen 2002. Kyllonen points out that all AIG systems must distinguish between those factors which significantly affect item difficulty (***radicals***) and those which do not (***incidentals***). There is no way to know a priori which aspects of a verbal item are radicals, and which are incidentals: it depends on the cognitive content of the item, and on the actual task the item requires test subjects to perform. However, it is clear that many if not the vast majority of the variables referenced by an NLG system must be incidentals: particular choice of words, of grammatical constructions, of fine details of phrasing, should not affect difficulty significantly (though they might.) It is useful, however, to separate verbal tasks into two fundamentally different elements: those which only involve ***decoding*** (determining the meaning of the text as written) and those which involve ***content manipulation*** (performing inference or other thought processes on the content after decoding.) There is a direct relationship between a natural language generation system and decoding: an NLG system specifies **how to encode** what test subjects must **decode;** and what is more, if it is well-designed, it specifies how to do so without causing difficulties in comprehension. Unless a verbal item is explicitly testing decoding skills, therefore, radicals should involve content manipulation, and we can (as a first approximation) treat an NLG system as a way of abstracting away from incidentals only relevant to decoding.

Another way of putting the same point is that *wording typically interacts with radicals at the level of content*. This can take a variety of forms: interactions of content with task difficulty, correlations of content with task complexity, and verbal complexities which arise as the relationship between content and task becomes more and more indirect. For instance, Enright & Sheehan (2002) show that math word problems differ in difficulty depending on the real world content to which they are applied, e.g. when a particular mathematical formula is stated as a distance-rate-time problem, it is significantly more difficult than the same mathematical formula stated as a cost and price calculation. Similarly, when a mathematical formula is cast in terms of probabilities, it is significantly more difficult than when it is cast in terms of percents.

Consider the implications for a method of test construction which seeks to ground test development in psychological reality, such as Evidence Centered Design or ECD (as in Mislevy et al. 2002). From this perspective, these considerations suggest the following operating hypotheses:

1. Verbal content matters and must be included in a task analysis, even for a math problem.

2. The act of constructing an NLG system for automatic item generation requires that one build a model of verbal content and separate it from details of verbal encoding; thus, building an effective NLG system can contribute to the construction of a realistic task model, *because the content of the underlying task and the content of an item which expresses that task are necessarily linked*.

3. Most of the NLG system will concern details of verbal encoding, which can arguably be treated as incidentals unless the construct is designed to asses encoding/ decoding skills.

In short, natural language generation can contribute far more to Automatic Item Generation than the convenience of producing novel items on demand. The very act of building an NLG system requires a systematic analysis of task content, and such an analysis can be leveraged to facilitate construction of task models within an ECD framework.

1.4. Prospectus, Caveats, Plan of Attack

This paper is an exploration of the conceptual issues which have arisen in the course of building a natural language generation system for automatic item generation. While natural language processing techniques are applicable to general verbal items (see Sheehan, Deane & Kostin, this forum) math word problems are particularly tractable targets for natural language generation techniques (rather like the analytical reasoning items discussed by Newstead et al. 2002). In particular: They are drawn from a constrained universe of discourse; the underlying task is susceptible to formalization, little complex inference is required, and encoding/decoding (basic verbal comprehension) is not at issue. Yet math word problems still involve a variety of formal contents, a broad set of real world contexts, and significant verbal complexity.

In what follows, a number of tools from linguistics and natural language processing will be introduced which help to reduce this complexity to a manageable range. Most critical to this enterprise is the notion of *semantic frame* which derives ultimately from Fillmore (1968)'s concept of semantic case, since it provides a method for effectively organizing verbal content within an NLG system. In particular, the analysis will show how use of one such semantic frame -- the Transportation Frame -- makes it possible to analyze the variation of distance/rate/time word problems and isolate a serious of variables with clear Task-Model relevance.

2. Automatic Generation of Math Word Problems

2.1. Cognitive Analysis of Mathematical Problem Solving

Considerable progress has been made in understanding the cognitive structure of word problems in mathematics (cf. McArthur, Stasz & Hotta 1986-1987, Nathan, Kintsch & Young 1992, Paige & Simon 1966, Sebrechts, Enright, Bennett & Martion 1996, Singley, Anderson & Gevins 1991, Sweller, Mawer & Ward 1983). Much of this structure can be isolated by considering the formal structure of the math and the tasks which must be carried out in order to apply an (instructionally) appropriate algorithm. Such task variable analysis (cf. Brown & Burton 1978, Goldin & McClintock 1984; Riley, Greeno & Heller 1983) can be very effective at identifying specific factors which affect item difficulty.

However, word problems present two additional complications: verbal comprehension, and translation from verbal materials to formal mathematical representations. While mathematics word problems are not designed to test verbal comprehension, verbal content appears to have a strong impact on learning and generalization. Classifications of word problems (e.g. Hinsley, Hayes & Simon 1977, Mayer 1981, Hall et al. 1989, and Reed et al. 1990) typically pay close attention to the verbal scenario, or "story line". Mathematically unsophisticated learners appear to pay more attention to verbal content than to mathematical structure (Silver 1979; Reed, 1987; Reed et al. 1990; Weaver & Kintsch, 1992), and it is the combination of the story line with its translation into a mathematical formula which appears to drive learning. Instructional frequency of the story line/formula combination is strongly associated with item difficulty (Mayer, 1982) , leading to the theory that students learn by acquiring particular ***schemas*** -- pairings of verbal content with mathematical formulas, along the lines sketched by Marshall (1995).

Schema theory suggests that excessive particularity is the greatest instructional danger with respect to word problems: if the story line/mathematical formula pairings are too concrete, students will fail to generalize word problems effectively. Effective instruction requires that schemas be formed at a sufficiently abstract level to support accurate generalization to new problem types (Weaver & Kintsch, 1992).

The literature suggests a level of granularity in which verbal content establishes broad classes of problems, subdivided into specific schemas by the details of the story line/mathematical formula mapping. For instance, Mayer (1981) identifies 25 general families of algebra word problems, based upon verbal content: motion, current, age, coin, work, part, dry mixture, wet mixture, percent, ratio, unit cost, markup/discount/profit, interest, direct variation, inverse variation, digit, rectangle, circle, triangle, series, consecutive integer, physics, probability, arithmetic, and word. Within the set of motion problems, Mayer identifies 13 different templates depending on details of verbal content which affect the mapping onto mathematical formulas: e.g., whether one vehicle overtakes another, whether two vehicles converge on the same point, whether speed changes during a trip, whether one vehicle undertakes a round trip, and so forth.

The striking thing to note about such classifications is how extensively they are driven by the structure of verbal content, and not by the mathematical skills they are designed to test.

2.2. Schemas and Templates: Automatic Item Generation for Math Word Problems

Schema theory has provided the basis for automatic item generation of math word problems. Probably the most ambitious such effort is the Math Test Creation Assistant (Math TCA) project reported in a number of recent publications, including Singley & Bennett 2002. This system is designed to provide a general tool for constructing items in line with the principles of schema theory, supporting construction of families of models and allowing the analyst to assign specific mappings from words to equations. The Math TCA can be viewed as a general-purpose automatic item generation system for math, and as a simple template-based natural language generation system. Its -- very considerable -- strengths lie in its complete customizability and in the high degree of control it offers over the mathematical structure of models. But viewed as a natural language generation system, or as a theory of the verbal content used in math word problems, its content is minimal.

Beyond fixed verbal templates

The essence of a template-based system lies in the postulation of a series of fixed verbal template. For example, a very simple distance-rate-time model could be built around the following verbal template:

> A ____ traveled ___ miles in ___ hours.
> On average, how fast did the ____ move during this time period?

But there are many wordings compatible with this choice of variables. We could easily substitute another template:

> It took ___ hours for a ____ to go ____ miles.
> What was the _____'s average speed?

There is a broad range of wordings available, subject to the principles of verbal paraphrase. In a template based system, paraphrase possibilities have to be enumerated exhaustively: not a problem when dealing with a single template, but highly problematic when managing a library containing thousands of templates. On the other hand, one of the potential

advantages of a natural language generation system is that paraphrase possibilities can be derived by choosing different grammatical and lexical options during generation.

Moreover, the paraphrase possibilities for any given model family are not random: they reflect the resources the language makes available for that particular class of verbal content. If the same verbal content is reused in a different testing context, the same resources should be available for reuse in the new context.

Beyond ad-hoc word lists

In a verbal template system, slots are filled by choosing items from lists. This provides great power to fine-tune the generative power of the system, but the content of the lists is not fixed by any principle other than authorial judgment.

Another way to make the point is that the blanks in a verbal template correspond to **concepts** not to randomly chosen words. In other words, the simple template for motion problems discussed earlier would better be expressed as

A VEHICLE traveled INTEGER miles in INTEGER hours.

Or, even better, as

VEHICLE MOVE(DISTANCE,TIME)

If we know English, we know what words correspond to vehicles; we know what expressions indicate distance and time; we know what verbs and grammatical constructions are available to express motion. All of this linguistic detail is part of the linguistic **encoding** of verbal content. A general understanding of distance/rate/time word problems can only be ignored by abstracting from linguistic details and mapping directly to mathematical formulas from the verbal content **at this level of abstraction**. Indeed, if someone speaks Japanese or Spanish instead of English, their ability to solve math word problems will be essentially the same as long as they know how to map from general concepts like VEHICLE, MOTION, DISTANCE and TIME to the right mathematical knowledge.

One of the advantages of a natural language generation system, if it is well designed, is that the categories and relationships it contains will embody a theory of the expression of verbal content. In the case of math word problems, one of the generalizations we wish to capture is that the relationship between mathematics and verbal content must be represented at the level of generic concepts like VEHICLE, MOVE, DISTANCE, RATE, or TIME, and not at the level of individual words and fixed verbal templates. In fact, there is evidence that students at low skill levels solve math word problems by memorizing verbal and mathematical templates, and that their ability to transfer these skills to new problem types is limited by their failure to represent the problem at the right level of generality (Reed, Dempster, & Ettinger 1985).

Slot dependencies derive from constraints on verbal content

In a pure template system, the selection of slot fillers should be entirely independent: it should not matter what has been chosen to fill slot A when it comes time to fill slot B. In actual practice, there are important dependencies across slots. If a vehicle is moving 5,000 miles per hour, it is not a bicycle. If a vehicle is being driven, if is not an airplane. If its operator is called a pilot, it is not an automobile. Mechanisms can be provided to keep track of such dependencies, but if that is done item by item, it rather misses the point. Given ordinary knowledge of the world and the language we use to describe it, such dependencies follow as a matter of common sense. A template-based system has no way of capturing common sense knowledge. By contrast, some types of natural language generation systems allow for explicit modeling of domain-specific knowledge (cf. Beale, Nirenburg, Viegas & Wanner 1998). Or to make the point rather baldly: when verbal content is what one wishes to model, there is really no substitute for a model of verbal content.

Schema families are driven by the structure of verbal content

Finally, it is worth considering just what drives the multiplication of templates in a schema-driven model. Mayer (1981) postulates separates schemas for motion problems in which

-- one vehicle overtakes another
-- two vehicles travel in opposite directions
-- two vehicles converge on a single destination

-- one vehicle goes on a round trip

-- two vehicles take unrelated trips

and so forth.

Each of these problem types entails different mathematical relationships: but the math is not driving the variation in problem forms. It is the content of the domain. Given the basic logic of motion through space, and a choice of one or two vehicles to move, the set of motion scenarios is entirely predictable. What is less predictable (because it requires a deeper understanding of the math) is the relationship between scenarios and mathematical formulas.

To sum up the argument: verbal content is not an afterthought in math word problems. It has a fundamental role to play. The range of problem types is highly constrained by the structure of the content domain, which also largely determines their conceptual properties and the linguistic resources by which their underlying mathematical relationships can be expressed.

These considerations suggest the verbal content is the critical level for the analysis of math word problems, because it functions as the interface between ordinary language and mathematical thought. On the one hand, the mathematical properties of word problems are irreducibly tied to the abstract conceptual structure of the content domain. On the other hand, once verbal content has been selected, the rest of the automatic item generation process reduces to linguistic encoding: potentially complex, but essentially irrelevant to the task model.

2.3. Moving beyond Templates: Using Semantic Frames to Formalize the Concept of Word Problem Families

The problem, then, is to have a method for representing verbal content which can play a dual role, serving both as a representation of the generic conceptual structure that appears to be critical for math word problems, and which can provide the basis for natural language generation. An excellent candidate for this purpose can be found in **Frame Semantics**, a linguistic theory of word meaning which appears to have exactly the needed properties.

Frame semantics is due largely to the work of Charles Fillmore (Fillmore 1968, 1975, 1976, 1977a,b, 1982, 1985; Fillmore & Atkins 1994; Baker, Fillmore & Lowe 1998; Fillmore & Baker 2001). It belongs to the same general family of theoretical constructs as Minsky's concept

of *frame* as "a data structure representing a stereotyped situation" (1975:212) and Schank and Abelson's (1977) concept of *script*. It differs from these in its emphasis on using frames as tools to characterize word meaning, using them to capture not only stereotypical expectations about a scene, but to characterize the semantic relationships among words which reference common underlying knowledge.

Frame semantics derives primarily, however, from Fillmore's work on the concept of *semantic case frames*. Fillmore (1968) observed that the relationships between a verb and the nouns associated with it fall into a small set of basic patterns. For instance, verbs of physical action typically involve patterns like the following:

John broke the plate with a hammer.
The hammer broke the plate.
The plate broke.

The patterns we see here are typical for verbs in which an AGENT (typically a human being) uses an INSTRUMENT (a tool or other item) to produce a change in a PATIENT.

Fillmore identified a small set of these basic case frames, corresponding to high-level abstract concepts such as MOVE, CHANGE, and PERCEIVE. E.g., an AGENT CHANGES a PATIENT with an INSTRUMENT, a THEME MOVES from ORIGIN to DESTINATION along a ROUTE, an EXPERIENCER PERCEIVES a PERCEPTION, and so on.

Case frames represent only the most skeletal aspects of meaning. In ensuing work, Fillmore examined how case frames are deployed in more specific lexical items, and developed a theory of meaning in which abstract case frames are instantiated by more specific semantic frames, and argued that the meanings of words can only be stated properly if the semantic frames are taken as basic and the words are defined by the role they play in particular semantic frames. For instance, the abstract semantic cases THEME, SOURCE and GOAL correspond in one instance to the concepts of (commercial) GOODS, the BUYER, and the SELLER. In terms of Fillmore's theory, there is a *Commercial Transaction* frame. This frame includes a list of **participants**, most importantly, the buyer, the seller, the goods, and money. It also includes a set of **actions** and **events**, most importantly, the transfer of goods from the buyer to the seller in exchange for money. A wide range of terms can then be defined against this frame. For instance,

the verbs *buy*, *sell*, *spend*, and *cost* all evoke the same background concepts, but use them differently. *Buy* focuses our attention on the buyer's getting the goods. *Sell* focuses our attention on the seller's giving up the goods. *Spend* focuses our attention on the buyer's giving up the money. *Cost* focuses our attention on the price, i.e., the relationship between the goods and the money. In each case, the conceptual base is the same; what differs how each word selects particular parts of the frame for emphasis or elaboration.

In Fillmore's conception, semantic frames form a schematic hierarchy. Thus, beneath the relatively abstract Commercial Transaction frame can be found such frames as the frame for lending money, where the basic concepts of buyer, seller, goods, and money are elaborated into the more specific concepts of borrower,lender, loan, and interest. Critically, Frame Semantics allows both a specification of the conceptual relationships among concepts and a detailed linguistic specification of how those concepts are expressed verbally.

There is an ongoing NSF-funded research project, the FrameNet project, which has as its goal the construction of a comprehensive database of semantic frames and the classification of a broad segment of English vocabulary in terms of Frame Semantics (Fillmore & Atkins, 1998 Johnson & Fillmore 2000, Ruppenhoffer, Baker & Fillmore 2002). As it increases in coverage, it will provide detailed syntactic and semantic information about words stated in terms for semantic frame theory. Frame-based semantics has already been used effectively for construction of semantic parsers in computational linguistics (cf. Gildea & Jurafsky 2002). But one of the striking features of frame semantic analysis is that it brings together all of the vocabulary that is relevant to talking about a particular topic and arranges it into a structure where the relationships among the elements are made explicit.

2.4. Frame Semantics and the Structure of Word Problems

The relevance of frame semantics becomes clear as soon as we examine patterns of vocabulary use in math word problems. The vocabulary and syntax are constrained, highly systematic, and essentially stereotyped, and appear to have been purposely selected for high prototypicality. In essence, each type of word problem appears to make systematic use of a specific semantic frame.

That is, the vocabulary and syntax used are typically drawn from core areas of human experience: e.g., motion and transportation, building and creating, buying and selling, or from important instances of these core concepts, such as interest and taxation. For any given semantic frame

- there are a limited number of roles and relationships, with regular rules about how these roles and relationships map onto sentences
- there is a significant amount of interdependency and specialized lexical knowledge (motorists drive cars, pilots fly airplanes; investors earn interest on accounts, governments collect tax from taxpayers)

These linguistic details matter for natural language generation, but they can be ignored for other purposes. From a frame semantic point of view, if we do not care about the linguistic details, a wide range of sentences can be reduced to simple semantic formulas, such as PERSON OPERATES VEHICLE, RECIPIENT GETS MONEY, and the like. If we abstract word problems in this fashion, the resulting templates appear to be commensurate with the schemas postulated in schema theory. That is, the variables we get out of the verbal template and the variables we need to include in a task model can be matched essentially one to one.

2.4.1. A Sample Domain: Distance-Rate-Time Problems

Distance-rate-time problems provide an excellent illustration of the principles involved. They deploy some of the most common and central linguistic concepts: verbs of motion, expressions for speed, time and distance, and related terms. The actual mathematical content, expressed in terms of measurements of distance, time, and speed, is overlaid by a rich conceptual structure in which people or vehicles follow paths from one place to another. The verbal expression of distance rate time problems can therefore be reduced to a set of choices within this conceptual frame -- e.g., the number of distinct travel events, the type of participants in these events (what sort of person or vehicle is involved), and the relationship among these events (for instance, round trip or multiple-leg trips) -- combined with specific choices for the concepts used for the measurement of distance, rate and time. These choices are of the sort that can be incorporated readily into a task model; moreover, once these choices have been made, the task of

generating the actual language is highly constrained, allowing construction of a natural language generation system that maps these high-level conceptual choices directly to actual word problems.

Let us examine some of the details of this system, starting with an examination of the Transportation frame commonly deployed in Distance-Rate-Time word problems, and proceeding through enough of the linguistic detail to see semantic frames provide an excellent representational mode for the schematic structure of math word problems of this type.

Semantic Case: Motion

At the most abstract level the vocabulary being deployed is that of *motion* -- and so the highest level schema involves --

The **Theme** -- the entity which moves or is moved;
The **Source** -- the location from which the theme moves;
The **Goal** - the location to which the theme moves;
The **Route** -- the path from source to goal
The **Distance** -- the length of the path from source to goal

This basic frame must be elaborated in two ways. First, we must take into account the relationship between motion and time; i.e.,

The **Start time** -- the time at which the Theme leaves the Source;
The **End time** -- the time at which the Theme reaches the Goal;
The **Duration**, or overall time -- the time it takes the theme to reach the goal;
The **Motion Event** -- the motion which occurs between the start and end time
The **Rate**, or speed -- the relationship between distance and duration for a motion event

Second, we must allow for actions in which an agent causes movement to take place, in which case we have to allow for the following additional roles:

The **Agent** -- typically a person, the entity which causes motion to take place
The **Instrument --** the means by which motion is effected

The basic motion vocabulary involves different relationships among these roles; in particular, the choice of motion verbs depends upon which combination of roles is expressed in a sentence.

Semantic Frames: Transportation

Given the basic motion case frame, a broad range of more specific concepts can be defined. Among these are the concepts associated with transportation by vehicle. The Transportation frame can be described as a specific instance of the generic motion frame in which --

(i) The Agent is also the Theme (i.e., people cause themselves to move)
(ii) The Instrument is a vehicle (i.e., a moving object which carries the Agent and possibly other people)

The transportation frame inherits all of the participant roles associated with the Motion frame, but adds additional or more specialized roles. In particular, we must distinguish

Operator -- (Agent and Theme) -- the person who controls the vehicle
Passenger -- (Theme) -- other people carried by the vehicle
Vehicle -- (Instrument) -- the object used to transport the operator and any passengers
Fuel -- (Means) -- a substance which powers the vehicle

Beneath this generic Transportation frame we must specify a hierarchy of more specific frames which define the detailed vocabulary of motion. We must distinguish types of vehicles: land vehicles, typically driven on a road, air vehicles, typically flown through the air, sea vehicles, typically sailed on a body of water. We must distinguish subframes even more specific, in order to account for such details as the fact that **chauffeurs** drive **limousines**, or that we

steam a **steamship**. Each specialized term inherits grammatical and semantic properties from the higher-level frames, and derives specific meaning from the relationship it holds to other words with the same semantic base.

From Frames to Schematic Logical Representations

Given a frame analysis, one can severely reduce the information necessary to represent a word problem, as the choice of particular words is almost entirely driven by the choice of subframe. In the work being reported on here, a variant of second-order predicate logic is used to represent word problems at this level of abstraction, which is (approximately) that of the general Transportation frame. For instance, a sentence like *a car drove 600 miles* could be represented logically as:

VEHICLE|W TRAVEL(DISTANCE="600"|P)

where lexical elements (**VEHICLE**, **TRAVEL**, **DISTANCE**) are represented as abstract labels, to be filled in by choosing a frame and substituting appropriate lexical and grammatical materials (which are essentially predictable given the choice of content expressed by the logic.)

From Logical Representations to Mental Models

Of course, real distance-rate-time word problems involve more than one sentence, and more than one instance of vehicles traveling from one point to another. Word problems can describe sequences of events, events happening to multiple people, events in a variety of times and places, subject to two main constraints. First, the verbal content must fit the structure of the mathematical content being tested, which means (for example) that distance-rate-time problems are most sensibly used to test people's ability to solve simple three-variable algebra problems; second, that the verbal content should not be unnecessarily lengthy, abstruse, or complex. Given these constraints, the variations observed among items fall within clear limits. A word problem might mention two people's names: it is very unlikely to mention five or six. A word problem might describe a sequence of two or three distinct events: it is most unlikely to mention a

sequence of five or six. A word problem might compare two different types of transportation: it is most unlikely to compare transportation with some utterly unrelated class of event.

As a result, word problems have a complexity and structure which is well suited to the construction of mental models in the sense of Johnson-Laird (1983) and Gentner & Stevens (1983): a small mental structure with a limited number of component entities and relationships, subject to direct imagination and inspection. The frame-based conceptual structures which underlie the logic of distance-rate-time problems can easily be deployed to create mental models in this sense, and analysis of a set of GRE distance-rate-time items suggests that such problems vary along the following dimensions:

I. Mental Model Structure

- **Ia** The number of events
- **Ib** The semantic frame associated with each event
- **Ic** The primary participants in each event
- **Id** The identity of primary participants across events
- **Ie** The secondary roles that are relevant to each event
- **If** The identity of secondary roles across events

II. Task-Relevant Problem Structuring

- **IIa** The mapping from mathematical variables to frame-semantic roles
- **IIb** Which variables are explicitly given values in the text
- **IIc** Which variables are queried, so that their values must be determined to answer the question
- **IId** Which other variables are left implicit so that their values can only be determined by inference or calculation
- **IIe** Additional operations required to complete the task
- **IIf** Modifications or limitations on variables to guarantee correct solutions

III. Document Format and Structure

- **IIIa** Document type
- **IIIb** Use of metalanguage summarizing the relationship among events

IIc Arrangement of content, e.g., the choice and placement of sentences & phrases directly expressing each event

IV Variations in Language

IVa Alternative sentence structures

IVb Alternative methods of identifying referents

IVc Other variations in detailed phrasing

This framework makes it possible to analyze variations among word problems of the same type by reference to the underlying structure of the conceptual domain. Different word problems, rather than being conceptualized as arbitrarily different, can be viewed as varying systematically along specific dimensions; differences among word problem types, on the other hand, can be viewed as involving similar dimensions but different underlying frames. And this precise specification of the content is what is necessary to support true natural language generation (NLG) systems.

3. Natural Language Generation of Math Word Problems

The discussion thus far has been concerned, in essence, with the feasibility of natural language generation of math word problems. The analytical methods outlined in this paper have been used as the basis for the construction of a prototype NLG system, Model Creator, which supports the automatic item generation of distance-rate-time problems. The system is currently being extended to cover interest and taxation problems, using an architecture which exploits the analytical methods illustrated in this paper, and will soon be extended to cover a range of other math word problem types.

The prototype system was implemented using VisualText, an integrated development environment (IDE) for natural language processing. The major advantage of using VisualText for the NLG task was the fact that the underlying programming language, NLP++, supports close programmatic integration of parse trees with the rest of the system. Major components were implemented as separate VisualText modules in a pipeline architecture implemented as an MFC (Microsoft Foundation Classes) GUI (graphical user interface).

The essence of the system is that it provides a user interface in which users need only control the high level parameters -- the structure of the model and the task-relevant manipulations of the problem structure. I.e., it provides control over parameters at a level compatible with building a task model. The process of fleshing out those choices in verbal form is entirely under the control of the NLG component.

The prototype system as it stands does not directly control for difficulty, though there is reason to believe that models which use the same choices and parameters will prove to perform at roughly comparable levels of performance. However, a system of this kind makes it possible to classify any given instance of a particular problem type in terms of the variables that underlie the system. It is anticipated that future work will address the extent to which these variable affect item difficulty.

Also, the NLG prototype discussed here does not yet address one of the potentially most interesting uses of natural language generation. This is the idea that **the same content can be reused in a variety of problem types**.

In principle, the range of basic sentence patterns and prototypical real world situations is fairly limited: for instance, at the level of basic semantic case frames ("thematic roles"), no linguistic system postulates more than forty or fifty fundamental semantic relationships by which language structures its description of real world scenes. Though the elaboration into concrete examples multiplies the range of possibilities many orders of magnitude, grammatical rules for expressing content seldom go to that level of detail. Moreover, the abstract semantic structure seems to be pretty much universal, recurring from one language to the next.

In short, once a description of a reasonable range of (mathematically relevant) vocabulary has been completed, there is a realistic hope of reaching a new level of generalization in which the same linguistic content can be reused for different item types, different mathematical equations, or even in different languages

The real power of NLG for automatic item generation will come when language resources can be relinked and reused; but even in its current form, it offers consider power and flexibility for generating new items in a controlled manner.

In summary NLG offers the opportunity to develop AIG systems with very flexible capacity for generating realistic word problems, where it is possible rapidly to create a large set of items, whether on the fly or for formative assessment

References

Baker, C. F., Fillmore, C. J., & Lowe, J. B., (1998). *The Berkeley FrameNet Project.* In *Proceedings of the COLING-ACL*, Montreal, Canada. Association for Computational Linguistics.

Beale, S., Nirenburg, S., Viegas, E. & Wanner, L., (1998)."De-c*onstraining text generation*". In *Proceedings of the Ninth International Workshop on Natural Language Generation*, Niagara-on-the-Lake, Ontario.

Bejar, I. I. (1986). *Analysis and generation of hidden figure items: A cognitive approach to psychometric modeling.* Princton, NJ:ETS

Bejar, I. I. (1993) A generative approach to psychological and educational measurement. In N. Frederiksen, R.J. Mislevy & I.I. Bejar, (Ed.), *Test theory for a new generation of test*s (pp. 323-359). Hillsdale, NJ: Lawrence Erlbaum Associates.

Bejar, I. I. (1996) *Generative response modeling: Levering the computer as a test delivery medium* (Research Report RR-96-13). Princeton, NJ: Educational Testing Service.

Bejar, I. I. (2002) Generative testing: from conception to implementation. In S.H. Irvine & P. Kyllonen (Eds), *Item generation for test development* (pp. 199-218). Mahwah, NJ: Lawrence Earlbaum Associates.

Bejar, I. I., Lawless, R., Morley, M. E., Wagner, M. E., Bennett, & R. E., Revuelta, J. (2002). *A feasibility study of on-the-fly item generation in adaptive testing*. (GRE Board Professional Report No. 98-12P; ETS Research Report 02-23) Princeton, NJ: Educational Testing Service.

Bejar, I. I. & Yocom, P. (1991). A generative approach to the modeling of isomorphic hidden-figure items. *Applied Psychological Measurement, 15*(2), 129-137.

Bennett, R. E. (In press) *Automatic item generation: An overview*. Princeton, NJ: Educational Testing Service.

Brown, J. S. & Burton, R. R. (1978) Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science, 2*, 155-192.

Cahill, L., Doran, C., Evans, R., Mellish, C., Paiva, D., Reape, M., Scott, D., & Tipper, N. (1999). In search of a reference architecture for NLG systems. In *Proceedings of the 7th European Workshop on Natural Language Generation (EWNLG '99)*, pp. 77-85. KIT Report 97, Toulouse, France.

Cahill, L. & Reape, M. (1999). *Component tasks in applied NLG systems*. (Information Technology Research Institute Technical Report No. ITRI-99-05), University of Brighton.

Dennis, I., Handley, S., Bradon, P., Evans, J. & Newstead, S. (2002). Approaches to modeling item-generative tests. In S.H. Irvine & P. Kyllonen (Eds), *Item generation for test development* (pp. 53-72). Mahwah, NJ: Lawrence Earlbaum Associates.

Embretson, S. E. (1993). Psychometric models for learning and cognitive processes. In N. Frederiksen, R. J. Mislevy, & et-al (Eds.), *Test theory for a new generation of tests* (pp. 125-150) Hillsdale, NJ: Lawrence Erlbaum Associates.

Embretson, S. E. (1998). A cognitive design system approach to generating valid tests: Application to abstract reasoning. *Psychological Methods, 3*, 380-396.

Embretson, S. E. (1999). Generating items during testing: Psychometric issues and models. *Psychometrika, 64*(4), 407-433.

Embretson, S. E. (2002). Generating abstract reasoning items with cognitive theory. In S. H. Irvine & P. Kyllonen (Eds), *Item generation for test development* (pp. 219-250). Mahwah, NJ: Lawrence Earlbaum Associates.

Enright, M. K., Morley, M. & Sheehan, K. M. (1999). *Items by design: Theimpacdt of systematic feature variation on item statistical characteristics*. (GRE Research Report No. 99-15-R). Princeton, NJ: Educational Testing Service.

Enright, M. K., & Sheehan, K. (2002). Modeling the difficulty of quantitative reasoning items: Implications for item generation. In S. H. Irvine & P. Kyllonen (Eds), *Item generation for test development* (pp. 129-158). Mahwah, NJ: Lawrence Earlbaum Associates.

Fillmore, C. J. (1968): The case for case. In Bach and Harms (Ed*.): Universals in linguistic theory* (pp. 1-88), Holt, Rinehart, and Winston, New York

Fillmore, C. J. (1976): Frame semantics and the nature of language. In *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, Volume 280 (pp. 20-32).

Fillmore, C. J. (1977a): Scenes-and-frames semantics, linguistic structures processing; in Zampolli, Antonio (Ed.): *Fundamental studies in computer science*, No. 59, North Holland Publishing.

Fillmore, C. J. (1977b): The need for a frame semantics in linguistics; in Karlgren, Hans (Ed.): *Statistical Methods in Linguistics*

Fillmore, C. J. (1982): Frame semantics. In *Linguistics in the morning calm* (pp. 111-137), Hanshin Publishing Co., Seoul, South Korea.

Fillmore, C. J. (1985). Frames and the semantics of understanding. *Quaderni di Semantica*, 6(2), 222-254.

Fillmore, C. J. & B. T. S. Atkins (1994). Starting where the dictionaries stop: The challenge for computational lexicography. In Atkins, B. T. S. and A. Zampolli, eds. *Computational Approaches to the Lexicon*. Clarendon Press.

Fillmore, C. K., & Baker, C. F. (2001). Frame semantics for text understanding. In *Proceedings of WordNet and Other Lexical Resources Workshop*, NAACL, Pittsburgh, June 2001.

Gentner, D., & Stevens, A.L., Eds. (1983) *Mental models*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Gildea, D. & Jurafsky, D. (2002). Automatic labeling of semantic roles. *Linguistics 28*(3), 245-288.

Goldin, G. A. & McClintock, C. E. (Eds.) (1984). *Task variables in mathematical problem solving.* Philadelphia, PA: Franklin Institute Press.

Hall, R., Kibler, D., Wenger, E. & Truxaw, C. (1989) Exploring the episodic structure of algebra story problem solving. *Cognition and Instruction, 6*(3), 223-283.

Hinsley, D. A., Hayes, J. R., & Simon, H. A. (1977) From words to equations -- Meaning and representation in algebra word problems. In M. A. Just & P. A. Carpenter (Eds.), *Cognitive processes in comprehension* (pp. 89-96) Hillsdale, NJ: Lawrence Earlbaum Associates.

Hively, II, W., Patterson, J. L. & Page, S. H. (1968). A "universe-defined" system of arithmentic achievement tests. *Journal of Educational Measurement, 5*(4), 275-290.

Hombo, C. M. & Dresher, A. R. (2001). A simulation study of the impact of automatic item generation under NAEP-like data conditions. *Paper presented at the annual meeting of the National Council of Educational Measurement*, Seattle, WA. April 2001.

Irvine, S. H. & Kyllonen, P. (Eds.). (2002). *Item Generation for test development*. Mahwah, NJ: Lawrence Earlbaum Associates, Inc.

Irvine, S. H., Dunn, P. L. & Anderson, J. D. (1990). Towards a theory of algorithm-determined cognitive test construction. *British Journal of Psychology, 81*, 173-195.

Johnson-Laird, P. N. (1983). Mental models: Towards a cognitive science of language, inference, and consciousness. Cambridge: Cambridge University Press; Cambridge, MA: Harvard University Press

Johnson, C. R. & Charles J. Fillmore. (2000). The FrameNet tagset for frame-semantic and syntactic coding of predicate-argument structure. In *Proceedings of the 1st Meeting of the North American*

Chapter of the Association for Computational Linguistics (ANLP-NAACL 2000), April 29-May 4, 2000, Seattle WA, pp. 56-62

Kyllonen, P. (2002). Item generation for repeated testing of human performance. In S. Irvine & P. Kyllonen (Eds.), *Item generation for test development*. (pp. 251-276). Mahwah, NJ: Lawrence Earlbaum Associates.

LaDuca, A., Staples, W. I., Templeton, B. & Holzman, G. B. (1986). Item modeling procedure for constructing content-equivalent multiple choice questions. *Medical-Education, 20*(1), 53-56.

Marshall, S. (1995). *Schemas in problem solving*. New York: Cambridge University Press.

Mayer, R. E. (1981). Frequency norms and structural analysis of algebra story problems into families, categories, and templates. *Instructional Science, 10*, 135-175.

Mayer, R. E. (1982) Memory for algebra story problems. *Journal of Educational Psychology, 74*(2), 199-216.

McArthur, D., Stasz, C. & Hotta, J. Y. (1986-1987) Learning problem-solving abilities in algebra. *Journal of Educational Testing Systems, 15*, 303-324.

Meisner, R. M, Luecht, R. & Reckase, M. D. (1993). *The comparability of the statistical characteristics of test items generated by computer algorithms.* (ACDT Research Report Series No. 93-9) Iowa City,IA: The American College Testing Program.

Minsky, M. (1975). A framework for representing knowledge. In *The Psychology of Computer Vision*, (pp. 211-277). New York: McGraw-Hill

Mislevy, R. J., Steinberg, L. S., and Almond, R. G. (2002). On the roles of task model variables in assessment design. in S. Irvine & P. Kyllonen (Eds.), *Item generation for test development* (pp. 97-128). Mahwah, NJ: Lawrence Earlbaum Associates.

Nathan, M. J., Kintsch, W. & Young, E. (1992) A theory of algebra-word-problem comprehension and its implications for the design of learning environments. *Cognition and Instruction, 9*(4), 329-389.

Newstead, S., Bradon, P., Handley, S., Evans, J. & Dennis, I. (2002). Using the psychology of reasoning to predict the difficulty of analytical reasoning items. In S. Irvine & P. Kyllonen (Eds.), *Item generation for test development*. Mahwah, NJ: Lawrence Earlbaum Associates.

Paige, J. M. & Simon, H. A. (1966) Cognitive processes in solving algebra word problems. In B. Kleinmutz, (Ed.), *Problem solving: Research, method and theory* (pp. 51-119) New York: Wiley.

Paiva, D. S. 1998. *A Survey of Applied Natural Language Generation Systems.* (Information Technology Research Institute Technical Report Series ITRI-98-03), University of Brighton

Pianta, E. & Tovena, L. M. (1999). Mixing representation levels: The hybrid approach to automatic text generation. In *Proceedings of AISB'99,* pp. 8-13.

Reed, S. K. (1987) A structure-mapping model for word problems. *Journal of Experimental Psychology: Learning, Memory and Cognition, 13*(1), 124-139.

Reed, S. K., Ackinclose, C. C. & Voss, A. A. (1990) Selecting analogous problems: Similarity versus inclusiveness. *Memory and Cognition 18,* 83-98.

Reed, S. K., Dempster, A. & Ettinger, M. (1985). Usefulness of analogous solutions for solving algebra word problems. *Journal of Experimental Psychology, Learning, Memory, and Cognition, 11*(1), 106-125.

Reiter, E. (1995) NLG vs. Templates. In *Proceedings of the Fifth European Workshop on Natural Language Generation*. Leiden, The Netherlands.

Reiter, E. & Dale, R. (1997) Building applied natural language generation systems. *Natural Language Engineering 3*(1):57-87.

Riley, M. S., Greeno, J. G. & Heller, J. I. (1983) Development of children's problem-solving ability in arithmetic. In H.P. Ginsburg (Ed.), *The development of mathematical thinking* (pp. 153-196) New York: Academic Press.

Ruppenhofer, J., Baker C. F. & Fillmore, C.J. (2002): Collocational Information in the FrameNet Database. In Braasch, Anna and Claus Povlsen (eds.), *Proceedings of the Tenth Euralex International Congress*. Copenhagen, Denmark. Vol. I: 359-369

Schank, R. C., Ableson, R. P. (1977). *Scripts, plans, goals and understanding: An enquiry into human knowledge structures*. Hillsdale, N.J.: Lawrence Earlbaum Associates.

Sebrechts, M. M., Enright, M., Bennett, R. E. & Martin, K. (1996). Using algebra word problems to assess quantitative ability: Attributes, strategies, and errors. *Cognition and Instruction 14*(3), 285-343.

Sheehan, K. M., Deane, P. & Kostin, I. (2003). A partially automated system for generating passage-based multiple-choice verbal reasoning items. *Paper presented at the National Council on Measurement in Education Annual Meeting*, Chicago, IL, April 2003.

Silver, E. A. (1979) Student perceptions of relatedness among mathematical verbal problems. *Journal for Research in Mathematics Education*, *10*, 195-210.

Singley, M. K., Anderson,J. R. & Gevins, J. S. (1991). Promoting abstract strategies in algebra word problem solving. In Y*he International Conference on the Learning Sciences: Proceedings of the 1991 Conference*. Charlottesville, VA: Association for the Advancement of Computing in Education.

Singley, M. K. & Bennett, R. E. (2002). Item generation and beyond: Applications of schema theory to mathematics assessment. In S. Irvine & P. Kyllonen (Eds.), *Item generation for test development*. Mahwah, NJ: Lawrence Earlbaum Associates.

Sweller, J., Mawer, R. & Ward, M. (1983). Development of expertise in mathematical problem solving. *Journal of Experimental Psychology: General, 112*, 639-661.

Weaver, C. A., & Kintsch, W. (1992) Enhancing students' comprehension of the conceptual structure of algebra word problems. *Journal of Educational Psychology, 84*(4), 419-428.

Wright, D. (2002). Scoring tests when items have been generated. In S. H. Irvine & P. Kyllonen (Eds), Item generation for test development. Mahwah, NJ: Lawrence Earlbaum Associates.