AN ONTOLOGICAL APPROACH TO THE EXTRACTION OF FIGURES OF SPEECH

Christiana Panayiotou

The Cyprus University of Technology, Cyprus

ABSTRACT

The purpose of the current paper is to present an on- tological analysis to the identification of a particular type of prepositional natural language phrases called figures of speech [1] via the identification of inconsis- tencies in ontological concepts. Prepositional noun phrases are used widely in a multiplicity of domains to describe real world events and activities. However, one aspect that makes a prepositional noun phrase poetical is that the latter suggests a semantic relationship between concepts that does not exist in the real world. The current paper discusses how a set of rules based on Wordnet classes and an ontology repre- senting human behavior and properties, can be used to identify figures of speech. It also addresses the problem of inconsistency resulting from the assertion of figures of speech at various levels identifying the problems involved in their representation. Finally, it discusses how a contextualized approach might help to resolve this problem.

1 INTRODUCTION

Research in computational linguistic creativity has gained a renewed attention over the last decade and falls under the auspices of Artificial Intelligence, Nat- ural Language Processing and Linguistics. Poetry, as a special form of creative writing makes intense use of identifiable linguistic tools such as figures of speech [1]. Poetry, is characterized as Art partly due to itsaesthetic qualities which appeal to the human senses and due to its notional and semantic content. In [2] poetry is defined as the art form in which human lan- guage is used for its aesthetic qualities in addition to, or instead of, its notional and semantic content.

Poetic writings frequently violate the syntactical, phonological and semantic rules of natural language text [25]. However, they also possess some distinct characteristics that help to identify poetic writings and set the grounds upon which the automatic recog- nition of poetic phrases can be done. Our analysis aims to show that certain literary tools e.g. figures of speech [9] violate ontological relationships among concepts in order to cause emotional and cognitive effects. Although we focused on a very simple subset of these phrases, our ideas can be expanded to more complex phrases in the future.

As a starting point to our work, a number of sim- ple prepositional noun phrases were extracted from the file ‘blake-poems.txt’ which includes poems by William Blake from the Gutenberg’s collection [3]. Then, via the use of Wordnet [23] hypernym rela- tions, conflict relations were identified giving rise to certain types of figures of speech [1]. Results were promising since even at this primitive stage and via the implementation of a few conflict rules, we have been able to retrieve most of the prepositional noun phrases constituting figures of speech in the particu- lar file. This enabled us to argue that for a certain type of phrases it is possible to determine whether they constitute figures of speech, like for example per- sonification phrases [6].

Natarajan Meghanathan et al. (Eds) : CSTY, AI, MaVaS, SIGI, FUZZY - 2019
pp. 53-64, 2019. © CS & IT-CSCP 2019 DOI: 10.5121/csit.2019.91405

To motivate our discussion further, let us con- sider the prepositional noun phrase the tent of God appearing in the poem ‘The Little Black Boy’ by William Blake [4]. This prepositional noun phrase relates the words ‘tent’ and ‘God’ in the particular type of natural language phrase. The conceptual hierarchy of the word ‘tent’ in Word- net [23], as provided by the closure of the first synset (including only the names of synsets), is:

‘shelter.n.01’, ‘structure.n.01’, ‘artifact.n.01’ ‘whole.n.02’, ‘object.n.01’, ‘physical entity.n.01’, ‘entity.n.01’

Unfortunately, the word ‘God’ in this case does not have a hypernym hierarchy. For this reason we consider the gloss [23] of the synonym which defines God as ‘the supernatural being conceived as the perfect and omnipotent and om- niscient originator and ruler of the universe’. Another option is to take the second sense of the word, which is the synset Synset(‘deity.n.01’), and has the following concept hierarchy:

Synset(’spiritual being.n.01’), Synset(‘belief.n.01’),
Synset(‘content.n.05’), Synset(‘cognition.n.01’),
Synset(‘psychological feature.n.01’), Synset(‘abstraction.n.06’),
Synset(‘entity.n.01’)

Either of the first sense’s gloss definition of ‘God’ (Supernatural thing) or the second sense’s hypernym hierarchy show that the term ‘God’ does not have a physical referent and is an abstraction whilst the term ‘tent’ refers to a physical entity, and is an artifact. Prototyping the relevant phrase for clarity, we arrive at phrases of the form:

the ARTIFACT of a SUPERNATURAL
the ARTIFACT of a SPIRITUAL

where in place of ARTIFACT we can place any ob- ject including ‘artifact’ in its hypernym closure, and similarly in the place of SUPERNATURAL any ob- ject whose hypernym closure contains any sense of ‘supernatural’ object.

Although Wordnet [23] is useful for relating words to concepts and to concept hierarchies, each poly- semous word is described in terms of several senses (synsets) aiming to capture contextual aspects of its use (as seen in the above example). Each synset is connected to other synsets via the hypernym, hy- ponym and meronym semantic relations but not any other domain -specific semantic relations with other words. Since Wordnet aims to cover everyday En- glish [5], any specific relations need to be addressed via semantically enriched formalisms like for exam- ple, Description logics. Considering different senses is beyond the scope of the current work.

The next section discusses important work in com- putational poetry analysis and generation. There is no work to our knowledge that attempts to address the problem we are going to solve. However, an onto- logical approach can enrich the results of traditional approaches.

2. BACKGROUND

The main focus of research in computational creativ- ity over the past decade has been in the creation of models capable of generating poetry and in the classi- fication of poetry. Several models have been created for the automatic generation of poetry. However no other work to our awareness addresses the problem of automatic recognition of figure of speech [1]. In the following paragraphs we discuss some of these ap-proaches and we refer to the properties of poetic text that makes poetry generation and understanding a distinct challenging problem on its own.

2.0.1. Figure Of Speech

Before we discuss some of the most important works in this area, we need to explain the meanings of figure of speech [1] and personification [6], used extensively in this work .

Deftnition 2.1 (Figure of Speech [1]). A figure of speech is a phrase or word having different meanings than its literal meaning. It conveys meaning by iden- tifying or comparing one thing with another, which has connotation or meaning familiar to the audience. That is why it is helpful in creating a vivid rhetorical effect.

Deftnition 2.2 (Personification phrases [6]). Per- sonification phrases are a particular type of figures of speech where non-human objects are portrayed in such a way that we feel they have the ability to act like human beings.

2.0.2. Existing Work in Computational Po- etry

Among the first advocates of automatic generation of poetry is R.W. Bailey [10]. Toivanen et al. [26] introduced predicates which explicitly recorded the possible words that can exist at each line and each position in a poem combined with constraints e.g. that only one candidate word can exist in a particu- lar position in a particular line of a poem. Althought this approach is very important in that it attempts to address the problem of poetry generation by for- malizing the syntactic features characterizing poetry it suffered from the need to record explicitly all the rules and predicates about the data.

Manurung [22] proposed a poetry generation sys- tem, which, given some metrical constraints as in- put, it uses the dynamic programming technique of chart generation to efficiently construct all valid para- phrases of a natural language utterance.The chart could be used both as a transducer for the produc- tion of logical form of utterance strings and as a gen- erator from logical forms to strings [22]. The above technique addresses the issue of semantic meaning and paraphrasing of utterances via the use of a lex- icon whose semantics subsume the semantics of in- put. With the advent of the Semantic Web tools and in particular with the introduction of ontologies as a tool enabling the representation of semantic relations between concepts, it is now possible to enrich the se- mantics of the poems generated via the use of a wider range of semantic relations between the concepts in- volved.

Another interesting technique was advocated in [15]. The basic strategy adopted in this case, was to produce poetry in collaboration with the user. The task is accomplished by firstly parsing the input line in order to analyze its poetic structure and then gen- erating a new line. The output line relied on syllabifi- cation engine and a Support Vector Machine (SVM). Creative text does not conform to the normal pro- duction rules governing non-poetic natural language text, creating an even more challenging problem. Ma- nurung [21] refers to two distinct aspects of poetry generation that make it a unique and difficult prob-lem in NLP [21]:

1. The interdependent linguistic phenomena and surface constrains due to ‘unity’ of poetry. Unity in this context refers to the fact that every sin- gle linguistic decision potentially determines the success of the poem.

2. Lack of clear, well-defined communicative goal

Although poetic text does not adhere to a valid defi- nition, it satisfies the following properties:

1. Meaningfulness:The text must convey some con- ceptual message that is meaningful under some interpretation (this property actually holds for all types of text)

2. Grammaticality: A Poem must obey linguistic conventions that are prescribed by a given gram- mar and lexicon. Although this property also holds for all types of text, grammaticality in po- etry is probably less constrained than that of or- dinary texts and is governed by figurative lan- guage tropes.

3. Poeticness: A poem must exhibit poetic fea- tures such as phonetic patterns, rhythmic pat- terns and rhyme.

3. WORDNET AND THE RECOGNITION OF FIGURES OF SPEECH

Our initial data consists of poems extracted via the use of regular expressions from the nltk [20] Guten- berg file of William Blake poems. Information about poems is inserted into a dictionary so that the entry poems dictionary[k][j] holds information about the jth line of the ith poem. The dictionary includes information about the noun phrases, prepositional phrases etc. Prepositional phrases were extracted via the use of the Pattern [16] library for Python. Us- ing Wordnet [23] and some basic rules (for example the two following rules stated in Python, were used to derive figures of speech) about the categories of words, we have been able to derive a list of phrases adhering to the definition of figures of speech.

```
 defrule1(c1,c2):
    if concept_is(c1,‘artifact’) and concept_is(c2,‘abstraction’):
           return True
    elif concept_is(c1,‘abstraction’) and concept_is(c2,‘artifact’):
           return True else:
            return False

 defrule2(c1,c2):
    if concept_is(c1,‘physical entity’) and concept_is(c2,‘abstraction’):
return True_
    elif concept_is(c2,‘physi-cal entity’)and
    concept_is(c1,‘abstraction’):
         return True
    else:
         return False

 defrule3(c1,c2):
    ifconcept_is(c1,‘artifact’)
       and concept_is(c2,‘body part’):return True
    elifconcept_is(c2,‘artifact’)and concept_is(c1,‘body part’):
          return True        _
    else:
          return False
```

In the table below we include some examples of figures of speech derived by our rules. Note that the order of appearance of concepts is important:

An initial ontology is shown in figure 1. The on- tology (which we will call OR from now on) aims to provide a conceptualization of Humanly possessed qualities and observable behavior, like for example the ability to make facial expressions (represented by the object property has facial expression), which can be used to identify entities which do not, by na- ture, have these properties. Our intention is to use this part of the ontology to identify personification phrases, and other classes and properties which can be used to identify other figures of speech. Let us con- sider, for example, the class Human Gesture which includes the class Facial expressions which in turn in- cludes the object Smile. Assuming that only humans can make facial expressions in real life, the property has facial expression can be made only about an in- stance of the class Human. In order to enforce this requirement, we designate the domain of the prop-erty to be Human. Every assertion stating that an entity belonging to a disjoint class has this property, will lead to an conflicting KB when added to the on- tology.

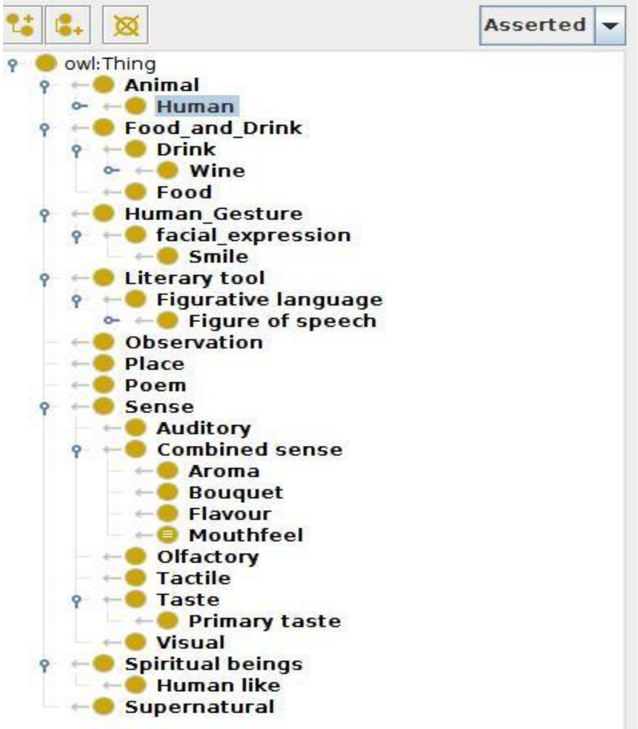
Table 1: Examples of figures of speech derived from a set of conflict-identification rules

Concept$_1$	*Concept*$_2$	**Example**
Person	Quality	*the daughter of beauty*
Person	Abstraction	*man of Liberty*
Person	Feeling	*man of woe*
Message	Location	*the secrets of the land...*
Possession	Feeling	*the land of sorrows*
Furnishing	Body part	*little curtain of flesh*
Physical Entity	Spiritual	*the vales of har*
Psychological Feature	State	*image of weakness*

In certain occasions it is not necessary to use prop- erties specific to humans since we can get the same result by abstracting away from the classes involved. For example, consider the phrase: ‘tent’ of ‘heaven’. There is no need to refer to the domain of the prop- erty ‘tentOf’, since from the closure of the hypernyms of the first synset of ‘tent’, it follows that ‘tent’ is an artifact:

```
Synset( ‘shelter.n.01’ ),
Synset( ‘structure.n.01’ ),
Synset( ‘artifact.n.01’ ),
Synset( ‘whole.n.02’ ),
Synset( ‘object.n.01’ ),
Synset( ‘physical\_entity.n.01’ ),
Synset( ‘entity.n.01’ )]
```

Continuing with the above example, the concept hy- pernyms do not provide information about disjoint- ness relations between classes, and it is not obvious whether entities like ‘tent’ are not Human. Follow-ing Wordnet [23]: a ‘tent’ is a structure that provides privacy and protection from danger’. Relations be- tween classes are in most cases implicit (e.g.the word ‘Human’ is not mentioned in the above example). To record all possible relationships between concepts ex- plicitly would not be feasible. However, abstracting away enables us to capture a wide range of conflicting relations using fewer classes.



- owl:Thing
  - Animal
    - Human
  - Food_and_Drink
    - Drink
      - Wine
    - Food
  - Human_Gesture
    - facial_expression
      - Smile
  - Literary tool
    - Figurative language
      - Figure of speech
  - Observation
  - Place
  - Poem
  - Sense
    - Auditory
    - Combined sense
      - Aroma
      - Bouquet
      - Flavour
      - Mouthfeel
    - Olfactory
    - Tactile
    - Taste
      - Primary taste
    - Visual
  - Spiritual beings
    - Human like
  - Supernatural

Figure 1: preliminary ontology

To motivate our discussion further, lets consider the prepositional noun phrase: ‘smiles of heaven’. If Heaven is included as an individual of the class Su- pernatural in OR, which is disjoint with class Human, and the domain of property has facial expression is Human, then asserting that ‘Heaven’ has property has facial expression will end up in a conflicting KB. Nonetheless, this inconsistency (please refer to the notion of inconsistency below) leads to the identifica- tion of personification statement.

The problem, then is how to address this incon- sistency. Although it is not within the scope of this paper to provide a final solution to this problem, we discuss possible ways of handling inconsistency in the following subsections. In order to do that we need to define the Syntax and Semantics of terminological knowledge bases.

3.1. Basic Syntax And Semantics Of Terminological Knowledge Bases

DL based formalisms, like OWL DL, are a family of class- based knowledge representation formalisms equipped with well-defined model-theoretic semantics [17]. In order to discuss conflicts with ontological knowledge we firstly need to refer to the definition of an ontology, and the notions of interpretation and satisfiability.

3.1.1. Ontology

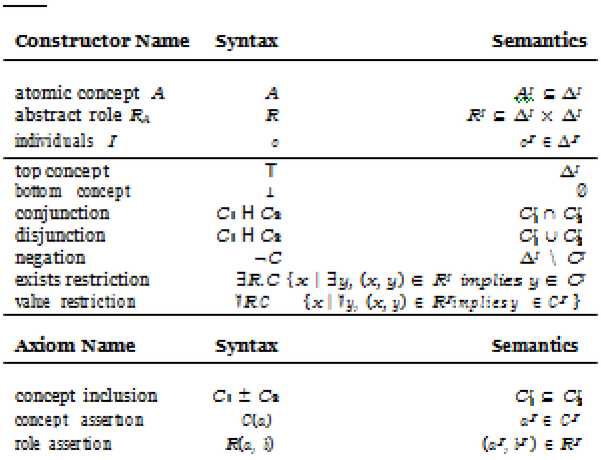
An ontology in this paper is described as a structure
, where denotes a DL TBox (a set of ter- minological axioms) and denotes a DL ABox (a set of grounded assertions) An interpretation I of an ontology O , consists of a domain ΔI and an interpretation function .I such that the relations in Table2 are satisfied. Note that the axioms refer- ring to the domain and range of properties take their usual meaning and are neglected due to limitation of space.

The notion of satisfiability is closely related to the notion of consistency. A named concept C in an on- tology O is satisfiable iff there is an interpretation I such that CI ∅

Deftnition 3.1 (Inconsistency). An ontology is in- consistent iff it has no interpretation.
Examples of inconsistent formulas (where the usual meanings of disjointness and domain apply) are:

- {A(b), A ± B, A ± ¬B},
- {C(a), ¬C(a)}, {A(a), B(a), Disjoint(A, B)}
- {Domain(R, A), Disjoint(A, B), R(a, b), B(a)}

Table 2: Syntax and Semantics of Basic DL



Constructor Name	Syntax	Semantics
atomic concept A	A	$A^I \subseteq \Delta^I$
abstract role R_A	R	$R^I \subseteq \Delta^I \times \Delta^I$
individuals I	o	$o^I \in \Delta^I$
top concept	T	Δ^I
bottom concept	⊥	∅
conjunction	C_1 H C_2	$C_1^I \cap C_2^I$
disjunction	C_1 H C_2	$C_1^I \cup C_2^I$
negation	$\neg C$	$\Delta^I \setminus C^I$
exists restriction	$\exists R.C$	$\{x \mid \exists y, (x, y) \in R^I \text{ implies } y \in C^I$
value restriction	$\forall R.C$	$\{x \mid \forall y, (x, y) \in R^I \text{implies} y \in C^I\}$
Axiom Name	**Syntax**	**Semantics**
concept inclusion	$C_1 \pm C_2$	$C_1^I \subseteq C_2^I$
concept assertion	$C(a)$	$a^I \in C^I$
role assertion	$R(a, b)$	$(a^I, b^I) \in R^I$

Following the syntax and semantics of OWL DL, we could list a number of similar inconsistency con- ditions.

3.2. Practical Considerations about Inconsistency and Representation

As stated already, the figures of speech cannot be added as assertions in the A-Box of an ontology like OR since they will lead to inconsistency. In this sub- section we discuss how the existing tools and for- malisms can be used to represent figures of speech. Although we are currently talking about the simplest form of figure of speech, the approaches considered may be extended to handle more complex phrases.

3.2.1. Reiftcation

One way to overcome the problem of inconsistency, is to add figures of speech into the above ontol- ogy as a particular class of (non-factual) statements, Cfigures of speech, about which information is kept. This leads to the idea of reification, which is sup- ported by the RDF syntax. RDF [8] supported the reification of statements via a special vocabulary [13] in order to represent information about triples. An example of a reified figure of speech is included below:

```
_ex1: rdf:type rdf:Statement;
rdf:subject      #Heaven;
rdf:predicate #has_facial_expression;
 rdf:object      #Smile;
```

Information about ex1 can be added as follows:

_ex1: exuri:has_author #Whiltman;
_ex1: exuri:appears_in #PoemId

In RDF, the subject of reification is intended to refer to a concrete realization of an RDF triple, such as a document or surface syntax, rather than a triple considered as an abstract object [27].

Other forms of reification include N-ary relations [19], singleton properties [24], and Named Graphs[14]. Each of these approaches aims to solve a differ- ent problem. For example, N-ary relations [19] enable more than two individuals to participate in N-ary re- lations [19]), Named Graphs [14] enable the addition of provenance and trust information to web resources, and Singleton Properties [24] enable the creation of properties for a single statement. Different N-ary re- lation patterns are discussed in [19]. Each relation pattern uses a class to represent a relationship and n new properties to represent the association of each participating entity to the relation. This approach enables the addition of information about the en- tities participating to the relationship that a triple cannot express on its own. All of the above methods suffer from maintenance problems and the increased complexity caused by the number of new constructs created.

The extend to which contextualization needs to be formalized depends on the reasoning capabilities needed for the representation of figures of speech. An attempt to formalize contextualization of resources is provided in a separate paragraph below.

3.2.2. Programming Tools

For the simple relations characterizing the figures of speech currently under investigation, conflicting fig- ures of speech are added to a separate graph in a Dataset using the RDFLib [11]. Figures of speech are identified firstly in poems when they violate the constraints (disjointness, domain etc) of the ontology suggested.

The RDFLib package [11] enables us to create Datasets which may containing named graphs and contexts, which are sets of triples sharing the same URI. Datasets can be queried using SPARQL. This approach is not meant to be a fully-fledged contex- tual approach for local reasoning. The importance of a more contextualized approach remains to be inves- tigated

3.2.3. Algorithmic Approach

One way to create a figures of speech ontology algo- rithmically, $O_{\text{figures of speech}}$ say, is to create it pro- gressively by adding to it the figures of speech contra- dicting with the original ontology, updating at each addition the domain of the properties asserted so that it includes the domain of the subject of the relevant figure of speech. Since the figures of speech asserted violate the constraints of the real life ontology, the do- main of each property asserted to $O_{\text{figures of speech}}$ should be added to a local definition of the property in $O_{\text{figures of speech}}$. This implies that the properties would need to have different extensions in each on- tology, which leads to the notion of contextualization analyzed briefly below. Notably, it is not obvious yet whether we want to adopt this approach since we haven't assigned any terminological axioms w.r.t figures of speech yet. Our discussion in the follow-ing paragraph aims to analyze the complexity of the problem further.

3.2.4. Contextualizing Ontologies - An Ab-Stract Representation

n this section we borrow concepts from the work done in contextualized ontologies in [12] in order to derive the basic conceptualization of a context-based representation of our domain.
Before proceeding fur- ther we need to redefine Ontologies taking into con- sideration context.

A contextualized representation of ontologies adopts the principles of locality and compatibility un- derpinning the local model semantics [18]. The prin- ciple of locality states that reasoning requires only part of what is potentially available and the princi- ple of compatibility states that there is compatibility among the kinds of reasoning performed in different contexts [18]. In a context-based ontology approach, each ontology is indexed e.g. by an index i and an ontology O_i defines a language L_i. Every expression that appears either with an index i or no index is assumed to be in the language defined by O_i.

Deftnition 3.2. Le I be a set of indices, L be the disjoint union of C, R, and O, the set of strings denot- ing concepts, roles and individuals, respectively. An OWL ontology with index i is a pair (i, O_i) where $i \in I$ and $O_i \in (T_i, A_i)$ where T and A are a T-box and A-box respectively in $L \cup (I \times L)$ [12].

Since we are modeling a domain consisting of con- flicting ontologies, then the space of ontologies needs to be appropriately modeled, taking into considera- tion compatibility issues. Informally an OWL space is a set of ontologies appropriately indexed. Fol- lowing [12], an OWL space is a family of ontolo- gies $\{(i, O_i)\}$ i∈I such that every O_i is an ontology, and for each i *f*= j, the j-foreign language of O_i is contained in the local language of O_j. Let the OWL space of our scenario consist of three ontologies: O_H , O_R, Ofigures of speech . In addition, let C_i, R_i and O_i be the sets of strings denoting the concepts, roles and individuals of each ontology O_i respectively, and let $C \equiv C_i$, $R \equiv R_i$, $O \equiv O_i$.

Borrowing ideas from [12], we define a Simple Poetic Space to be the OWL space: $\{(URI_H , O_H), (URI_R, O_R), (URI_L, O_L)\}$, where $I_{i\in}\{URI_H ,URI_R ,URI_L\} = (\Delta^{Ii} , (.)^{Ii})$. For each prop-erty $i : p \in R_i$ for $i \in \{URI_R, URI_L\}$ of the ontologies O_R and $O_{\text{figures of speech}}$, O_H includes a property $URI_H : p$ such that $i : p$ is a subproperty of $URI_H : p$ in R_H . A localized interpretation for an OWL space is now defined as follows:

Deftnition 3.3 (OWL localized interpretation [12]). An OWL interpretation with local domains for the OWL space $(i, O_i)_{i\in I}$, is a family I = {Ii}i∈I , where each $I_i = (\Delta^{Ii} , (.)^{Ii})$, called the local interpretation of O_i, is an interpretation of L_i.

The above definition deviates slightly from the def- inition of OWL interpretation with local domains in [12] since our domain is much simpler and there is no need to include holes [12]. Instead of holes we use an Ontology (O_H above) so that the properties of all other ontologies in the OWL space entail the proper- ties of this ontology.

In order to make sure that any expression asserted in an ontology O_j is interpretable in an ontology O_i with a possibly different domain, we assume that the following conditions also hold [12]:

1. $(j : C)^{Ii} = C^{Ij} \cap \Delta^{Ii}$,
2. $(j : r)^{Ii} = (r)^{Ij} \cap (\Delta^{Ii} \times \Delta^{Ii})$
3. $(j : a)^{Ii} = (a)^{Ij}$

Finally, reasoning with resources from multiple contexts is achieved within the scope of a con- text space, equipped with a set of mapping rules (bridge rules). In [12], a context space is defined as pair $(\{(i, O_i)\}_{i\in I}, \{M_{ij}\}_{i,j\in I})$ where $\{(i, O_i)\}$i∈I is an OWL space and $M_{ij\ i,j\in I}$ a family of mappings from i to j for each pair i, j I Our conceptualization may be considered as a special case of a context space where the properties of O_R and $O_{\text{figures of speech}}$ are subsumed by the properties of O_H . As a result, if an assertion about a property is entailed by O_R or $O_{\text{figures of speech}}$, it is also entailed by O_H .

Example Consider the OWL space:
$\{(R, O_R), (L, O_L), (H, O_H)\}$ such that: $O_R \prec_P O_H$ and $O_L \prec_P O_H$ Then,

1. $I_R \models H \pm \neg S$, $I_R \models S(a)$,
   $I_R \models \forall\ \text{has}\,f\text{expr.SMILE} \pm H$

2. $I_L \models R : \text{has}\,f\text{expr}(R : a, s_1)$
   $I_L \models L : \text{has}\,f\text{expr}(R : a, s_1)$

3. $I_H \models \text{has}\,f\text{expr}.R : \text{SMILE}$

4. $\{R : \text{has}\,f\ \text{expr} \xrightarrow{\pm} H : \text{has}\,f\ \text{expr},$
   $L : \text{has}\,f\ \text{expr} \xrightarrow{\pm} H : \text{has}\,f\ \text{expr}\}$

4. CONCLUSION AND FUTURE WORK

The current paper sets the foundations for the recog- nition and identification of figures of speech by fol- lowing a semi-manual ontological approach. With the use of a set of some basic rules concerning Wordnet classes of entities and terminological knowledge de- rived from a real world ontology concerning the con- traints on properties, we have been able to identify a set of prepositional noun phrases constituting figures of speech. The results, are encouraging suggesting that some poetic phrases used as literature tools can be recognized and analyzed due to their contradic- tory nature when compared to a real domain ontol- ogy. In particular, the use of an ontology describing human features can help (via the use of constraints, e.g. disjointness, domain and range) to identify con- flicting assertions made by personification phrases. More complex phrases will be investigated in the fu- ture.

The paper also considered the challenges involved in the representation of figures of speech due to the inconsistency caused when added to the A-Box of the particular real-domain ontology employed in order to conceptualize observations regarding Human behav- ior and properties. We have referred to various ap- proaches in order to evaluate their merits and have also considered the application of contextualized on- tologies for the solution of the inconsistency problem.

In the future we intend to focus on the automatic extraction of a wide variety of figures of speech from poetic texts and the development of an appropriate formalism for their representation.

REFERENCES

[1] https://literarydevices.net/figure-of-speech.

[2] http://www.poetry.org/whatis.htm.

[3] http://www.gutenberg.com/.

[4] http://www.poetryfoundation.org.

[5] https://en.wikipedia.org/wiki/WordNet.

[6] https://literarydevices.net/personification.

[7] https://litcharts.com/literary-devices-and- terms/figurative-language.

[8] Rdf 1.1 semantics, 2014. https://www.w3.org/TR/rdf11-mt/.

[9] M.H Abrams and Geoffrey G. Harpham. A Glos- sary of Literal terms. Thomson Wadsworth, 1999.

[10] R.W. Bailey. Computer assisted poetry: The writing machine is for everybody. In J. L. Mitchell, editor, Computers in Humanities, pages 283–295. Cambridge University Press, 1974.

[11] Carl Boettiger. rdflib: A high level wrapper around the redland package for common rdf publications, 2018.

[12] Paolo Bouquet, Fausto Giunchiglia, Frank van Harmelen, Luciano Serafini, and Heiner Stuckenschmidt. C-OWL:contextualizing ontologies. In K. Sekara and J. Mylopoylos, editors, Proceed- ings of the Second International Semantic Web Conference, number 2870 in Lecture Notes in Artificial Intelligence, pages 164–179. Springer Verlag, 2003.

[13] Dan Brickley and R.V. Guha. RDF Vo- cabulary Description Language 1.0:RDF Schema. W3C Recommendation, World Wide Web Consortium, February 2004. http://www.w3.ord/TR/2004/REC-rdf- schema-20040210.

[14] Jeremy J. Carrol, Christian Bizer, Pat Hayes, and Patrick Stickler. Named graphs, prove- nance and trust. In Proceedings of the 14th International Conference on World Wide Web, WWW'05, pages 613–622, New York, NY, USA, 2005. ACM.

[15] Amitava Das and Bj¨orn Gamb¨ack. Poetic ma- chine: Computtional creativity for automatic poetry generation in bengali. In ICCC, 2014.

[16] T. De Smedt and W Daelemans. Pattern for python. Journal of Machine Learning Research, 13:2063–2067, June 2012.

[17] G. Flouris, Z. Huang, Jeff Pan, D. Plexoudakis, and H. Wache. Inconsistencies, negations and changes in ontologies. In Proceedings of AAAI Conference on Artificial Intelligence (18), January 2006.

[18] Chiara Ghidini and Fausto Giunchiglia. Local models semantics, or contextual reasoning = lo- cality + compatibility. 127(2):221–259, 2001.

[19] Pat Hayes and Chris Welty. Defining N- ary relations on the semantic web, 2006. https://www.w3.org/TR/swbp-n-aryRelations.

[20] Edward Loper and Steven Bird. Nltk: The natu- ral language toolkit. In Proceedings of the ACL-20 Workshop on Effective Tools and Method- ologies for Teaching Natural Language Process- ing and Computational Linguistics, volume 1 of ETMTNLP '02, pages 63–70, Stroudsburg, PA, USA, 2002. Association for Computational Lin- guistics.

[21] H. Manurung. An evolutionary algorithm aproach to poetry generation. Phd. thesis, Uni- versity of Edinburgh, 2004.

[22] H.M. Manurung. Chart generation of rythm- patterned text. In Proceedings of the First In- ternational Workshop on Literature in Cognition and Computers, 1999.

[23] George A. Miller. Wordnet: A lexical database for english. Communications of the ACM, 38(11), 1995.

[24] Vinh Nguyen, Oliver Bodenreider, and Amit Sheth. Don’t like rdf reification? making state- ments about statements using singleton prop- erty. In International World Wide Web Con- ference, pages 759–770, April 2014.

[25] Hugo Concalo Oliveira. Automatic generation of poetry:an overview. Technical report, CISUC, DEI, University of Coimbra, 2009.

[26] J.M. Toivanen, H. Toivonen, and A. Valitutti. Harnessing constraint programming for poetry composition. In In Proceedings of the Fourth In- ternational Conference on Computational Cre- ativity, pages 160–167, 2013.

[27] David Wood, Markus Lanthaler, and Richard Cyganiak. RDF 1.1 concepts and abstract syntax. W3C Recommenda- tion, World Wide Web Consortium, 2014. http://www.w3.ord/TR/2014/REC-rdf11- concepts-20140225.