From Simplified Text to Knowledge Representation using Controlled Natural Language

Hazem Safwat, Manel Zarrouk and Brian Davis

Insight Centre for Data Analytics,
National University of Ireland,
Galway, Ireland
{hazem.abdelaal,manel.zarrouk, brian.davis}@insight-centre.org

Abstract. Knowledge based systems provide means to store data and perform reasoning on top of it. Controlled Natural Language (CNL) is considered as an engineered subset of natural language. CNLs aim to abstract the complexity level of natural languages and reduce/abolish the characterizing ambiguity of these languages. Two major types of CNLs exist. The one aiming to improve human-human communication is called Simplified Language or human CNL. The second type is a machine-readable CNL, which is formal language used mainly as an Abstract Knowledge Representation (AKR) Language. In this paper, we present an approach to translate the Medline summaries of diseases presented in a Simplified English to a machine readable CNL. This approach will result in a formal Knowledge Base enabling machine processing, querying and reasoning while avoiding the usual main obstacle which is natural language ambiguity. We tested the approach on a sample corpus of the Anemia disease, and presented the concluded hypothesis.

Keywords: Natural Language Processing; Controlled Natural Language; knowledge Base Systems, Simplified Language, machine-readable CNL.

1 Introduction

In the recent years, the web has witnessed multiple efforts to control the massive explosion of data from different sources such as web pages, articles, mailing lists, etc. The common requirement for all these efforts is to represent this large amount of data in a way that allows their effective use [1]. The development of knowledge bases and expert systems make it possible to store this data in a structured form like ontologies, then perform some reasoning on top of it. Although the research in the area of knowledge representation is very active, most of the current systems are focusing on the use of knowledge base systems by expert and domain users. Only few studies were aiming for non-expert users who can not create ontologies until they learn a formal language like OWL.

For non-expert users to overcome the restriction of learning formal data representation and ontology creation, we argue that CNLs could be another alternative for knowledge creation and management. CNLs offer user-friendly means

for knowledge representation to non-expert users [3]. A CNL is an engineered subset of natural language that is restricted to the lexicon, the syntax and the semantics to be unambiguously machine-readable. It is a formal language represented in the form of natural language [2]. Thus, rewriting a natural language text into a CNL can provide an attractive solution for building knowledge bases for non-experts. Since the CNL itself is restricted, it would be difficult to rewrite an unrestricted text into CNL. However, this might be possible to apply on simplified text, also known as Human-oriented CNL defined by [4] as a language to improve the readability and comprehensibility of technical documentations (e.g. ASD Simplified Technical English[1]) as well as improve the communication among people for specific purposes (e.g. air traffic control). Simplified text such as Simple Wikipedia[2] is a form of Human-oriented CNL, written using style guides to avoid complex and ambiguous syntax for juniors and second language learners. This paper presents a model to rewrite simplified text into a CNL as an intermediate step in the knowledge base population process using rewrite rules. Rewrite rules have been previously used in different approaches targeting text generation such as machine translation, text simplification, question answering, and information retrieval [5]. However, an automatic rewriting system to convert text into a CNL is novel to our knowledge [6]. In this research, we show in a preliminary case study that it is possible to convert simplified text into a CNL using rewrite rules. Importantly, our approach can be used to populate a knowledge base with the CNL output of the conversion process, that could be used for different NLP/Semantic web applications such as querying, question answering, and semantic reasoning.

In the remainder of this paper, Section 2, presents an overview of the related work. Section 3 introduces the used corpus. In section 4, we explain our methodology and the proposed architecture to develop the rewriting system. Section 5 presents a case study of the Anemia disease. Section 6 discusses the evaluation criteria of the system, and finally, Section 7 outlines the discussion and the ongoing tasks.

2 Related Work

An early approach to address sentence transformation for NLP applications using rules was presented in [7]. The authors used hand crafted syntactic rules for text simplification over long sentences. In [8] the authors developed a controlled language rewriting system called KANT. The system uses a strictly defined controlled language with a formally specified syntax to guarantee automatic machine translation with high quality. The KANT controlled language does not limit the size of the vocabulary, and only rules out complex lexical and grammatical constructions. The implementation of the system combines a) Constraints on the lexicon to reduce lexical ambiguity and complexity, b) Constraints on the grammar to reduce parsing complexity, c) Using standardized General Markup

[1] `http://www.asd-ste100.org/`
[2] `https//simple.wikipedia.org/wiki/Main\_Page`

Language (SGML), to support the definition of domain terminology and phrasal constructions. The KANT rewriting engine utilizes a prescriptive rewriting approach, where any sentence that is not predefined in the controlled language grammar will not be parsed by the grammar checker, and must be rewritten. The author can resolve the parsing failures by rewriting the sentence and make sure it passes the grammar checker. The system has been deployed by Caterpillar Inc. to guarantee high quality authoring and translation of their complex technical documents.

The previous work presented in [9][10][11][12] tried to extract relations between text entities, or develop an ontology learning approach to create structured data in the form of RDF from plain text. The limitations of these approaches is that they rely on the precision of the extracted linguistic patterns to formulate semantic relations between the text entities, and accumulate the results on larger data sets of a predefined form.

In [15] the authors present an approach to convert a biomedical query written in a CNL named BIOQUERYCNL, into a program in a knowledge representation and reasoning paradigm called Answer Set Programming (ASP) [16] via Discourse Representation Structures (DRS) [14]. The advantage of this is to automate reasoning about biomedical ontologies using the ASP solvers. Our work is different, as we focus on the knowledge creation and representation and we do not include the querying part. Other approaches studied transforming the Natural Language (NL) into ASP include [17], where the authors present a research to transform simple sentences into ASP using Lambda calculus. As well as, the work presented in [18], where the authors use the Boxer framework [13] to perform semantic analysis over the output of the C&C parser, and then the reasoning is done using ASP. Although Boxer/C&C tools can parse and create a semantic representation in DRS for a natural language, the parser will not perform well in the any domain since it was trained on newspaper text corpus [19].

Another approach is studied in [20] and [21] to extract entities and relations from a military domain corpus [22], and to represent this knowledge using a CNL called Controlled English (CE). The main aim was to build a Knowledge Base (KB) that supports knowledge sharing and decision making across different groups without the need to transform the text into a formal notation, so that all users from different backgrounds can understand it. The approach is still in the development phase, and it is different from the approach presented in this paper in which we transform the whole text without extracting particular entities and relations from the text.

3 The Medline summaries corpus

The corpus used in our work is the Medline diseases summaries[3]. The corpus covers a summary of all the popular diseases. Each summary includes several

[3] `https://www.nlm.nih.gov/medlineplus/healthtopics.html`

sentences about the disease explanation, symptoms, causes, diagnosis, and treatment. Each summary ranges from 10 to 20 sentences. We selected this corpus specifically, as it fulfills the properties of a simplified language corpus. The corpus is written using style guides[4] to make the text easy to understand by any person regardless of age, background and reading level. Since the medical concepts and language are usually complex and difficult to read, the aim of the style guides is to help the authors write an easy to read health materials. The style guides involve organizing the writing to keep it within the range of 7^{th} or 8^{th} grade reading level. This could be achieved by finding alternatives for complex words or abbreviations, avoiding abstract syntax language, limit sentences length to be between 10 and 15 words, and using the active voices instead of the passive ones.

4 The System

4.1 Attempto Controlled Language (ACE)

We will use a CNL called ACE [23], a well known CNL used as a general knowledge representation language designed for authoring and verbalizing ontologies. It has an English syntax that can be automatically and unambiguously mapped into DRS, a variant of First Order Logic (FOL). Furthermore, the syntax can be mapped into other formal languages supported by existing automatic reasoners (e.g. OWL), using the Attempto Parsing Engine (APE) that includes all ACE construction and interpretation rules. Thus, ACE syntax is easily readable and understandable by humans as it is seemingly a natural language, and unambiguously processable by the machine as a formal language. In short, we selected ACE CNL for representing the formal knowledge, as it is the most developed CNL up-to-date with an expressive grammar supporting various complex structures.

4.2 System Overview

As shown in figure 1, the approach is divided into two phases, the pre-processing phase for preparing the sentences, constructing the ACE and domain lexicons, then generating parse trees using a general parser e.g. Stanford parser[5]. The second phase is the post-processing, responsible for rewriting the simplified sentences into ACE sentences, and this will involve inserting, replacing, or changing the order of some predicates in the sentence. The previous step will generate many ACE alternatives for each simplified sentence, so a domain expert needs to select the closest semantic alternative for each sentence. The machine will be trained to do this step automatically after scoring the expert selections. The whole process will involve a percentage of information loss which will be evaluated based on the percentage of successfully rewritten sentences.

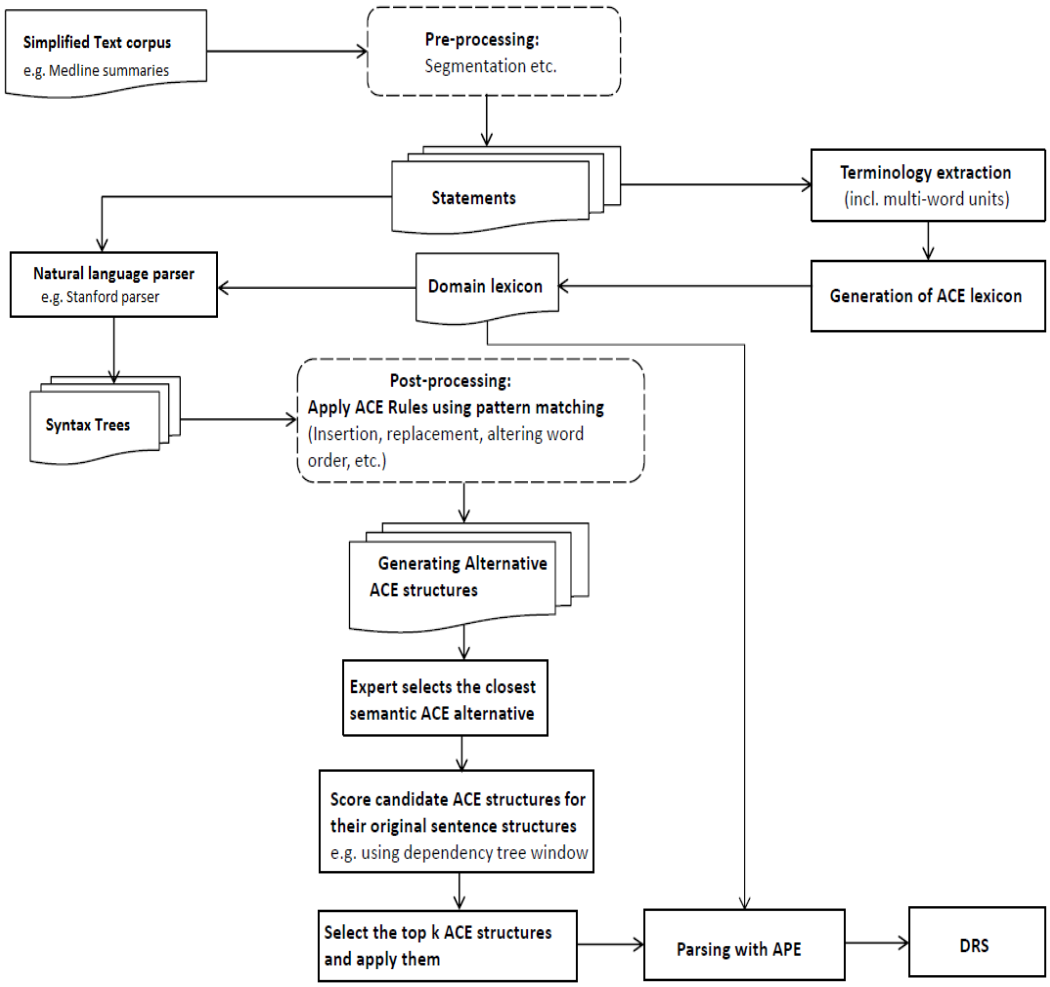

Fig. 1. The proposed system architecture to transform Simplified Language text into ACE CNL.

4.3 The architecture

In the pre-processing phase we will apply the common NLP tasks for preparing the corpus, these tasks include lexical and syntactic analysis based on language resources such as lexicons and grammars. The first step is the segmentation of the entire corpus, where the text is divided into segments separated by full stops. The divided segments of text are in the form of separate sentences. These sentences are normalized to make sure that all the information types are set into a standard format. We can perform this task using conversion rules to set a common standard format in the entire corpus. Terminology is extracted from the complex units (e.g. multi-word units). Both the ACE and domain lexicons need to be generated to be used later when the ACE engine (APE) generates the DRS.

Furthermore, a basic correference resolution module is applied on the sentences. The correference problem happens when it is referred to some entities in the text in different ways. This can occur due to the use of pronouns in our Medline summaries corpus. For instance, the sentence *Hemoglobin is an iron rich protein. It carries oxygen from the lungs to the rest of the body.*, the pronoun *it* refers to the noun *Hemoglobin*. The correference is a common research problem in the NLP field, and it could be solved using different methods based on the complexity of the corpus. The first method is the rule based approach taking into account the semantic information of each entity. The rule will filter the entities and determine the entities that have the highest probability to be correferent. Finding the semantics of each entity can be done manually or using other resources such as Wordnet. Another approach for correference resolution is measuring the distance between two entities, and if the distance between them is less than a predefined cluster radius, then the two entities belong to the same cluster. Furthermore, decision trees are used to solve this problem as well. The decision tree checks if two entities are correferent or not based on some predefined attributes such as gender, number, and semantic class. The correference problem in the Medline disease summaries corpus is not complicated, as the text is already simplified and written using style guides as discussed before. So, we expect that the distance between two entities will be small, and thus we will use the basic rule based approach.

In the post-processing phase each sentence will be in the form of a syntax tree. At this stage the ACE rules are applied on each tree to reconstruct it into an ACE tree. For instance, in ACE each noun should be preceded by an article or a quantifier (e.g. the, each, every, all, etc) which means that we have to insert an article or quantifier before each noun. In the following sentence *your body needs iron to make hemoglobin*, there are three nouns *body, iron, hemoglobin*. The noun *body* is preceded by a possessive *your* that needs to be replaced by either an article or a quantifier in order to follow ACE grammar rules. Also, both nouns *iron* and *hemoglobin* need to be preceded by either an article or a quantifier. In addition to, the constituent *to make* is not acceptable in ACE

[4] `https://www.nlm.nih.gov/medlineplus/etr.html`
[5] `http://nlp.stanford.edu:8080/parser/index.jsp`

grammar. Thus, one alternative would be to replace the constituent *to make* with *for the making of*. Finally, the whole sentence will require 2 replacements and 2 insertions. A possible reconstructed ACE sentence that is acceptable by the ACE grammar and processable by the APE engine could be *the body needs the iron to make the hemoglobin*. However, since there are many articles and quantifiers, the system will generate many possible reconstructions for the previous sentence. For instance, another possible alternative is *the body needs some iron for the making of some hemoglobin*, which seems a good semantic alternative of the original sentence. For this reason, we decided to add a module in the architecture that involves human intervention, where an expert will check each original sentence and all the generated ACE alternatives, then selects the closest semantic ACE alternative among all the options.

The next stage in the architecture involves a machine learning module. The available training and testing data sets are the Medline manually selected ACE trees. The main goal of the system is to automatically rewrite the Medline disease summaries into ACE CNL and store this data in a knowledge base. Thus, we will use a clustering approach to perform the rewriting process automatically while selecting the closest alternative for each sentence and its constituents. In order to achieve this we will give a positive score to the selected candidate structures, as well as negative scores for the other non-selected candidates. Each constituent can be predefined using a syntax tree window that will be based on the sentence structure. For instance, the sentence *your doctor will diagnose anemia with a physical exam and blood tests*, the constituent *blood tests* is originally preceded by *and*, the window consists of conjunction and noun. In this case, our approach will give a higher score to the quantifier *some*, and lower score to the rest of the determiners, so that the final ACE constituent will be *some blood tests*. The final step in this approach is to use the APE engine to process all the highest scores ACE statements and store them in the knowledge base.

5 Preliminary case study

In this section, we will go through our analysis of a sample disease from the corpus. We selected a random Medline summary Anemia disease. In the pre-processing phase, we segmented the text to obtain separate sentences. The whole text was normalized to a standard format, for example we removed the apostrophes, bullet points, hyphens and etc. This will generate a corpus of separate sentences that is ready for further processing. Then, we applied a Part of Speech (PoS) tagger on the generated corpus, to identify the structure of each sentence.

Since, the ACE grammar can not process a constituent of two or more consecutive nouns, we extracted these constituents to chunk them into one complex noun phrase and process it as a multi-word unit. These units will be added to the domain and ACE lexicons. The next step, is to look for the pronouns in the text and apply a correferent resolution clustering approach (e.g. the pronoun "it" refers to the noun "Hemoglobin") to match different names describing the

Table 1: A table showing a NL sentence and its reconstructed ACE alternatives after applying ACE rules

NL sentence	NL parse tree	ACE alternatives	ACE parse trees
Your body needs iron to make Hemoglobin.	(ROOT(S(NP(PRPYour)(NN body))(VP(VBZneeds)(NP(NN iron))(S(VP(TOto)(VP(VB make)(NP(NNhemoglobin)	The body needs (the/some/all/every/each) iron for the making of(the/some/all/every/each) hemoglobin.	(ROOT(S(NP(DTthe) (NNbody))(VP(VBZneeds) (NP(NP(DTthe)(NNiron)) (PP(INfor)(NP(NP(DTthe) (NNmaking))(PP(INof) (NP(DTthe)(NNhemoglobin)

PARAPHRASE

There is a body X1.
The body X1 needs an iron for a making of some hemoglobin.

DRS

```
[A,B,C,D,E]
object(E,body,countable,na,eq,1)-1/2
object(D,iron,countable,na,eq,1)-1/5
relation(B,of,C)-1/9
object(B,making,countable,na,eq,1)-1/8
predicate(A,need,E,D)-1/3
modifier_pp(A,for,B)-1/6
object(C,hemoglobin,mass,na,na,na)-1/11
```

Fig. 2. The DRS representation from the APE engine.

same entity. After applying the normalization and the correference resolutions tasks, each sentence will be parsed using a natural language parser (e.g. Stanford parser) generating the syntax trees. These trees will be needed for further analysis in the post-processing phase. The Anemia sample corpus includes 10 sentences, each sentence will have more than one ACE alternative in the rewriting process. An example of random selected sentence and its alternatives are shown in table 1. The table shows the original sentence "Your body needs iron to make hemoglobin", and its parse tree generated from the Stanford parser. The ACE alternatives present the different structures of the new reconstructed sentences after applying the ACE CNL rules.

Table 2: A table showing the transformation from NL grammar into ACE grammar.

The predicate	predicate grammar	ACE grammar	ACE predicate
your body	PRP+NN	Det+NN	the body
iron	NN	Det+NN	the iron
to make	to+vp	prep+Det+NN+prep	for the making of
hemoglobin	NN	Det+NN	the hemoglobin

The rules applied here are as follows: 1) remove all the possessive adjectives. 2) Each noun should be preceded by an article or a quantifier. 3) ACE grammar does not support to+verb constituent, so it has to be replaced by another alternative. The ACE parse tree shows the parse tree of the closest semantic

Table 3: A table showing the sample corpus, the selected ACE alternatives and their DRS paraphrases.

No.	NL sentence	ACE alternative	DRS paraphrase
1	If you have anemia your blood does not carry enough oxygen to the rest of your body.	if (a person) (has) Anemia (then) (the blood) does not carry (the enough oxygen) to the rest of (the body).	If a person has Anemia then there is a blood X1 and it is false that the blood X1 carries some enough oxygen to a rest of a body.
2	The most common cause of anemia is not having enough iron.	The most common cause of Anemia is (no) enough iron	There is a most common cause X1 of Anemia.If there is an enough iron X2 then it is false that the most common cause X1 is the enough iron X2
3	Your body needs iron to make hemoglobin.	(the body) needs (the iron) (for the making of) (the hemoglobin).	There is a body X1. The body X1 needs an iron for a making of some hemoglobin.
4	Hemoglobin is an iron rich protein that gives the red color to blood.	Hemoglobin is an (n:iron-rich-protein) that gives the red color to (the blood).	There is an n:iron-rich-protein X1.Hemoglobin is the n:iron-rich-protein X1.The n:iron-rich-protein X1 gives a red color to a blood.
5	Hemoglobin carries oxygen from the lungs to the rest of the body.	Hemoglobin carries (the oxygen) from the lungs to the rest of the body.	Hemoglobin carries some oxygen from at least 2 lungs to a rest of a body.
6	Anemia has three main causes blood loss and lack of red blood cell production and high rates of red blood cell destruction.	Anemia has three main causes (that are) (the n:blood-loss) (and) (the lack of the red n:blood-cell-production) and (the high rates) of (the red n:blood-cell-destruction).	There are 3 main causes X1.Anemia has the main causes X1.The main causes X1 are a lack of at least 2 high rates of a red n:blood-cell-destruction and a red n:blood-cell-production and a n:blood-loss.
7	Anemia can make you feel tired, cold, dizzy, and irritable.	Anemia can make (a person) (a tired person) and (a cold person) and (a dizzy person) and (an irritable person).	There is some anemia X1. It is possible that the anemia X1 makes a person a tired person and a cold person and a dizzy person and an irritable person.
8	You may be short of breath or have a headache.	(the person) may (have) (a short breath) and (a headache).	There is a person X1. It is possible that the person X1 has a short breath and a headache.
9	Your doctor will diagnose anemia with a physical exam and blood tests.	(the doctor) (can diagnose) Anemia with a physical exam and (some n:blood-tests).	There is a doctor X1. It is possible that the doctor X1 diagnoses some anemia with a physical exam and at least 2 n:blood-tests.
10	Treatment depends on the kind of anemia you have.	Treatment depends on the (type) of Anemia.	There is a treatment X1. The treatment X1 depends on a type of some anemia.

Table 4: The grammar predicates from each sentence, the ACE grammar conversion and the reference rule applied from the list

No	converted predicates	converted grammar	grammar rules
1	you → a person. have anemia → has the anemia. your blood → the blood. the enough oxygen → enough-oxygen. your body → the body.	PRP → DT+NN. VBP+NN →VBZ+NN. PRP+NN → DT+NN. JJ+NN → DT+JJ+NN. PRP+NN → DT+NN.	1,2,3,4,5
2	not having → no.	RB+VBG → DT.	3
3	your body → the body. iron → the iron. to make → for the making of. hemoglobin → the hemoglobin .	PRP+NN → Det+NN. NN → Det+NN. to+vp → prep+Det+NN+prep. NN → Det+NN .	2,3,6
4	iron rich protein → iron-rich-protein. blood → the blood.	NN+JJ+NN → NNS. NN → DT+NN.	3,5.
5	hemoglobin → the hemoglobin . oxygen → the oxygen.	NN → DT+NN. NN → DT+NN.	3.
6	blood loss → the n:blood-loss . lack → the lack. red blood cell production → the red n:blood-cell-production high rates → the high rates. red blood cell destruction → the red n:blood-cell-destruction	NN+NN → DT+NNs. NN → DT+NN. JJ+NN+NN+NN → DT+NNS. JJ+NN → DT+JJ+NN. JJ+NN+NN+NN → DT+NNS.	3,5,7
7	you → a person. feel tired → a tired person. cold → a cold person. dizzy → a dizzy person. irritable → an irritable person.	PRP → DT+NN. VB+JJ → DT+JJ+NN.	1,3,7. X+feel+Y → if+NP+VP+X+ then+NP+VP+Y.
8	you → the person . may be → may have. short of breath → a short breath. headache → a headache.	PRP → DT+NN. MD+VB → MD+VB. JJ+IN+NN → DT+JJ+NN. NN → DT+NN.	1,3. or → and.
9	your doctor → the doctor . blood tests → some blood-tests.	PRP+NN → DT+NN. NN → DT+NNS.	2,3,5. will → can
10	NA	NA	kind → type

alternative for the original sentence *the body needs the iron for the making of the hemoglobin*, that is selected by the expert and stored in the knowledge base. Table 2 shows how the system applied ACE rules on the previous example sentence to reconstruct and generate a new ACE grammar. In the previous example, since the article/quantifier is the only predicate generating many alternatives and the rest of structure will be know to the system how it could be reconstructed by fixed ACE rules. The system will give high scores to the article "the" over the other quantifiers in the following cases: 1) when the noun is singular and preceded by "your" e.g your body. 2) the noun is singular and is not preceded by quantities e.g. "the iron" and "the hemoglobin". Finally, the new reconstructed ACE sentence will be parsed with the ACE engine to generate the DRS as shown in figure 2.

The rewriting of a NL sentence into an ACE may inquire some modifications in the syntactic structure of the sentence. However, these modifications can change the semantics. We extracted all the patterns from the sample corpus that need to be changed in order for the sentence to be ACE. All the transformed NL sentences from the sample corpus and their ACE alternatives are shown in table 3.

From table 3 we can conclude the following grammar rules, which will be presented in table 4:

1. Possessive adjectives "you" can not be translated into ACE, so we replace it with a noun "person".
2. Possessive adjectives "your" can not be translated into ACE, so we replace it with an article.
3. Each noun has to be preceded by an article or a quantifier.
4. Each if statement should be transformed into if-then statement.
5. Successive nouns has to be combined into one noun.
6. The constituent "to+verb" is not supported in ACE grammar, so it has to be replaced by "for+the+verb+of".
7. Punctuation: use of comma, colon,semicolon, quotation marks, and parentheses as inter- and intra-sentential punctuation should be replaced with the right terms.
8. Modalities are not supported in ACE, so it has to be replaced by "can".

6 Evaluation Criteria

The system will be evaluated on two stages. Since the ACE CNL is restricted, where for example we can not express modalities in FOL. We expect a percentage of information loss during the transformation from NL into ACE. The first question will be, how semantically close is the ACE alternative for each sentence, and what is the quality of this alternative. In order to answer this question we will take a sample from the ACE sentences and the original NL sentences, where this sample should cover all the different structures of the grammar rules defined in the system. Then, we will apply a subjective evaluation such that users will be able to give a score for each grammar rule and its ACE alternative. After that,

we can have a general insight for the information loss percentage. The second question that we need to answer in our evaluations, is the accuracy of the system in regard to selecting the best fitting ACE grammar rule for each predicate. For this evaluation, we require a test set developed by an expert, that could be compared with the system output after the processing is done. Then, we can apply the equations of precision and recall to find the percentage of true/false positives and true/false negatives.

7 Conclusion

In this article, we have presented the initial steps taken towards automatically transforming Simplified Language text into a CNL that is fully machine processable. Then we can store the CNL in a KB to do reasoning. We discussed a manual processing of a tiny sample corpus, where we presented the grammar rules and the patterns that could be used in the transformation process. Although the processing of the sample corpus showed that there are some restrictions in the grammar constructions, we understand that these restrictions go beyond the FOL. Our approach showed that it could be flexible in the terms of providing many ACE alternatives for each grammar construction. While, we analyzed a sample of the corpus in this paper that does not represent the entire grammar constructions of the full corpus, we will analyze the entire corpus in the next step. The process will involve, parsing the entire corpus to generate the syntax trees, then rank the syntax trees according to the most dominating grammar constructions in the corpus. After that, we will take these dominating constructions as a training set to be manually converted into CNL grammar, then apply a clustering approach to classify each new sentence to the closest cluster in the corpus.

References

1. Perry, C.A. (1990) Knowledge bases in medicine: A review. Bulletin of the Medical Library Association 78:271282.
2. Hazem S. and Brian D. CNLs for the Semantic Web: A State of the Art. Journal of Language Resources and Evaluation (2016).
3. Hazem S. and Brian D. A Brief State of the Art of CNLs for Ontology Authoring. In 4th International Workshop, CNL 2014, August 20-22, 2014., pages 190200. Springer International Publishing, Galway.
4. Huijsen, Willem-Olaf. 1998. Controlled Language An Introduction. In: Proceedings of CLAW 98, pp.115.
5. Kristian W. and Mirella L. (2014). Text Rewriting Improves Semantic Role Labeling. Journal of Artificial Intelligence Research, 51:133164.
6. N. Eric, T. Mitamura, and W. Huijsen. (2003). Controlled language for authoring and translation. In Harold Somers, editor, Computers and Translation: A Translator's Guide. John Benjamins Publishing Company, pages 245281.
7. Chandrasekar, R., Doran, C., & Srinivas, B. (1996). Motivations and Methods for Text Simplification. In Proceedings of the 16th International Conference on Computational Linguistics, pp. 10411044, Copenhagen, Denmark.

8. T. Mitamura, E. H. Nyberg, 3rd, Controlled English for KnowledgeBased MT: Experience with the KANT System, Center for Machine Translation, Carnegie Mellon University, Pittsburgh, 1995
9. Cimiano, P., Volker, J.: Text2Onto. In: Montoyo, A., Munoz, R., Metais, E. (eds.) NLDB 2005. LNCS, vol. 3513, pp. 227238. Springer, Heidelberg (2005)
10. Dijkstra, E.W.:A note on two problems in connexion with graphs. NumerischeMathematik 1, 269271 (1959)
11. Cafarella,M.J., Re, C., Suciu, D., Etzioni, O., Banko, M.: Structured querying of web text: A technical challenge. In: Proceedings of the Conference on Innovative Data Systems Research, Asilomar, CA (2007)
12. Li, Y., Chu, V., Blohm, S., Zhu, H., Ho, H.: Facilitating pattern discovery for relation extraction with semantic-signature-based clustering. In: Proceedings of the ACM Conference on Information and Knowledge Management, pp. 14151424 (2011)
13. Curran, J.R., Clark, S., Bos, J.: Linguistically Motivated Large-Scale NLP with C&C and Boxer. In: Proceedings of the ACL 2007 Demo Session, pp. 3336 (2007)
14. Kamp, H., Reyle, U.: From Discourse to Logic: Introduction to Model-theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory. Studies in Linguistics and Philosophy, vol. 42. Kluwer, Dordrecht (1993)
15. Erdem, E., and Yeniterzi, R. 2009. Transforming controlled natural language biomedical queries into answer set programs. In Proc. of the Workshop on BioNLP, 117124.
16. Chitta Baral. 2003. Knowledge Representation, Reasoning and Declarative Problem Solving. Cambridge University Press.
17. Chitta Baral, Juraj Dzifcak, and Luis Tari. 2007. Towards overcoming the knowledge acquisition bottleneck in answer set prolog applications: Embracing natural language inputs. In Proc. of ICLP, pages 121.
18. Chitta Baral, Juraj Dzifcak, and Tran Cao Son. 2008. Using answer set programming and lambda calculus to characterize natural language sentences with normatives and exceptions. In Proc. of AAAI, pages 818823.
19. Johan Bos. 2008. Wide-Coverage Semantic Analysis with Boxer. In J. Bos and R. Delmonte, editors, Semantics in Text Processing. STEP 2008 Conference Proceedings, volume 1 of Research in Computational Semantics, pages 277286. College Publications.
20. D. Mott, D. Braines, S. Poteet, A. Kao, and P. Xue. Controlled Natural Language to facilitate information extraction. In 6th Annual Conference of the International Technology Alliance ACITA, 2012.
21. P. Xue, S. Poteet, A. Kao, D. Mott, D. Braines, C. Giammanco, and T. Pham. Information Extraction Using Controlled English to Support Knowledge-Sharing and Desicion-Making. In 17th ICCRTS Operationalizing C2 Agility, Fairfax VA, USA, 2012.
22. J. L. Graham, D. L. Hall, and J. Rimland. A synthetic dataset for evaluating soft and hard fusion algorithms. In SPIE Defense, Security, and Sensing Symposium, volume 8062, 2011.
23. N. E. Fuchs, K. Kaljurand, and T. Kuhn. Attempto Controlled English for Knowledge Representation. In Reasoning Web, Fourth International Summer School 2008, pages 104124. Springer, 2008
24. Kuhn, T. (2010a). Controlled English for Knowledge Representation. PhD thesis, University of Zurich.
25. ACE View an ontology and rule editor based on Attempto Controlled English. In 5th OWL Experiences and Directions Workshop (OWLED 2008), Karlsruhe, Germany. 12 pages.

26. Tobias Kuhn. AceWiki: A Natural and Expressive Semantic Wiki. In Semantic Web User Interaction at CHI 2008: Exploring HCI Challenges, 2008.