World WordNet Database Structure: An Efficient Schema for Storing Information of WordNets of the World

Hanumant Redkar, Sudha Bhingardive, Diptesh Kanojia, Pushpak Bhattacharyya

Center for Indian Language Technology, Indian Institute of Technology Bombay, Mumbai, India

hanumantredkar@iitb.ac.in, sudha@cse.iitb.ac.in, diptesh@cse.iitb.ac.in, pb@cse.iitb.ac.in

Abstract

WordNet is an online lexical resource which expresses unique concepts in a language. English WordNet is the first WordNet which was developed at Princeton University. Over a period of time, many language WordNets were developed by various organizations all over the world. It has always been a challenge to store the WordNet data. Some WordNets are stored using file system and some WordNets are stored using different database models. In this paper, we present the World WordNet Database Structure which can be used to efficiently store the WordNet information of all languages of the World. This design can be adapted by most language WordNets to store information such as synset data, semantic and lexical relations, ontology details, language specific features, linguistic information, etc. An attempt is made to develop Application Programming Interfaces to manipulate the data from these databases. This database structure can help in various Natural Language Processing applications like Multilingual Information Retrieval, Word Sense Disambiguation, Machine Translation, etc.

Introduction

The Princeton WordNet or the English WordNet (Miller, 1990) was the first WordNet which was developed at Princeton University. This WordNet inspired many countries and organizations in the world to develop WordNet in their own languages. Over a period of time, many individual language as well as multi-lingual WordNets evolved. Some of the individual language WordNets are GermaNet, Japanese WordNet, etc. and some of the multi-lingual WordNets are EuroWordNet (Vossen et al., 1997), IndoWordNet (Bhattacharyya, 2010), etc. All these WordNets use various methods to store their WordNet data. For example, Princeton WordNet uses text files to store WordNet data, whereas GermaNet uses relational database structure (Henrich et al., 2010). Also, some WordNets such as MultiWordNet uses multiple databases (Pianta et al., 2002) while WOLF (Sagot et al., 2008) uses XML file structure to store their data. All these storage methods are good in their own respects, but they also have some limitations – File system uses flat files which may not be efficient to store, manipulate and retrieve the relevant WordNet data. The database system uses relational databases to store data, but these databases do not capture all the WordNet and related data. XML also uses file system to store information.

We propose a World WordNet Database Structure (WWDS) which can possibly be made standard to store global WordNet data. It can be used to efficiently store and manipulate multi-lingual WordNet data. It uses multiple databases wherein the language independent information is stored in a single master database and language dependent information is stored in multiple language specific databases. The detailed description of WWDS is given in the following section. Further, the design and the database schema of WWDS are depicted. In the subsequent sections, the advantages and future work of WWDS are mentioned.

World WordNet Database Structure

World WordNet Database Structure or WWDS is designed based on IndoWordNet database structure (Prabhu et al., 2012). It consists of a single master database and multiple language specific databases. The master database is named as *wordnet_master*, which contains language independent data such as semantic relations, ontology details, etc., which are common across all the languages. The language specific database is named as *wordnet_<language>*, which contains language dependent data such as synsets, words, lexical relations, etc., which are specific to a particular language in consideration. For example, *wordnet_german*, *wordnet_hindi*, *wordnet_english* are language specific WordNet databases for German, Hindi, English languages respectively. In this way, we can make use of WWDS to store multi-lingual WordNets. In *wordnet_master* database, the table *wn_master_inter_lingual_synset* has inter-lingual synset id (*il_synset_id*) which links synset ids of all the language databases. This gives provision to access cross-lingual synset data. Application Programming Interfaces (APIs) for WWDS are developed based on IndoWordNet APIs (Prabhugaonkar et al., 2012). Also, an attempt has been made to develop scripts for importing data to WWDS.

Design of the WWDS

The basic design of WWDS has been borrowed from IndoWordNet database schema with further improvement to its database structure. This structure can possibly be adapted to most of the WordNets of the world. Figure 1 shows the basic block diagram of WWDS.

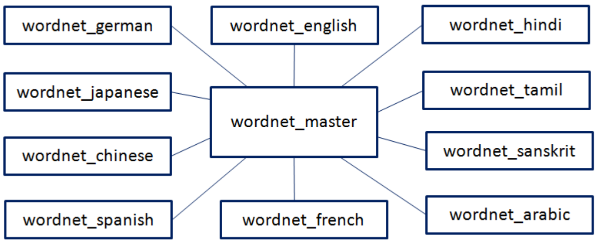

Figure 1. Block diagram of WWDS

WWDS Database Schema

Figure 2 depicts the WWDS database model diagram showing tables and their attributes. The left hand side of the figure has tables from *wordnet_<language>* database and the right hand side has tables from *wordnet_master* database. Primary key attributes are underlined and foreign keys attributes can be identified by the link between tables.

Some Advantages of WWDS

One of the major advantages of WWDS is that, using inter-lingual synset id, one can retrieve cross-lingual synset information of most of the languages. Another advantage is that, it is a centralized database which can be used in bigger applications like Cross Lingual Information Retrieval, Word Sense Disambiguation, Data Mining, etc.

Conclusion and Future Work

Multiple WordNets use various data storage methods such as flat files, databases, etc. In this paper, we have presented an efficient and normalized database schema *viz*. WWDS for storing, manipulating and retrieving multi-lingual WordNet information. WWDS is an attempt to link most of the WordNets together by providing a centralized storage structure. One can easily adapt this structure to store their WordNet data. Some of the applications of WWDS could be Cross Lingual Information Retrieval, Multi-lingual Word Sense Disambiguation, etc. APIs are built to access and manipulate the WordNet data. In the future, we plan to study features of individual languages and revise the WWDS. Also, we plan to store information such as corpus, annotated data, morphological features, etc in the WWDS.

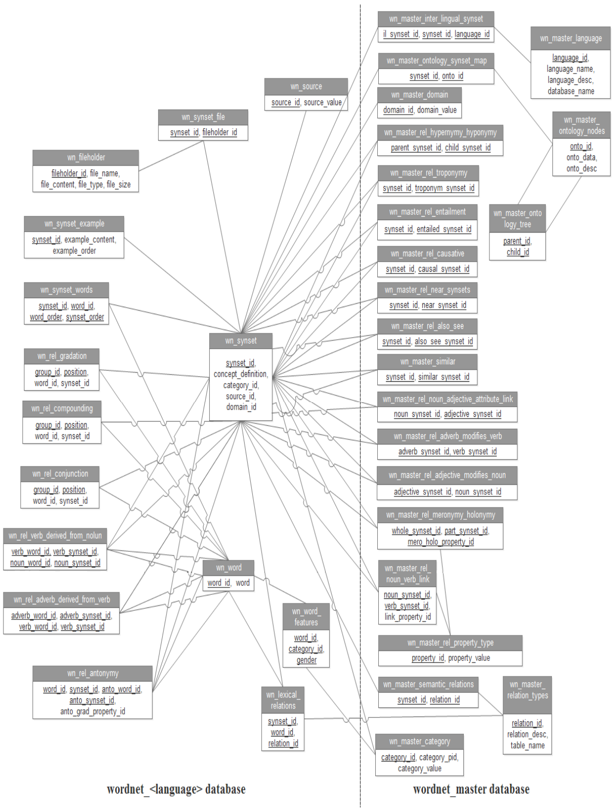
wn_fileholder
fileholder_id, file_name, file_content, file_type, file_size
wn_synset_file
synset_id, fileholder_id
wn_source
source_id, source_value
wn_synset_example
synset_id, example_content, example_order
wn_synset_words
synset_id, word_id, word_order, synset_order
wn_synset
synset_id, concept_definition, category_id, source_id, domain_id
wn_rel_gradation
group_id, position, word_id, synset_id
wn_rel_compounding
group_id, position, word_id, synset_id
wn_rel_conjunction
group_id, position, word_id, synset_id
wn_rel_verb_derived_from_nolun
verb_word_id, verb_synset_id, noun_word_id, noun_synset_id
wn_word
word_id, word
wn_rel_adverb_derived_from_verb
adverb_word_id, adverb_synset_id, verb_word_id, verb_synset_id
wn_word_features
word_id, category_id, gender
wn_rel_antonymy
word_id, synset_id, anto_word_id, anto_synset_id, anto_grad_property_id
wn_lexical_relations
synset_id, word_id, relation_id
wordnet_<language> database
wn_master_inter_lingual_synset
il_synset_id, synset_id, language_id
wn_master_language
language_id, language_name, language_desc, database_name
wn_master_ontology_synset_map
synset_id, onto_id
wn_master_domain
domain_id, domain_value
wn_master_ontology_nodes
onto_id, onto_data, onto_desc
wn_master_rel_hypernymy_hyponymy
parent_synset_id, child_synset_id
wn_master_rel_troponymy
synset_id, troponym_synset_id
wn_master_rel_entailment
synset_id, entailed_synset_id
wn_master_ontology_tree
parent_id, child_id
wn_master_rel_causative
synset_id, causal_synset_id
wn_master_rel_near_synsets
synset_id, near_synset_id
wn_master_rel_also_see
synset_id, also_see_synset_id
wn_master_similar
synset_id, similar_synset_id
wn_master_rel_noun_adjective_attribute_link
noun_synset_id, adjective_synset_id
wn_master_rel_adverb_modifies_verb
adverb_synset_id, verb_synset_id
wn_master_rel_adjective_modifies_noun
adjective_synset_id, noun_synset_id
wn_master_rel_meronymy_holonymy
whole_synset_id, part_synset_id, mero_holo_property_id
wn_master_rel_noun_verb_link
noun_synset_id, verb_synset_id, link_property_id
wn_master_rel_property_type
property_id, property_value
wn_master_semantic_relations
synset_id, relation_id
wn_master_relation_types
relation_id, relation_desc, table_name
wn_master_category
category_id, category_pid, category_value
wordnet_master database

Figure 2. Database model diagram of WWDS

References

Bhattacharyya, P. 2010. IndoWordNet. *Proc. of LREC-10*, Malta.

Henrich, V., & Hinrichs, E. W. 2010. GernEdiT-The GermaNet Editing Tool. *In ACL (System Demo)*, Malta. (pp. 19-24).

Miller, George A., R., Fellbaum, C., Gross, D., & Miller, K. J. 1990. Introduction to wordnet: An on-line lexical database. *International journal of lexicography*, OUP. (pp. 3.4: 235-244).

Pianta, E., Bentivogli, L., & Girardi, C. 2002. Developing an aligned multilingual database. *In Proc. 1st GWC*, Mysore, India.

Prabhu, V., Desai, S., Redkar, H., Prabhugaonkar, N., Nagvenkar, A., & Karmali, R. 2012. An Efficient Database Design for IndoWordNet Development Using Hybrid Approach. *COLING 2012*, Mumbai, India. (pp. 229).

Prabhugaonkar, N., Nagvenkar, A., & Karmali, Ramdas N. 2012. IndoWordNet Application Programming Interfaces. *COLING 2012*, Mumbai, India. (pp. 237 - 244).

Sagot, B., & Fišer, D. 2008. Building a free French wordnet from multilingual resources. *In OntoLex 2008*, Morocco.

Vossen, P. 1997. EuroWordNet: A multilingual database for information retrieval. *DELOS*, Zurich. (pp. 5-7).