Constructing Controlled English for Both Human Usage and Machine Processing

Ping Xue[1], Steve Poteet[1], Anne Kao[1], David Mott[2], Dave Braines[2]

[1]Research & Technology, The Boeing Company, USA
{ping.xue, stephen.r.poteet,anne.kao}@boeing.com
[2]Hursley Emerging Technology Services, IBM, UK
{mottd, dave_braines}@uk.ibm.com

Abstract. We present our on-going research on constructing and extending a version of Controlled English (CE) in support of knowledge sharing and decision-making for effective and efficient operations in the military coalition environment. This work would be useful for any multinational English speaking environment. This CE is intended for both human use and machine processing, providing:

(i) A user-friendly language in a form of English enabling the user to use it in a fairly intuitive way.
(ii) A precise language that enables clear, unambiguous representation of information that is amenable to rule-based interpretation and inferencing.

The paper focuses on the discussion of methods for CE construction while optimizing a balance between the naturalness for humans and machine readability of the CE language in light of theoretical considerations and empirical experimentations. We discuss certain aspects of CE syntax, semantics and the lexical model as examples. We also show sample CE-based knowledge-sharing capabilities.

Keywords: Ambiguity, coalition operations, Controlled English, decision-making, syntax and semantics, information extraction, linguistic variations, knowledge sharing, multi-nation collaborations, unstructured data

1 Introduction

As science and technology continue to advance, the volume of available data and information has been rapidly growing in both structured and un-structured forms, generated from a variety of sources including business processes, government and organizational policies, scientific activities, public web commentary as well as sensors and intelligence reports. These large data sets created by modern technologies, diverse in form and content, contain valuable and often critical information that, if acquired and properly represented, can provide significant insights to improve knowledge-sharing and to support decision-making. However, the data are creations of human design with bias, and correct interpretation of data requires domain knowledge. As a result, information acquisition from the available data and knowledge-sharing among or across organizations are difficult. A major difficulty, as we observed in the military coalition context, comes from the fact that organizations (even related organizations) may have somewhat different underlying conceptual models of the world in addition to linguistic variations in terms of terminology, sentence structure, language usage and style. For structured data, metadata may also vary in semantics between domains; identical metadata elements may be used to refer to similar but distinct concepts. It is clear that we not only need automated data analysis tools but also a user-friendly common language that enables unambiguous information representation and facilitates a closer human-machine interaction as well.

In this paper, we present our on-going research on a controlled English being developed within a collaborative research alliance called the International Technology Alliance (ITA)[1] to support information-acquisition and knowledge-sharing for decision-making during coalition operations. Section 2 introduces ITA Controlled English (ITA CE, hereafter CE) and discusses our on-going effort to extend CE. Section 3 discusses tools and applications based on CE, including our initial implementation of CE for fact-extraction by various users, from knowledge engineers and linguists to non-technical domain-specialist users. Finally, section 4 summarizes CE and future work on CE extension in syntax and semantics for general expressivity in order to be able to capture and represent a diversity of concepts and to support a wider range of applications in the context of military coalitions.

2 ITA Controlled English

2.1 Controlled Natural Language

A controlled natural language (CNL) is a subset of a natural language using a restricted set of grammar rules and a restricted vocabulary. Well-known examples of controlled languages include ACE [1], CPL [2], PENG [3], Rabbit [4], Caterpillar Fundamental English [5], and STE (Simplified Technical English) [6] , with a common goal: to eliminate (or reduce) the ambiguity and complexity of a natural language, and thus to improve (or at least maintain) readability of the text for humans while allowing it to be processable by machines. ITA CE is an ITA variant of a controlled natural language originally developed by John Sowa [7].

It should be noted that there is often a tension between the human user-friendliness and machine predictability [8]. Predictable interpretation and reliable computation of

[1] In 2006, the US Army Research Laboratory (ARL) and the UK Ministry of Defence (MoD) established a collaborative research alliance with academia and industry partners called the International Technology Alliance (ITA) to address fundamental issues in Network and Information Sciences to enhance the abilities of the US and UK to conduct coalition operations.

a CNL requires deterministic property of the language. But a deterministic CNL is not necessarily easy for users who have not had any training in the CNL, because the restricted grammar and lexicon of the CNL may compete with his/her normal English intuition (i.e., the grammar and lexicon that the user has been exposed to since his/her birth). In general, the closer the CNL to the normal natural language, the more natural and the easier to use by humans, but the less predictable and the more computationally complex it will be for machines. How to achieve a good balance between these two is an important criterion as we design and develop the CE.

We need to make a distinction between ease of reading and ease of writing. In general, CNLs will be easier to read than to write, since the specific restrictions in syntax and vocabulary are more difficult for the writer to remember and follow. To this end, we have developed the CE Query Builder tool to help with the writing of CE, as discussed below in Section 3.1.

2.2 Goals of ITA CE

In addition to the general goal of balancing human usability and machine interpretability, CE should provide:

- A single, standardized language for various users, from different groups (e.g. the UK and the US) but also with different roles in the overall system (for example, end-users or soldiers in the battlefield, military analysts and planners, and system developers)
- The ability to express the basic facts (or propositions) required for a particular domain or application
- The ability to express the epistemic status of propositions (i.e. whether they are true or false, or just an assumption, possibly with a specified degree of certainty), or who believes the proposition; not all propositions will be assumed to be true with the same confidence
- Logical inference rules to allow the inferencing of additional facts from an initial set; in order to encode anything more than the most trivial kinds of knowledge, the ability to infer new facts from existing ones is necessary
- The ability to express the rationale behind a particular proposition; given the ability to infer new facts from old ones, decision makers need to understand the provenance and rationale of the facts on which they base their decisions
- Extensibility: the ability for users with various roles to extend the language in different ways; no system will ever be complete and the easier it is for various types of users to add new knowledge, the more adequate the system will be

2.3 Examples of ITA CE

The current version of CE is roughly consistent with First Order Predicate Logic and with existing ontology modeling languages such as OWL (Web Ontology Language).[2] It provides an unambiguous representation of information for machine processing, while aspiring to provide a human-friendly representation format that is directly targeted at non-technical domain-specialist users (such as military planners, intelligence analysts or business managers) to encourage a richer integration between human and machine reasoning capabilities [9]. In addition to more traditional areas such as knowledge or domain model representation and corresponding information, CE also encompasses the representation of logical inference rules [10], rationale (reasoning steps) [11], assumptions, and statements of truth (and certainty).

The CE currently permits a set of "plain" English sentences for stating propositions referring to entity existence, properties and relations:

- there is a person name Fred.
- the person Fred has French as language.
- the person Fred is married to the person Jane.

The CE also permits meta-statements that specify information about propositions such as their truth status or assumption:

- it is true that there is a person named Fred
- it is assumed that the person Fred is married to the person Jane.
- it is true to degree CV that Fred is a father.

Queries of a set of facts represented in CE can be made in forms like the following:

- for which X is it true that the person X is married to the person Jane

[2] It's the semantics rather than the syntax of CE that is compatible with FOL and existing modeling languages. The current relation between CE semantics and FOL is as follows: 1) the basic CE sentences are all given a FOL semantics in the definition of CE reference; 2) there are some parts of CE that have not yet been given a formal semantics in CE (such as the assumption-based logic); 3) not all of FOL can be represented in CE, for example certain combinations of existential quantifiers embedded in the scope of a universal quantifier. The relationship between CE and OWL (semantics) is as follows: 1) there are some parts of CE that cannot be represented in OWL, e.g. rules, assumptions, although potentially these could be represented in extensions to OWL such as RIF. 2) There are some parts of OWL that cannot (easily) be represented in CE, e.g. lists of explicit values for properties 3) There are one or two fundamental differences in philosophy, for example we prefer to make (nearly) all of the rules of inference to be explicit, whereas in OWL there are many implicit rules of inference. For more on the relationship between CE, FOL, and web modeling languages, see [10].

- for which X, Y and Z is it true that
    the person X is the brother of the person Y and
    the person Y is the father of the person Z

The creation or extension of a domain model (or a general model across domains) using CE is accomplished by the definition of (domain) concepts, relationships and properties. These are all achieved through the “conceptualise”[3] statement:

- conceptualise a ~ person ~ P.

A conceptualise statement creates the concept in question within the CE domain model. The concept is assumed to have a unique meaning, allowing unambiguous interpretation of a text string such as “person” in the above examples. The conceptualise statement introduces new concepts (including entity and event types, as well as their attributes and relations they can enter into) by putting them between tildes and using capitalized letters/strings for variables that would be replaced in a fact assertion or proposition statement.

Slightly more advanced examples are:

- conceptualise a ~ person ~ P that is an agent.
- conceptualise the person P
    that has the value H as ~ height ~ and
    has the value W as ~ weight ~
- conceptualise the person P
    ~ is married to ~ the person P2

The first CE sentence creates “person” as a sub-concept of “agent” and the second indicates that it can have the properties “height” and “weight”. The last sentence asserts that it can have a “married” relationship with someone.

The meaning of a concept can be more fully defined by rules representing the logical relations between concepts and their properties. Logical rules can be represented as follows:

- if PREMISES then CONCLUSION

    if (the person X has the person Y as brother) and
      (the person Z has the person X as parent)
    then
      (the person Z has the person Y as uncle)

[3] The spelling of "conceptualise" is due to the origin of CE at IBM, UK.

If there are CE facts in the repository satisfying the first two premises (where the matching variables must have matching values), then the conclusion can be asserted, again with variables filled in from the premises, in accordance with traditional modus ponens inferencing.

Rationale for a particular proposition is based on the following form of statement:

- CONCLUSION because PREMISES

  the task T1 has the agent A1 as executor
  because
  the plan P1 has the agent A1 as executor and
  the plan P1 contains the task T1.

A rationale would consist of a chain or network of such statements, tracing the provenance of a particular assertion and allowing the user to see the source and status (assumption or fact, or degree of certainty) of the different premises the fact is ultimately based on. The rationale also provides a basis for more complex processing, for example propagating degrees of certainty according to some particular theory uncertain inferencing, or simply highlighting assumptions or propositions with degrees of certain below a particular threshold, although we have not investigated this yet.

Note that the meaning of a concept is given by the inferences that can be made from CE statements that use that concept. These, in turn, are related to the place of the concept in a domain concept is-a hierarchy, the various attributes it can take and the types of relationships it can enter into, and ultimately the logical inference rules that it participates in.

Clearly, the examples given above are simplistic and basic, but with these simple mechanisms, CE has been used in practical applications with reasonable coverage, which we will discuss in Section 3.

2.4 Extending CE

As mentioned above, an important goal of CE is extensibility of the language. The conceptualise construct already allows for lexical extensibility, introducing new concepts for things and their associated properties and relationships[4]. However, it does not provide an ability to extend the syntax of CE. It is desirable to allow extension of the syntax because there are some areas where, although the concept can be expressed in CE, the expression is not very natural. For example, CE currently does not have adjectives per se. Adjectival expressions are captured via noun-like concepts. So in order to say that “my car is red”, we have to say:

- The car mycar is a red thing.

where “red thing” is an unanalyzed concept, despite the space in its name.

[4] Note that “thing” here is at the highest level in the ontology, including things like events and situations.

To allow more felicitous expression of facts like these, we have been working on a set of possible extensions to CE syntax and semantics. While it would be possible to introduce some of these extensions directly into the language, we have been trying to develop a more general means of extending the syntax and associating semantics with it to allow developers and advanced users to make additional extensions as needed in the future. One approach we have experimented with is to use simple transformation rules to capture the linguistic structures that share the same semantics with the existing CE sentences [12]. For example, the construction "there is a red car named mycar" could be transformed into "the car mycar is a red thing" by the following rule:

- there is a <name1> <noun2> named <name3>

  ==>
  the <noun2> <name3> is a <name1> thing

where items contained in angle brackets are patterns that match certain components in the sentence (for example a <noun> is a word that is contained as a concept in the current conceptual model), and the double arrow expresses a mapping from extended to basic CE:

- extended CE ==> basic CE.

This allows straightforward extensions without the need to define new semantic interpretation rules. The only "hard" extension we would have to build into CE is the mapping syntax. The obvious drawback of this approach is that the possible extensions are rather limited. Later, we will briefly mention another more flexible approach to defining extensions to the language that we have begun to explore.

3 CE-based Capabilities and Applications

An extensible CE is most useful in situations that have the following characteristics:

1. A high degree of human-computer interaction, usually involving specialist users with complex needs in non-trivial environments.
2. A likelihood of rapidly evolving or uncertain tasks, queries or other knowledge-based activities.
3. The need for collaboration, either between different people or teams, and/or across different disciplines.

CE is of less value if there is no human-involvement, little complexity, or very firm and stable requirements, and in such circumstances traditional application development processes are a much more straightforward and low risk solution. In cases where there is a high degree of customization, development, uncertain requirements or short lead times, especially in areas where human-led planning, thinking or decision-making are required, then CE (or similar human-friendly information processing environments) could be a very useful capability. Along these lines, we have developed (and are continuing to develop) CE-based capabilities such as "CE Store" to support military coalition applications.

3.1 CE Store: a Development Environment for CE-Based Applications

CE Store is a research-grade runtime implementation of the CE language and ecosystem. It provides a basic CE processing environment that allows for the relatively quick and efficient testing of new concepts in CE and the development of prototype applications. CE Store includes the following high-level capabilities:

1. Basic CE sentence parsing
2. Definition and extension of any concept model (i.e. the possible types of things, including events and situations, and their possible attributes and relationships)
3. Assertion of any CE sentence conforming to a concept model
4. Loading and querying of any existing concept model and associated sets of facts
5. Definition and execution of any CE query including an example "visual query composition" element
6. Definition and execution of any logical inference rule, in the form of a "query with conclusion clauses" that can be used to assert new CE information
7. Definition and execution of "CE agents" which conform to a simple "CE Store" interface

CE Query Builder (CEQB) is a visual query drawing tool embedded in the CE Store environment, which makes use of drag-and-drop and contextual (popup) menus to allow the user to draw, execute and save a CE query or rule. It provides one means to help the user write more accurate rules and queries. The intention of the CEQB is that it is a useful "exploratory" environment in which CE queries can be constructed in a convenient manner. Once the query (or rule) is constructed it can then be saved, executed, etc. The CEQB is a "model aware" component of the environment and is directly integrated into the CE Store APIs. Therefore it will allow one to create queries relating specifically to the information that is currently loaded into the CE Store, indicating the existing concepts but also allowing the user to create new concepts via the conceptualize statement and new rules employing these and other concepts. Note also that, while the query is constructed graphically, the results are displayed in CE.

The purpose of the CE Store is to demonstrate an (almost) "pure" CE-based implementation of an information-processing environment within which human and machine agents can contribute and interact with complex information based on common conceptual models of a domain. In addition, the concept of CE Agent is core to the CE Store approach, allowing domain-specific modules to be constructed and integrated into a CE application. CE Agents may be constructed completely in CE or, if necessary or more convenient, constructed in Java with a CE interface or API to the rest of the CE Store. Java-based CE Agents are used mainly in two cases: 1) When the functionality required is not expressible in if-then rules - e.g. low level text preprocessing or complex algorithmic or statistical processing like that needed for spatial information processing, or 2) When the writing of rules to express the required behavior is too complex or too tedious. It is more likely that, people with roles related to the infrastructure of the system and with IT backgrounds would develop the Java-based agents, if necessary and other users would simply interact with them in CE.

There is a publically available version of the CE Store, known formally as the "IBM Natural Language Processing Environment", available for download from the IBM developerWorks site, here:

http://ibm.co/RDIa53

3.2 Example Applications

CE has been used in the development of a number of prototype systems, including fusion of hard data (from sensors) and soft data (human reports) for situation awareness on the battlefield [13], real-time integration of maps, photos, and messages about events [14], and collaborative planning [15]. The different applications have been based on different versions of CE and have focused on different areas of development of the language. The focus of this paper is not on applications, but the following gives a brief overview of some of the applications that have been developed with CE. See the particular papers for more details.

The data fusion application uses CE to manage the direction, collection, processing, and dissemination of data to support decision making. These require the expression of information needs, the description of asset capabilities, and the conversion of information products generated by each asset into a machine-processable form, consistent with the metadata specified for that asset. This supports the processing and ultimate delivery of data to meet the original information needs [13].

The second application provides for the assignment of objects like buildings and vehicles to locations on a map, the association of photos taken by agents in the field of those objects with icons on the map, and the identification and location of objects extracted from short human generated messages on the map [14].

The "Collaborative Planning Model" is a multi-layer set of conceptual models to enable collaborative planning as a specialized form of general problem solving with support from higher level models for spatial and temporal reasoning [15].

3.3 CE-based Fact Extraction

If a system is developed from scratch, then CE can be defined as the language to use in interacting with the system or with other humans through the system. However, in many applications, there is already a great deal of information available in unstructured form (i.e. free text) that has not been written in controlled English. In order to make this information available, we are currently developing a more extensive and in-depth Fact Extraction system based on CE. Our motivation for using CE to develop this system is the many sub-domains that need to be developed and the rapid rate of change of the conceptual model and language, requiring users at various levels and with various roles to participate in its extension, from system infrastructure builders, to linguists and knowledge engineers, to end-users.

Our method employs natural language processing techniques to parse the language text, recognize the sentence structures, detect properties of the analyzed sentence units, and identify and extract the targeted information items, such as entities, events, relations and facts. CE is used in two different roles in our system: 1) to express the content of the extracted information, using domain specific terminology as specified in the underlying conceptual model, and 2) to describe the linguistic structure of the natural language text being analyzed and to express the processing rules used to get from the natural language text to that expression of the content in CE.

In order to map between the syntax of the sentence and the semantics of the domain, we are currently employing an open source parser, (specifically the Stanford Parser [16]) to provide a basic syntactic parse tree, allowing users to focus on the mapping of this parse tree into the meaning of the sentence, i.e. the specific entities, events, and situations represented in the analyst's conceptual model for the domain. The parse tree produced by the Stanford parser is converted into CE, so that it is amenable to processing by other CE rules.

A lexical model has been constructed in CE to support language processing and the construction of application demonstrations, such as the second application described above. This model is constantly being developed further to provide more complex linguistic concepts, while continuing to support the more basic applications. Recent extensions include:

8. representation of morphologically related sets of words
9. representation of lexical semantics

In the extended lexical-model, we introduce the notion of grammatical form, which includes grammatical information about the word such as the "part of speech" and inflectional features. In the lexical model, grammatical forms are represented in CE, For example, the following is a partial CE representation of a verb:

- conceptualise

  the grammatical form GF ~ is an inflection of ~ the grammatical form GF1

- conceptualise

  the grammatical form GF has the value V as ~ person ~ and
  has the value V1 as ~ tense ~

The concept of grammatical form permits the association of different forms of the same word (for example the singular and plural forms of a noun, or the various forms of a verb) with the lemma or base form of the word and therefore with each other, as well as linking them with their shared lexical meaning while distinguishing their inflectional meanings.

As information requirements vary from one domain to another, we are taking an ontology-based information extraction approach. The domain conceptual model that provides explicit specifications of concepts within the domain plays a crucial role in our information extraction process. Entities and events are primary types of information to be extracted, with entities typically corresponding to noun phrases and events or situation typically corresponding to verbs (together with their arguments and modifiers. Our system correlates the conceptual representations and lexical/grammatical representations by means of the “expresses” relation between words and phrases on the one hand and concepts in the domain model on the other, e.g.:

- the singular noun NN expresses the entity concept EC[5]

In order to specify the semantics of an entire phrase or sentence, we are developing the idea of a linguistic frame, which specifies syntactic structures with grammatical relations and other necessary (or optional) components. For each linguistic frame, there is a unique semantic interpretation rule. Thus, a linguistic frame is a complete description of a grammatical structure including the syntax and semantics. For example, a basic transitive verb phrase defines a relationship based on a transitive verb (with certain morphological, syntactic and semantic features) followed by an (object) noun phrase of certain type, and has the semantics of the relation concept defined. These are defined in such a way as to preserve the semantics of the phrases they cover and pass the resulting semantics up, thus allowing them to incorporate the principle of compositional semantics [17] while at the same time allowing specific construction-level semantics to be added, if needed.

For example, the following linguistic frame describes a predicate nominative construction with “is”, e.g. “Ms. Davis is a professor”:

- there is a linguistic frame named vp1 that

  defines the verb phrase VP and

  has the sequence
  (the present third singular verb '|is_VBZ|' , the determiner '|a_DT|' , and
  the singular noun NN)
  as syntax and

  has the statement that

[5] Note: a more recent model has “word sense” linked to as an additional entity linked between the grammatical form and the domain conceptual model.

(the singular noun NN expresses the entity concept EC)[6]
as preconditions and

has the statement that
(the thing X realises the entity concept EC)
as semantics and

has the thing X as phantom variable.

The “phantom variable” is used to specify an unrealized variable in the definition of the transitive verb that will be matched with the subject at the sentence level, somewhat in the manner of the lambda calculus [17].

This is similar to a feature-based phrase structure approach [18] [19], but there are two differences: (i) the linguistic frames for CE are typically more specifically defined (e.g. the above example including a specific verb); (ii) the linguistic frames in our approach are written in CE. Note that the current CE adopts the “one meaning per CE word[7]” principle and has transparent syntax-semantics mapping, allowing no ambiguity. This will ensure deterministic interpretation for each linguistic frame.

We are currently exploring the use of lexical resources like WordNet [20] and VerbNet [21] to aid both developers and end-users in lexical development, helping them to determine which sense of a word they wish to add and whether it already exists in the model (perhaps under a different name). VerbNet is organized into verb classes based on Levin classes [22], and incorporates both syntactic and semantic information. It encodes detailed information about possible syntactic realization of the argument structure for English constructions such as transitive intransitive-alternations. With its assignment of different roles and types to different arguments (both noun phrases and other phrases), VerbNet can aid the system in the correct parsing of sentences with those verbs. We have developed code to convert these resources into CE and are developing the machinery within CE to use them to perform these functions.

4 Conclusion and Future Work

In Summary, given large volumes of structured and unstructured data, information acquisition and knowledge sharing will need common information structures and representations that are unambiguous to support information sharing and interoperability among teams and team members across domain boundaries. This is the primary motivation for our work on CE and CE-based capabilities.

---

[6] Note: as above, a more recent model has “word sense” linked to as an additional entity linked between the grammatical form and the domain conceptual model.

[7] A “word” in CE is actually referred to as a “concept term”, since it is assumed that each such “word” maps uniquely and unambiguously to a single concept.

While the current CE is still basic, it has been used in a number of example applications as discussed above to model and interact with complex real-world environments, with a reasonable number of concepts, relationships, and rules. We are currently focused on three efforts: 1) extending the CE language to make it more natural, 2) improving the fact extraction capabilities of the system, and 3) continue leveraging external linguistic sources like WordNet and VerbNet, both for the basic fact extraction system and to assist end-users with only folk-linguistic knowledge to in extending conceptual models for their domain along with the associated lexical knowledge.

Our major effort currently focused on extending the CE language to make it more natural. For CE extension, we have no intention to include all the grammatical structures of English. Instead, we will continue to focus on basic English phrase and sentence structures, especially those that are structurally unambiguous. We have explored the use of transformational rules to extend CE basic syntax. However, there is probably a more powerful and flexible way of doing that. As discussed above, we are using linguistic frames to define natural language structures. We are planning to investigate the use of the same mechanism to extend CE itself. CE is an example of a natural language and the linguistic frame provides a way of specifying new syntactic structures and their associated semantics. This means we would be using CE to extend CE, which should not be a problem as long as it is ultimately grounded in some very fundamental CE.

Managing ambiguity is fundamentally important for the work of extending CE. As we extend CE, potential ambiguity may arise from either lexical resources or potentially ambiguous syntactic patterns. In English, the majority of words are inherently ambiguous. As mentioned throughout the paper, ITA CE is intended to be highly predictable with no ambiguity allowed. We define the relation between a word and a concept by using a ‘conceptualise’ statement but the definition exercise will need to be based on domain relevance including the data used in the relevant domain and the conceptual domain model so as to ensure the right definition of the lexical meaning for the lexical item in question. Possible ambiguities can also result from syntactic structures that allow multiple interpretations, including the well-known example of prepositional phrase attachment. Extended CE will define the most natural interpretation as the only possible structure, again based on the empirical data.

ACKNOWLEDGMENT. This research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

References

1. Fuchs, N. E., Schwertel, U., and Schwitter, R.: Attempto Controlled English. Proceedings of LOPSTR'98 (1998)
2. Clark, P., Harrison, P., Jenkins, T., Thompson, J., and Wojcik, R.: Acquiring and Using World Knowledge Using a Restricted Subset of English. In Proceedings of FLAIRS'05 (2005)
3. Schwitter, R.: Processable English. See http://web.science.mq.edu.au/~rolfs/peng/ Retrieved August, 18 (2010).
4. Engelbrecht, P., Hart G., and Dolbear, C.: Talking Rabbit: a User Evaluation of Sentence Production. Ordnance Survey. Workshop on Controlled Natural Language (CNL 2009). 8-10 June 2009. Marettimo Island, Italy. Appears in Controlled Natural Language, Volume 5972, of Springer’s LNCS/LNAI series (2009)
5. Verbeke, C. A.: Caterpillar Fundamental English, Training and Development Journal, 27, 2, 36-40, Feb 73 (1973)
6. MacDonald, M.L.: Simplified Technical English for All: A Customer-friendly Specification. AeroSpace and Defence Industries Association of Europe (ASD), http://www.x-pubs.com/resources/2008conf/downloads/4X-Pubs2008_Maria_McDonald_Simplified_Technical_English_For_All.pdf (2008)
7. Sowa, J.: Common Logic Controlled English, March 2007, http://www.jfsowa.com/clce/clce07.htm (2007)
8. Clark, P., Harrison, P., Murray W. R., Thompson J.: Naturalness vs. Predictability: A Key Debate in Controlled Languages. Workshop on Controlled Natural Language (CNL 2009). 8-10 June 2009. Marettimo Island, Italy. Appears in Controlled Natural Language, Volume 5972, of Springer’s LNCS/LNAI series (2009)
9. Mott, D.: Summary of Controlled English, ITACS, https://www.usukita.org/papers/5658/details.html (2010)
10. Mott, D.: The representation of logic within semantic web languages, ITACS, url: https://www.usukita.org/papers/5242/details.html (2009)
11. Mott, D., Giammanco, C., Braines, D., Dorneich, M., and Patel, D.: Hybrid Rationale and Controlled Natural Language for Shared Understanding. In Proceedings of the Fourth Annual Conference of the International Technology Alliance, London, UK, September (2010)
12. Mott, D and Hendler, J.: Layered Controlled Natural Languages, In Proceedings of the Third Annual Conference of the International Technology Alliance, Maryland, USA (2009)
13. Preece, A., Pizzocaro, D., Braines, D., Mott, D., de Melz, G., and Pham, T.: Integrating Hard and Soft Information Sources for D2D Using Controlled Natural Language, April 2013, SPIE Defense, Security, and Sensing (2013)
14. Braines, D., Mott, D., Laws, S.: Controlled English to Facilitate Human/machine Processing. April 2013, SPIE Defense, Security, and Sensing (2013)
15. Dorneich, M. C., Mott, D., Bahrami, A., Allen, J., Patel, J., Giammanco, C.: Lessons Learned from an Evaluation of a Shared Representation to Support Collaborative Planning. In, 7th Knowledge Systems for Coalition Operations, KSCO (2012)
16. Klein D. and Manning, C. D.: Accurate Unlexicalized Parsing. Proceedings of the 41st Meeting of the Association for Computational Linguistics, pp. 423-430 (2003)
17. Cann, R.: Formal Semantics: An Introduction, Cambridge University Press, Feb 6 (1993)
18. Gazdar, G., Klein, E., Pullum G., and Sag, I.: Generalized Phrase Structure Grammar. Harvard University Press, Cambridge, MA (1985)

19. Pollard C. and Sag, I.: Head-Driven Phrase Structure Grammar. University of Chicago Press, Chicago, IL (1994)
20. Fellbaum, C. (ed.). WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press, Cambridge, MA (1998)
21. Edward, L., Yi, S., and Palmer, M.: Combining Lexical Resources: Mapping Between PropBank and VerbNet. Proceedings of the 7th International Workshop on Computational Semantics. Tilburg, the Netherlands (2007)
22. Levin, B.: English Verb Classes and Alternations: A Preliminary Investigation, University of Chicago Press, Chicago, IL (1993)