

A Linguistic Light Approach to Multilingualism in Lexical Layers for Ontologies

Alexander Trousov, John Judge, Mikhail Sogrin, Amine Akrouit

IBM

Email: {atrosso, johnjudge, sogrimik, amine_akrouit}@ie.ibm.com

Brian Davis, Siegfried Handschuh

DERI

Email: {brian.davis, siegfried.handschuh}@deri.org

Abstract—Semantic web ontologies are being increasingly used in modern text analytics applications and Ontology-Based Information Extraction(OBIE) as a means to provide a semantic backbone either for modelling the internal conceptual data structures of the Text Analytics(TA) engine or to model the Knowledge base to drive the analysis of unstructured information in raw text and subsequent Knowledge acquisition and population. Creating and targeting Language Resources(LR)s from a TA to an Ontology can be time consuming and costly. In [1] the authors describe a user-friendly method for Ontology engineers to augment an ontologies with a lexical layer which provides a flexible framework to identify term mentions of ontology concepts in raw text. In this paper we explore multilinguality in these lexical layers using the same framework. We discuss a number of potential issues for the “linguistic light” Lexical Extensions for Ontologies (LEON) approach when looking at languages more morphologically rich and which have more complex linguistic constraints than English. We show how the LEON approach can cope with these phenomena once the morphological normaliser used in the lexical analysis process is able to generalise sufficiently well for the language concerned.

I. INTRODUCTION

SEMANTIC web ontologies are being increasingly used in modern text analytics applications and Ontology-Based Information Extraction(OBIE) as a means to provide a semantic backbone either for modelling the internal conceptual data structures of the Text Analytics(TA) engine or to model the Knowledge base to drive the analysis of unstructured information in raw text and subsequent Knowledge acquisition and population. Creating and targeting Language Resources(LR)s from a TA to an Ontology can be time consuming and costly. A Language Engineer working with a TA system must typically manually align existing internal linguistic resources with a new Ontology or create new LR’s to support a domain shift. If the creation of LR’s for an TA system is integrated into the Ontology engineering process via user friendly Ontology lexicalisation for non-linguists. A lexical layer, which describes the various lexical realisations of Ontological term facilitates such a process. The “linguistic light” approach described in [1] outlines such a lightweight lexical layer which can be easily implemented into an existing ontology. The lexical layer (LEON)—(Lexical Extensions for Ontologies) can be subsequently traversed and compiled into internal LR’s of the TA engine. Additionally, Organisations working in multilingual

environments creates a demand for multilingual Ontologies [2]¹ The authors also claim that their approach can be retargetted to a new domain or language by simply providing the appropriate lexical information. The cross language portability of this approach and associated issues will be presented in this paper. Portability across languages is an important characteristic for an approach to lexical layers because of the cost and effort involved in redeveloping an ontology for a new language. One of the main principles behind the semantic web is the ability to easily exchange and utilise semantic information so by having a unified approach to identifying occurrences of Ontological terms in text across a number of languages we can maintain this inter-operability by using the same ontology.

This rest of this paper is structured as follows: Section II discusses related work, Section III gives an overview of LEON type lexical layers, Section IV describes how LEON can provide a multilingual lexical layer and highlights some potentially problematic features of languages besides English, Section VI explains how the LEON approach copes with these phenomena and discusses the implications of false positives, finally, Section VII concludes.

II. RELATED WORK

The inclusion of a linguistic or lexical layer into an Ontology or Ontology lexicalization is by no means a new phenomenon. For example, LingInfo, was developed as part of the SmartWeb³ project[3]. The work conceptualized the idea of a linguistic layer for a Semantic Web Ontology or more specifically a “multilingual/multimedia lexicon model for Ontologies” [3]. Linguistic representation in LingInfo can consist of: a Language identifier, POS (Part of Speech) tag, morphological data, and syntactic compositional data as well a contextual data in the form of grammar rules of N-grams. Furthermore, content and knowledge are organized into four layers, where the Ontology layer is located at the central layer and linguistic features and their subsequent associations to the central layer are located in the outer middle layers with the outer layer containing textual content. The Ling-Info model is applied to the SmartWeb Integrated Ontology

¹Although against good Ontology Engineering practice, a substantial amount of Ontologies on the Web are in English which forces the need for localising knowledge. One can observe this easily by accessing such tools as OntoSelect²

³<http://smartweb.dfki.de/>

SWIntO, whereby the linguistic feature layer is compiled into Language Resources (gazetteers) within the SProuT IE engine based on a mapping between the SWIntO and SProuT's TDL Type Description Languages. This mapping is applied to both SWIntO concepts and properties. The work of [3] is influenced strongly by LMF Lexical Markup Framework,[4], which is part of the ISO TC37/SC4⁴ working group on the management of Language Resources. LMF has its origins in Language Engineering standardization initiatives such as EAGLES⁵ and ISLE⁶. LingInfo also caters for multilingual Ontology lexicalisation, but we argue that the LingInfo model is too complex for use but non-linguistic engineers where LEON attempts to shield the Knowledge Engineer from complex linguistic formalisms.

Ontology lexicalisation is closely related to work within Linguistic Ontologies. Linguistic Ontologies are used to describe semantic constructs rather than to model a specific domain and they are typically characterised by being bound to the semantics of grammatical or linguistic units i.e. GUM and SENSUS [5]. Ontologies such as Wordnet [6] and EuroWordnet [7] however are concerned with word meaning. Certain linguistic Ontologies are language independent such as EuroWordnet while the majority are not. EuroWordnet is a multilingual database containing wordnets for several European languages [8]. Each language specific word net is similarly structured to the English WorldNet and are linked via an Interlingua index. Consequently, one can access the translation of similar words in a target language for a given word within the source language. Linguistic Ontologies are primarily descriptive though they are frequently exploited by NLP systems either directly or to bootstrap the creation of new Language Resources. LEON on the other hand is designed explicitly to support the text analytics (or IE) task by replacing the manual retargeting of multilingual LRs within an IE system to an Ontology either (semi-)automatically.

Ontology Localization is also a closely related field to that of Ontology lexicalization. "Ontology Localization consists of adapting an Ontology to a concrete language and cultural community"[2]. In [2] the authors describe LabelTranslator, an Ontology localization tool which automatically translates ontological term labels (rdfs:labels of classes, instances and properties) in a source language to their target language equivalent. The system caters for English, German and Spanish. LabelTranslator attempts to best the most approximate translation by accessing translation services such as Babelfish and Free-Translation, in addition to various Language Resources such as EuroWordnet [7], Wikitionary and GoogleTranslate. A ranking method based on the Normalized Google Distance(NGD) [9] is also applied to propose the most approximate target translation label from collection of suggested translations by taking into account the similarity of the source language label's lexical and semantic context. The LEON approach is

tailored towards an IE task which is very different from that of localisation, since as already shown in [1], rdfs:labels is a form of Ontology lexicalisation are too simplistic to capture the linguistic idiosyncrasies of certain surface forms as is the case with Multi Word Expressions.

Finally, we note other OBIE systems such as GATE⁷ which can be deployed as a multilingual OBIE platform [10], however LRs in GATE must be manually aligned to the Ontology, while the LEON approach attempt to subsume part of the Dictionary creation process within the Ontology Engineering process.

III. LEON

A lexical layer which describes the lexical realisations corresponding to concepts encoded in an ontology provides an interface between the ontology and text processing applications which seek to exploit the semantics encoded in the ontology.

The number and type of lexical expressions which correspond to a particular semantic entity varies from concept to concept, however, often they occur in a form which is different from the citation form because of inflectional or grammatical needs imposed by the language. These lexical realisations are often complex and appear as multiple word units, which in turn are not always fixed expressions and can vary depending on the context.

It would appear that an adequate approach to provide such a lexical layer requires some level of linguistic knowledge to be encoded alongside the semantics. This approach however becomes somewhat untenable in practice as there are many different linguistic theories to choose from which can lead to incompatibilities between ontologies, not all linguistic theories can be implemented effectively, and the knowledge engineers who work with modern ontologies usually have little or no linguistic background.

To address these issues surrounding lexical layers [1] propose a "linguistic light" approach to lexical layers for ontologies called LEON. The LEON approach proposes that the lexical layer for an ontology consists of a tuple of the form

(CitationForm, Constraints)

for each semantic entity with a lexical realisation encoded in the ontology. The first element of the tuple, the citation form, is the basic form of the lexical realisation. The second element of the tuple is a set of constraints which specifies if and how the citation form can vary. This facilitates linguistic phenomena such as inflection and derivation as well as allowing the modelling of multi-word units which vary in both their surface form and word order using this simple approach. This approach does not focus on the linguistic description of vocabulary associated with a concept but on the linguistic features of a given concept in order to identify class instances in text. This allows for when one concept might have

⁴<http://www.tc37sc4.org>

⁵<http://www.ilc.cnr.it/EAGLES96/browse.html>

⁶<http://www.mpi.nl/ISLE/>

⁷General Architecture for Text Engineering

several different lexical realisations with different linguistic descriptions for example:

- New York, Big Apple, NY
- Rosetta Stone, Stone of Rosetta
- International Business Machines, IBM, Big Blue

In addition, it is deliberately less complex in order to cater for users, in particular knowledge engineers, who lack a linguistic background, but may wish to develop an ontology with linguistic features included. We note that the interlingua here is an the ontology. Therefore the design and conceptualisation used in the ontology could be a limiting factor where there are semantic divergences between languages or domain terminology.

IV. LINGUISTIC LIGHT MULTILINGUALISM

The “linguistic light” paradigm for lexical layers is flexible and can be applied multi-lingually with little effort. This is because the extensions are not tied to any particular language or formalism.

A. Extending LEON’s lexical layer

In order to expand the LEON lexical layer description to cope with another language we must provide an appropriate citation form and set of constraints for the lexical realisation(s) of that concept in the new language. This second tuple can then be merged with the existing data giving rise to a tuple consisting of a set of citation forms and a set of constraint sets corresponding to each citation form.

$$\left\langle \left\{ \begin{array}{l} CitationForm_{EN} \\ CitationForm_{FR} \\ CitationForm_{DE} \\ CitationForm_{...} \end{array} \right\}, \left\{ \begin{array}{l} \{cnstr1_{EN}, cnstr2_{EN}, \dots\} \\ \{cnstr1_{FR}, cnstr2_{FR}, \dots\} \\ \{cnstr1_{DE}, cnstr2_{DE}, \dots\} \\ \{cnstr1_{...}, cnstr2_{...}, \dots\} \end{array} \right\} \right\rangle$$

The multilingual lexical layer can then be used to easily retarget the ontology to a given language or locale by using the appropriate citation forms and constraint sets.

B. Signature Detection

Effective use of the linguistic light LEON lexical layer in text analytics and ontology-based information extraction applications relies on unstructured text being processed and the “signature” of a term mention being detected. The lexical analyser used to process the text needs to have some means of normalising variant forms to a common stem or lemma in order to be able to put forward potential signature tokens.

To ensure high recall, normalisation of constituents is important, especially for languages with more a complex morphology than English. In this paper we pay more attention to the normaliser as a component of a linguistic light solution. Given proper normalisation, we believe that the LLA/LLS approach will provide very high recall in a multilingual environment.

V. “LINGUISTIC LIGHT” NORMALISATION

A. Character Normalisation

This type of normalization accounts for typographic variances like using capitalisation and diacritics in Latin and Cyrillic based scripts (“Böblingen” vs. “Boeblingen”), the use of different scripts in Japanese texts, auxiliary usage of vowels in Arabic or Hebrew; regular spelling variations (British “colour” vs. American “color”). Some types of character normalisation might be efficiently performed by algorithmic methods.

B. Morphological Normalisation

Morphology is the subfield of linguistics that studies the internal structure of words. In linguistics, two types of morphological normalization are traditionally referred to, namely lemmatization and stemming. Lemmatization accounts for inflectional variants of the same word where part of speech is preserved. For example different cases, genders, numbers (like singular form of noun *database* and its plural form *databases*).

Stemming frequently involves a more “aggressive” normalization, which accounts for both inflectional and derivational morphology, where related words are mapped onto the same index, even if they have different parts of speech. For example, one can map the words *computerization*, *computerize*, *computer*, *computing*, *compute* onto the same index. An index term can be a non-word like *comput* (a minimal and hopefully unambiguous denotation of all related terms). Stemming therefore has the effect of “conflating” the index more aggressively than lemmatization, by mapping a wider set of word forms to a single index term, thereby resulting in higher recall i.e. in any query term finding more documents during search.

C. Synonym Normalisation

At least for some domains, if not for language in general, it might be reasonable to consider some words as exact synonyms and map them into the same index (for example, liver/hepatic, renal/kidney). Dictionaries of linguistic synonyms are not frequently used in indexing because linguistic synonyms are typically not exact synonyms (for example, using the chain of synonyms in MS-Word: average \approx mean \approx nasty \approx shameful one can wrongly equate average with shameful). The quality of IR and IE (depending on the task) is characterized by two intrinsic metrics: recall (the ratio of the number of relevant documents returned to the total number of relevant documents in the collection of documents indexed) and precision (the ratio of the number of relevant documents retrieved to the total number of documents retrieved). Search engines typically trade off precision for recall. In the absence of accurate relevancy ranking algorithms the user is left to sort through extensive lists of documents for the correct information. So the challenge is to achieve high precision without significantly reducing recall.

Word normalization is essential for the quality of IR systems. Research to date indicates that some character normalization is indispensable to improve recall. Morphological normalization in general improves recall, but may degrade precision. Although stemmers are widely used by the majority

of IR systems, their role for IR is frequently disputed; however, it is generally accepted that morphological normalization is indispensable for highly inflected languages (like Finnish) when the same word might have dozens of forms. It is also needed for languages with frugal morphology (like English) in the scenarios where most of the analysed documents are short. For example, if the document about databases is rather long, one might expect that the term database will be encountered in both grammatical forms: data-base and databases, and one can afford not to map both forms into the same index because the document will be retrieved as relevant to the query containing the search item database anyway. However, if the document is short, it might happen that the document will contain only mentions of plural form, in which case the document will be missed. For some time Google did not use stemming in order “to provide the most accurate results to its users”. However, Google subsequently introduced stemming technology into its system “Thus, when appropriate, it will search not only for your search terms, but also for words that are similar to some or all of those terms. If you search for “pet lemur dietary needs”, Google will also search for “pet lemur diet needs”, and other related variations of your terms⁸.”

D. Reversed Finite State Normalisation

Following [11], which is based on the work of [12], a normaliser can be built from the lexicon by combining common suffixes in a finite state automaton. A finite state automaton (FSA) is a computational model made up of states and transitions. Given an input sequence (e.g. a word as a sequence of letters), the FSA moves through a series of states according to transitions that given a current state match the current input symbol (letter). In Figure 1 there are a number of possible input sequences that reach the final state e.g. smart, start etc. The final state can be associated with information about the sequence that leads to it, such as an algorithm that produces a normal form.

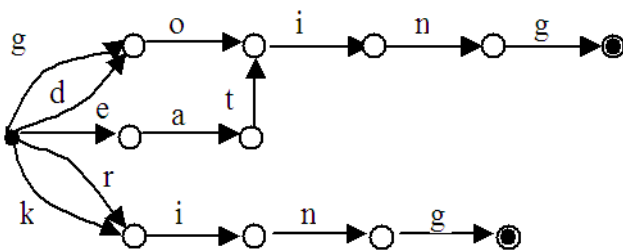


Fig. 1. Finite State Automaton

A reversed finite state normaliser is a finite state automaton which traverses the input string in reverse character order. A reversed FSA can be compiled from a full form word list, electronic dictionary or similar resource for the language or domain concerned. The resulting FSA will be such that

morphological suffixes are conflated into common paths of transitions leaving word stems following branching states.

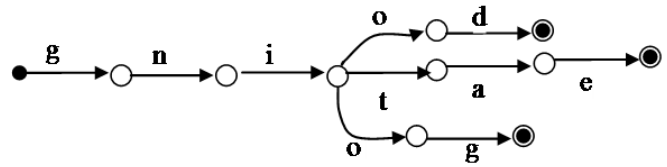


Fig. 2. Reversed Finite State Automaton

Notice that by exploiting common endings in this way the size and complexity of the FSA is reduced. This computational approach to building a normaliser does not necessarily produce proper root form lemmas for the input, instead a reduced stem is produced. These stems can often be non-word tokens but they will correspond to the orthographical root of one or more full form of the actual word it represents. These stems can then be used by a “Linguistic Light Scanner” (described in [1]) to increase recall in the identification of term mentions in text. The LEON constraints for a given citation form then determine which (if any) variants are permissible for a valid recognition.

By combining LEON and reverse FSA normalisation these two linguistically light, but computationally efficient models for lexical analysis no precision is lost because the stemming process reduces full forms to concise stems while the LEON constraints then allow or disallow inflected (or otherwise orthographically different) forms. This makes the process of adding a new language to a lexical layer relatively simple and quick to implement without any significant linguistic knowledge about the language, all that is necessary is a word list. For these reasons we suggest this type of approach to normalisation and signature detection. This approach can also deal with character normalisation where adding all variations into a full form lexicon would become unwieldy

VI. MULTILINGUAL ISSUES

When dealing with identifying term mentions in multiple languages the compatibility of the lexical description with features of the various languages is an important consideration. In the previous work ([1]) the only language considered is English, which is relatively frugal with respect to morphology, casing, and agreement when compared with other languages. Other languages also have different constraints on sentential word order which can be important to detection. We will look at some examples of how these aspects of language can be problematic and how they are handled in the linguistic light paradigm.

1) *Agreement*: Many languages require that, for example, adjectives and nouns agree with respect to number, gender, case etc. So, for instance, a singular noun can only have a singular adjective used to describe it. These constraints are important regarding the grammaticality and correctness of the language. This type of constraint is not enforced in the LEON

⁸Taken from <http://www.google.com/help/basics.html>

model. However, as the following example shows, it is often beneficial not to enforce such linguistic constraints as to do so would affect recall where there has been a human error, or a deliberate mistake owing to creative licence.

Take the French term “Intelligence Artificielle,” in this example, the gender and number agreement of the two tokens is obligatory. If it occurs with a disagreement, then it is likely a human typing mistake like “intelligence artificiel.” In this case, the disagreement is a typing error, as there is a disagreement between the noun (“intelligence”: singular feminine) and the adjective (“artificiel”: singular masculine). This can occur in texts, and it will be detected if the exact string match is turned off (to allow inflected variants of the citation form). However it would be missed if the agreement constraint were to be strictly enforced. This also allows the detection of instances in other contexts. For example, “vie et intelligence artificielles”, where “artificielles” disagrees in gender and number because here it refers to two entities which are “vie” and “intelligence.”

Likewise in Russian gender agreement is a present and important feature for grammatical correctness, however if we take the term “sistemnuj administrator” (system administrator) where both terms are in the masculine, and change one to feminine like

systemnaja	administrator
Adj Masc	Noun Masc

This ungrammatical noun phrase yields a single hit in a search on Google’s index.⁹ The text in which the example was found is using the gender disagreement as a subtle device to highlight that the person in question is woman and draw attention to this fact.

2) *Word Order*: Some languages have a less rigid restriction on sentential word order than others, German for example has quite strict rules regarding word order, Russian on the other hand is less so. This needs to be considered with regards detecting MWU lexical realisations in text analysis. A language with a freer word order means there are more possible ways of constructing a sentence which refers to a given concept. Therefore, in theory, the search space is larger and correspondingly so is the likelihood of detecting false positives.

Consider the French MWU “Maladies Sexuellement Transmissibles” (sexually transmitted diseases). If we encounter the same words in varying order and forms like “maladie mortelle sexuellement transmissible,” and “Maladie transmise sexuellement” the underlying concept which is being referred to remains the same. So a language with a freer word order is not necessarily a problem for MWU lexical realisations.

VII. CONCLUSIONS

We have examined a number of linguistic considerations for ontology lexicalisation across multiple languages. We have also discussed the LEON “linguistic light” approach to adding a lexical layer and shown how it is robust enough to handle various linguistic nuances without having to explicitly encode linguistic information. The caveat, however, is that in order to detect the linguist “signatures” of term mentions in

text the LEON approach needs some suitable normalisation of the input text.

Following in the linguistic light vein we have shown how a simple, robust normaliser can be induced from a wordlist in the form of a reverse finite state automaton. Once the lexical layer for an ontology has been implemented, the appropriate wordlist already exists in the form of the citation form lexical realisations encoded in the lexicon, so a reverse FSA normaliser can be rapidly produced for the appropriate vocabulary. By combining these two linguistic light approaches to analysing natural language in text an ontology can be rapidly retargeted to a new language or domain with little or no linguistic information or expertise other than an appropriate vocabulary.

ACKNOWLEDGMENTS

The work presented in this paper was supported (in part) by the European project NEPOMUK No. FP6-027705 and (in part) by the Lion project supported by Science Foundation Ireland under Grant No. SFI/02/CE1/I131.

REFERENCES

- [1] B. Davis, S. Handschuh, A. Troussov, J. Judge, and M. Sogrin, “Linguistically Light Lexical Extensions for Ontologies,” in *Proceedings of LREC 2008*, Marrakech, Morocco, 2008.
- [2] M. Espinoza, A. Gómez-Pérez, and E. Mena, “Enriching an ontology with multilingual information,” in *ESWC*, 2008, pp. 333–347.
- [3] P. Buitelaar, T. Declerck, A. Frank, S. Racioppa, M. Kiesel, M. Sintek, R. Engel, M. Romanelli, D. Sonntag, B. Loos, V. Micelli, R. Porzel, and P. Cimiano, “LingInfo: Design and Applications of a Model for the Integration of Linguistic Information in Ontologies,” in *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, 2006.
- [4] G. Francopoulo, M. George, N. Calzolari, M. Monachini, N. Bel, M. Pet, and C. Soria, “Lexical Markup Framework (LMF),” in *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC)*, 2006.
- [5] A. Gomez-Perez, O. Corcho, and M. Fernandez-Lopez, *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web. First Edition (Advanced Information and Knowledge Processing)*. Springer, July 2004. [Online]. Available: <http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/1852335513>
- [6] G. A. Miller, “Wordnet: a lexical database for english,” *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.
- [7] Piek Vossen, Ed., *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Norwell, MA, USA: Kluwer Academic Publishers, 1998.
- [8] Wim Peters and Piek Vossen and Pedro Díez-Orzas and Geert Adriaens, “Cross-linguistic Alignment of Wordnets with an Inter-Lingual-Index,” pp. 149–179, 1998.
- [9] R. Cilibrasi and P. M. B. Vitanyi, “The google similarity distance,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, p. 370, 2007. [Online]. Available: <http://www.citebase.org/abstract?id=oai:arXiv.org:cs/0412098>
- [10] D. Maynard and H. Cunningham, “Multilingual adaptations of annie, a reusable information extraction tool,” in *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 219–222.
- [11] A. Troussov and B. O. Donovan, “Morphosyntactic Annotation and Lemmatization Based on the Finite-State Dictionary of Wordformation Elements,” in *Proceedings of Speech and Computer (SPECOM)*, Moscow, Russia, 2003, pp. 27–29.
- [12] J. Daciuk, “Incremental Construction of Finite-state Automata and Transducers, and their Use in the Natural Language Processing,” Ph.D. dissertation, Technical University of Gdansk, 1998.

⁹Search performed on June 25th 2008