AUTOMATIC GENERATION OF CONCEPT TAXONOMIES FROM WEB SEARCH DATA USING SUPPORT VECTOR MACHINE

Robertas Damaševičius
Software Engineering Department, Kaunas University of Technology Studentų 50, LT-51368, Kaunas, Lithuania

Keywords: Taxonomy learning, Web mining, Data mining, Machine learning, Support Vector Machine.

Abstract: Ontologies and concept taxonomies are essential parts of the Semantic Web infrastructure. Since manual construction of taxonomies requires considerable efforts, automated methods for taxonomy construction should be considered. In this paper, an approach for automatic derivation of concept taxonomies from web search results is presented. The method is based on generating derivative features from web search data and applying the machine learning techniques. The Support Vector Machine (SVM) classifier is trained with known concept hyponym-hypernym pairs and the obtained classification model is used to predict new hyponymy (is-a) relations. Prediction results are used to generate concept taxonomies in OWL. The results of the application of the approach for constructing colour taxonomy are presented.

1 INTRODUCTION

The Semantic Web is a vision for the future of the Web in which information is given explicit meaning, which makes it easier for machines to automatically process, interpret and integrate information available on the Web (Berners-Lee *et al.*, 2001). A critical part of the Semantic Web infrastructure are ontologies that define and structure the terms used to describe and represent an area of knowledge in an abstract and machine-interpretable form (Maedche and Staab, 2004). Ontologies are needed for many Semantic Web tasks such as for exchanging data between parties who have agreed to the definitions beforehand or for applications that search across or merge information from diverse sources. Ontologies also enhance the machine readability and understandability of web documents.

Domain concept taxonomies and ontologies are very important in software engineering as a part of domain analysis to facilitate knowledge representation, reuse and enable development of high-level system models (Damaševičius *et al.*, 2008; Damaševičius, 2009), and in e-Learning to support automated construction and sharing of learning resources (Štuikys *et al.*, 2008).

Ontologies use classes to represent concepts and define many different types of relations between classes, their instances and attributes. The central components of ontologies are taxonomies, which define only taxonomical relationships between concepts. In fact, many ontology development methodologies such as METHONTOLOGY (Fernandez-Lopez *et al.*, 1997) consider construction of a taxonomy of domain terms (concepts) as the initial stage of the ontology creation.

A taxonomy is a hierarchical representation of domain concepts based on a division of a set of domain concepts into a set of categories. As such, taxonomies constitute a central part of the conceptual models in many Semantic Web applications. Properly structured taxonomies allow to introduce order to the elements of a conceptual model, are particularly useful in presenting limited views of a model for human interpretation, and play a critical role in reuse and integration tasks (Welty and Guarino, 2001).

There are many different ways to construct a taxonomy. A taxonomy can be based on the semantics of the taxonomic relationship (hyponymy/hypernymy, is-a, subsumption, etc.), on different types of the taxonomical relations (generalization, specialization, subset hierarchy), on the constraints involved in multiple taxonomic relationships (covering, partition, etc.), or on the structural similarities between descriptions (Welty and Guarino, 2001).

DamaÅąeviÃ▪ius R.
AUTOMATIC GENERATION OF CONCEPT TAXONOMIES FROM WEB SEARCH DATA USING SUPPORT VECTOR MACHINE.
DOI: 10.5220/0001842206660673
In *Proceedings of the Fifth International Conference on Web Information Systems and Technologies (WEBIST 2009)*, page
ISBN: 978-989-8111-81-4

The manual design and construction of domain ontologies (e.g., Wordnet (Felbaum, 1998)) and, particularly, taxonomies is a time and labour-costly process that requires an extended knowledge of the domain and often results in knowledge acquisition bottleneck. Because of human expertise, the accuracy of manually constructed concept hierarchies is usually high. Therefore, approaches that reduce human effort and time requirements as well as provide even more accuracy and objectivity should be considered. Currently such approaches are usually based on mining of data source representing domain knowledge (e.g., web pages (Clerkin *et al.*, 2001; Kashyap *et al.*, 2005; Sombatsrisomboon *et al.*, 2003; Davulcu *et al.*, 2003), web search data (Sanchez and Moreno, 2004), web forms (Roitman and Gal, 2006), text corpora (Sanderson and Croft, 1999; Maedchen and Staab, 2000; Cimiano *et al.*, 2004), etc.) and attempt to create domain ontologies or parts thereof (semi-)automatically.

Automated techniques for ontology (taxonomy) mining, extraction and learning are considered by several researchers. Basically, there are two approaches for generating concept hierarchies:

1) *Natural language processing* (NLP) approaches are based on the statistical and syntactical analysis (parsing) of text and discovering significant patterns that can be applied for generating ontological concepts and relationships (Kashyap *et al.*, 2005; Sanderson and Croft, 1999; Daille, 1996; Degeratu and Hatzivassiloglou, 2002; Nakayama, 2008; Pottrich and Pianta, 2008). The disadvantage of NLP is that it requires significant human involvement, making it expensive and infeasible for many Semantic Web applications.

2) *Supervised machine learning* based approaches are based on constructing a large number of training examples from the available data for a classifier (such as a Support Vector Machine or Naïve Bayes classifier). A trained classifier then can be used to make predictions on the ontological relationships between concepts in new data. Based on these predictions, new taxonomies can be created (Clerkin *et al.*, 2001; Suryanto and Compton, 2002; Etzioni *et al.*, 2004), or existing taxonomies can be integrated (Zhang *et al.*, 2004). A description of the supervised and unsupervised approaches to extract semantic relationships between terms in a text document is presented in (Finkelstein-Landau and Morin, 1999).

The aim of this paper is to create an initial taxonomy of concepts using a supervised machine learning approach. We present a methodology to extract information from the web search results to build automatically a taxonomy of terms (concepts). The methodology is used to implement an agent for learning and generation of concept taxonomies using web search data.

The outline of the paper is as follows. Section 2 presents our taxonomy derivation methodology. Section 3 presents a case study in automatic taxonomy construction from web search data. Finally, Section 4 presents conclusions and discusses future work.

2 TAXONOMY DERIVATION METHODOLOGY

2.1 Analysis of Semantic Relationships in Taxonomy and Task Formulation

Further we accept the following definition of a taxonomy: "*A taxonomy is a system of knowledge organization that represents relationships between topics such that they arrange these concepts from general, broader concepts to more specific concepts*" (Kashyap *et al.*, 2005).

Taxonomy of concepts is a hierarchical structure, where concepts are related by hyponymy relation. Hyponymy (Fromkin and Rodman, 2008) is the relationship between a general term such as colour and specific instances of this term. For example, red, white, and blue are hyponyms of colour. Therefore, a hyponym has a narrower semantic range than its counterpart, a hypernym.

In knowledge representation and object-oriented programming, a hyponym-hypernym relationship is also known as the *is-a* relationship (subsumption). *Is-a* is a relationship where one class *A* is a subclass of another class *B* (and *B* is a superclass of *A*). In other words "*A* <*is-a*> *B*" usually means that concept *A* is a specialization of concept *B*, and concept *B* is a generalization of concept A. Formally, subsumption is defined as follows: a concept *A* is a sub-concept of a concept *B*, if $A \subseteq B$.

We can formulate our task as follows. Given a list of paired concepts (*A, B*), $A \in C$, $B \in C$, where *C* is a set of concepts, determine whether concepts *A* and *B* are related by the *is-a* (subsumption) relation.

The basis of our approach is the following hypothesis: given the abundance and redundancy of information on the internet, there is a fuzzy functional relation between the broadness of a concept and the spread of this concept on the internet. Since the expression of this functional

relationship is not clear, we use a binary supervised machine learning method to analyze web search results and to infer the taxonomical relationships between concepts.

Now we can formulate our task more detailed. Given a set of search queries Q and a set of logic relations R (only A, only B, or, and, not) $R=\{A,B,\cup,\cap,\neg\}$, $r:C\to C, r\in R$ between A and B belonging to a concept set C, where each query $q\in Q$, $q:(A,B,R)\to N$ returns an integer number N on concepts A and B, discover a taxonomical relationship prediction function $g:Q\to\{1,-1\}$, where 1 indicates a taxonomical relationship between sub-concept A and super-concept B, and -1 indicates that there is no taxonomical relationship between A and B. We separate the solution to this task into the following sub-problems, which are explained in detail later:

1) Selection of concept words and formulation of queries.
2) Derivative feature generation and dataset construction.
3) Taxonomy learning using machine learning approach.
4) Taxonomy representation and generation.
5) Taxonomy evaluation.

2.2 Selection of Concept Words and Formulation of Queries

We assume that each concept is characterized by a concept word. Concept words must satisfy the following requirements:

1) Concept words must have a minimum size (e.g. 3 characters) and must be represented with a standard ASCII character set.

2) Concept words must be relevant, i.e., prepositions, modal words, and common words ("stop words") can not be used as concept words.

Table 1: List of queries for web search engine.

Query name	Formal definition	Web search query
Parent	|B|	B
Child	|A|	A
Intersection	|B∩A|	B AND A
Union	|B∪A|	B OR A
Only Parent	|B|–|B∩A|	B –A
Only Child	|A|–|B∩A|	–B A

A standard web search engine is used as the provider of the knowledge on the concept. A search query is formed from the concept words and is sent to the web search engine. Considering our task formulation, queries must reflect possible logical relations between concepts. The list of queries for predicting the *is-a* relationship between the parent (super-concept) and child (sub-concept) concepts is presented in Table 1.

Additionally, each query may contain search restrictions, which allow to narrow search for obtaining more precise results: 1) *Domain restriction:* Restricts the search to documents in a web site. 2) *Position restriction:* Restricts the search to documents that contain the search word in the title or in the body text of the web documents. 3) *Search space restriction*: Restrict the search to documents that also contain additional words that allow to narrow/specify the domain. 4) *File type restriction:* Restricts search to documents of the specified type.

2.3 Derivative Feature Generation and Dataset Construction

The numerical data obtained from web search queries (the number of web documents satisfying the supplied search query) constitutes 6 primary features for each concept pair. The data is further processed and normalized to obtain the derivative features following such procedure. Each primary feature pair (f_1,f_2) is replaced with six derivative features:

$$\left(\begin{array}{l}(f_1>f_2)?(1):(-1),\dfrac{\min(f_1,f_2)}{\max(f_1,f_2)},\dfrac{\mathrm{abs}(f_1-f_2)}{f_1+f_2},\\ \dfrac{\mathrm{abs}(f_1-f_2)}{\max(f_1,f_2)},\dfrac{\min(f_1,f_2)}{f_1+f_2},\dfrac{\max(f_1,f_2)}{f_1+f_2}\end{array}\right) \quad (1)$$

Having 6 primary features this procedure allows us to obtain 90 derivative features thus expanding the search space for optimal separability between two categories of data.

Web mining results are further partitioned into two datasets. The training dataset is used to train a machine learning algorithm, and the testing dataset is used to evaluate its prediction accuracy. To avoid the problems associated with imbalanced datasets, we construct each dataset of 50% positive examples, when a given concept pair has a taxonomical relation, and of 50% negative examples, when a given concept pair has not a taxonomical relation. Dataset usage for concept relationship classification and prediction is summarized in Figure 1.

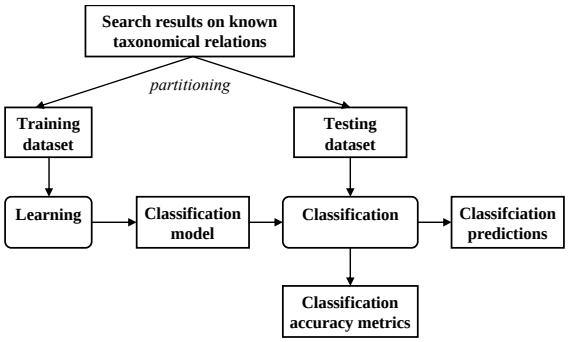

Figure 1: Dataset usage for concept relationship classification.

2.4 Example

We provide a small example how a dataset is constructed (see Figure 2). First, concept words that characterize the concepts are formulated. Each pair of concepts words is used to construct 6 web search queries following Table 1. The results of web search (the number of pages satisfying the queries) constitute a set of primary features. Derivative features are constructed from primary features using Eq. 1. These derivative features are further used by a classifier to make a prediction whether a given pair of concepts are related with *is-a* relationship or not.

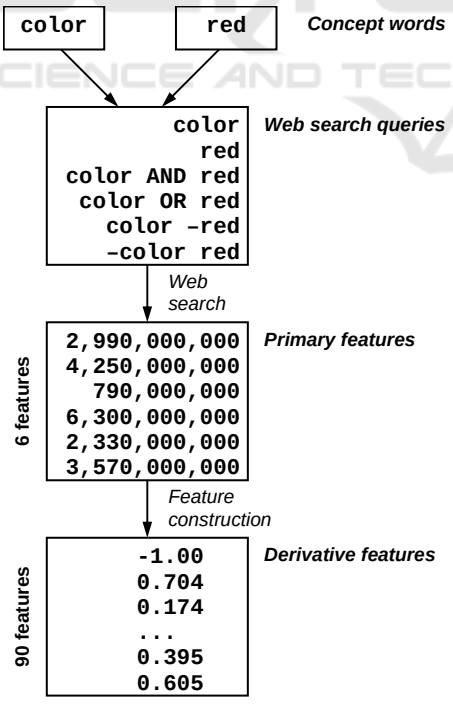

Figure 2: Example of dataset construction.

2.5 Taxonomy Learning using Machine Learning

For inferring relationships between concepts, we use Support Vector Machine (SVM; Cristianini, 2000), a supervised machine learning method for creating binary classification functions from a set of labeled training data. SVM requires that each data instance is represented as a vector of real numbers in *feature space*. First, SVM implicitly maps the training data into a (usually higher-dimensional) feature space. A *hyperplane* (decision surface) is then constructed in this feature space that bisects the two categories and maximizes the margin of separation between itself and those points lying nearest to it (the *support vectors*). This decision surface can then be used as a basis for classifying unknown vectors.

Consider an input space X with input vectors $x_i \in X$, a target space $Y = \{1,-1\}$ with $y_i \in Y$ and a training set $T = \{(x_1, y_1),...,(x_N, y_N)\}$. In SVM classification, separation of the two categories $Y = \{1,-1\}$ is done by means of the *maximum margin* hyperplane, i.e. the hyperplane that maximizes the distance to the closest data points and guarantees the best generalization on new examples. In order to classify a new point x_j, the classification function $g(x_j)$ is used:

$$g(x_j) = \mathrm{sgn}\left(\sum_{x_i \in SV} \alpha_i y_i K(x_i, x_j) + b\right) \quad (2)$$

where SV are the support vectors, $K(x_i, x_j)$ is the kernel function, α_i are weights, and b is the offset parameter.

The classification function is further used to predict categories of unknown data. If $g(x_j) = +1$, x_j belongs to the *Positive* (P) category, if $g(x_j) = -1$, x_j belongs to the *Negative* (N) category, and if $g(x_j) = 0$, x_j cannot be classified.

2.6 Taxonomy Representation and Generation

The obtained prediction results are used to construct a hierarchy of concepts with *is-a* relations. The hierarchy of concepts is represented using a standard ontology representation language: Web Ontology Language (OWL). OWL is a semantic markup language for publishing and sharing ontologies on the World Wide Web. OWL is supported by many

ontology visualizers and editors, such as Protégé 2.1 (Protege), allowing the user to easily explore, analyse or modify the ontology.

To resolve a problem of subclassing from multiple parents, each such subconcept is represented as a set of equivalent sub-concepts, which are subclassed from only one super-concept and related by the equivalence relation, i.e., if we have super-concepts c_1, c_2 and subconcept s such as s = isa(c_1, c_2) then s → (s_1, s_2), where s_1 = isa(c_1), s_2 = isa(c_2), $s_1 \equiv s_2$. The taxonomy generation algorithm is given in Figure 3.

```
algorithm Taxonomy generation
begin
   generate OWL file header
   for all concepts pairs (A, B)
      if prediction(A,B) = 1 then
         if class B is not generated then
            generate class B
         endif
         if subclass A is not generated then
            generate subclass A of class B
         else
            generate subclass A with modified name A_B
                     of class B equivalent with class A
         endif
   endfor
   generate OWL file footer
end
```

Figure 3: Taxonomy generation algorithm.

2.7 Taxonomy Evaluation

Our aim is to classify between concepts related and not related by the taxonomical (*is-a*) relationship. Therefore, here we have a binary classification problem in which the outcomes are labelled either as positive (*P*) or negative (*N*) category. There are four possible outcomes from a binary classifier. If the outcome from a prediction is *P* and the actual value is also *P*, then we have a *true positive* (*TP*); however if the actual value is *N* while a prediction is *P* then we have a *false positive* (*FP*). Conversely, a *true negative* (*TN*) has occurred when both the prediction outcome and the actual value are *N*, and *false negative* (FN) is when the prediction outcome is *N* while the actual value is *P*. To evaluate the precision of classification the following metrics will be used:

1) *Precision* (or *Positive Prediction Value*, PPV) is the number of items correctly labeled as belonging to the positive category divided by the total number of items labeled as belonging to the positive category:

$$PPV = \frac{n_{TP}}{n_{FP} + n_{TP}} \cdot 100\% \quad (3)$$

2) *Recall* (or *True Positive Rate*, TPR) is the number of TPs divided by the total number of items that actually belong to the positive category:

$$TPR = \frac{n_{TP}}{n_{TP} + n_{FN}} \cdot 100\% \quad (4)$$

3) *F-measure* (F) is the harmonic mean of precision and recall:

$$F = \frac{2 \cdot PPV \cdot TPR}{PPV + TPR} \quad (5)$$

Precision can be seen as a measure of exactness or fidelity, Recall is a measure of completeness, and F-measure is the measure of accuracy. The best possible classification method would yield 100% recall (no false negatives are found), 100% precision (no false positives are found), and 100% F-measure.

2.8 Architecture of Taxonomy Construction Framework

Our methodology for automated taxonomy construction was implemented as a set of tools, which is summarized in Figure 4. Taxonomy construction works in two modes: learning and generation and has two stages:

1) **Web Mining** stage**:** known examples of relations (learning mode) or assumptions on relations (generation mode) are given to Web Search Query Generator (WBSG). WBSG generates web search queries, connects to a standard Web Search Server and returns query results. The returned results are processed by Derivative Feature Generator (DFG), which generates derivative features and creates a training file (learning mode) or a data file (generation mode).

2) **Machine Learning** stage**:** SVM Learner uses a training file to generate a relationship classification model (learning mode) and SVM Classifier uses a data file and the classification model to generate predictions on relationships, whether these relationships belong to the positive or to the negative category (generation mode). The OWL Taxonomy generator uses these predictions as well as initial assumptions on relationships to generate a concept hierarchy (taxonomy) in OWL format.

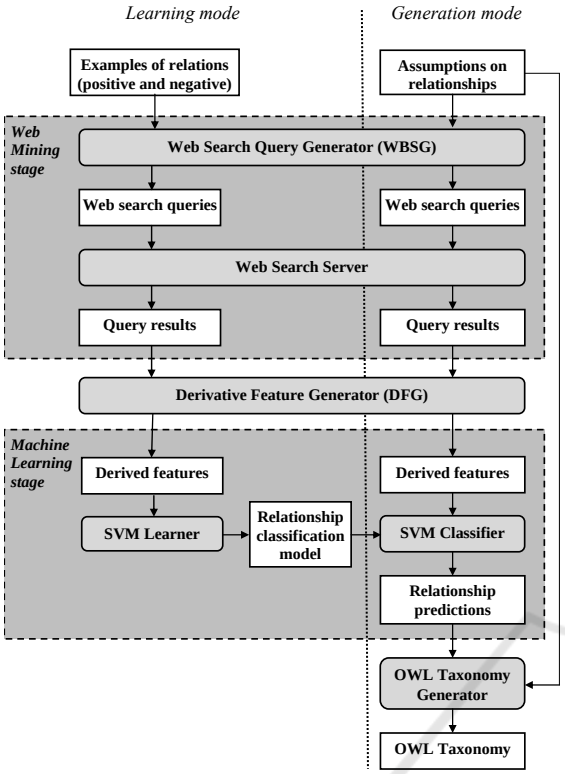

Figure 4: Taxonomy construction framework.

3 AUTOMATIC CONSTRUCTION OF COLOUR TAXONOMY

To validate our methodology we consider the following task: construct a taxonomy of colours according to their shade (tint). The construction of such taxonomy is not a trivial task, because words representing colour concepts can be divided into abstract colour words and descriptive colour words, though the distinction is blurry in many cases. Abstract colour words are words that only refer to a colour. In English white, black, red, yellow, green, blue, brown, and gray are abstract colour words. However, descriptive colour words are only secondarily used to describe a colour, but primarily are used to refer to an object or phenomenon that has that colour. For example, *Salmon* and *Lilac* are descriptive colour words in English because their use as colour words is derived in reference to natural colours of salmon flesh and lilac blossoms respectively. Such semantic blurriness aggravates the automated construction of colour taxonomy.

Furthermore, colour shade assignment in some cases is ambiguous: some of the shades are assigned to several colours, e.g., viridian is assigned to green and cyan colours. Also there are hierarchical relationships between colours, e.g., pink is defined as a shade of red, and white is defined as a shade of gray. Such ambiguities both in terms and in hierarchical relationships make colour taxonomy a good case study to evaluate our approach.

The list of colours used in this case study was extracted from (Wikipedia). It contains 11 main colours (white, pink, red, orange, brown, yellow, gray, green, cyan, blue, violet) and 198 unique shades. We have performed web mining experiments using Yahoo search engine with: 1) no restrictions on search space and keyword location; 2) concept word position restricted to document title; 3) search domain restriction (in English Wikipedia pages only; other authors have used Wikipedia as a source for ontology construction, too (Ponzetto and Strube, 2007; Cui *et al.*, 2008)); 4) search space restriction (searching only in documents that contain keywords "color" or "colour"); 5) file type restriction (search only in PDF and DOC documents).

For each experiment, web search results were post-processed (feature derivation performed) and the obtained datasets (with 90 features) were used to randomly construct training and testing datasets. Each dataset contains 216 examples (108 positive and 108 negative). The datasets were supplied to the SVM^{light} (Joachims, 2008) learner and classifier. We trained the SVM learner using RBF kernel and used the obtained classification model for the prediction of relationships in testing dataset. To evaluate the accuracy of predicted taxonomical relationships, the precision, recall, and F-measure metrics were used. The results are presented in Table 2.

Table 2: Taxonomical relationship prediction results.

No.	Query restriction	Restriction value	Classification accuracy		
			PPV	TPR	F
1	Global	-	61.19	36.28	45.56
2	Position	title	79.57	65.49	**71.84**
3	Domain	en.wikipedia.org	59.69	68.14	63.64
4	Search space	color OR colour	66.41	76.99	**71.31**
5	Doc. type	pdf	63.77	77.99	70.12
		doc	53.09	91.15	67.10

We can evaluate our results as satisfactory, given that a related research (Cimiano *et al.*, 2005) reports an F-measure of about 33% with regard to the accuracy of a less than 300 concept domain-dependent ontology generated from scratch (our generated taxonomy contains 235 concepts).

```
<owl:Ontology rdf:about="">
  <rdfs:label>Color taxonomy</rdfs:label>
</owl:Ontology>
<owl:Class rdf:ID="Color">
 <rdfs:subClassOf rdf:resource="#Thing" />
</owl:Class>
<owl:Class rdf:ID="red">
 <rdfs:subClassOf rdf:resource="#Color" />
</owl:Class>
<owl:Class rdf:ID="gray">
 <rdfs:subClassOf rdf:resource="#Color" />
</owl:Class>
<owl:Class rdf:ID="white">
 <rdfs:subClassOf rdf:resource="#gray" />
</owl:Class>
<owl:Class rdf:ID="pink">
 <rdfs:subClassOf rdf:resource="#red" />
</owl:Class>
<owl:Class rdf:ID="Amaranth">
 <rdfs:subClassOf rdf:resource="#pink" />
</owl:Class>
<owl:Class rdf:ID="AmaranthRed">
 <rdfs:subClassOf rdf:resource="red" />
 <rdfs:equivalentClass rdf:resource="#Amaranth"/>
</owl:Class>
```

Figure 5: A fragment of the generated taxonomy in OWL.

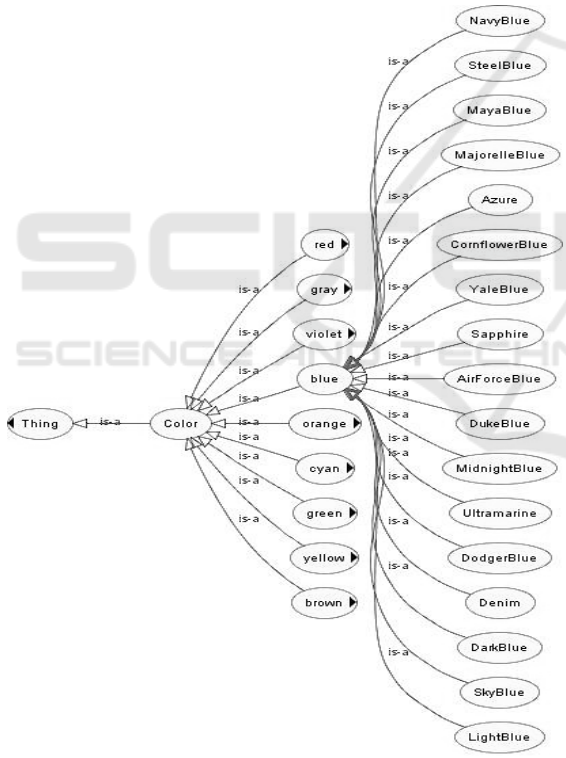

Figure 6: Visualization of generated taxonomy in Protégé (a fragment).

The prediction results were used to generate the colour taxonomy in OWL. Each colour is represented as a subclass of class `Color`, which is a subclass of class `Thing`. Each shade is represented as a subclass of specific colour class. A part of the generated OWL file is presented in Figure 5, and a fragment of its visualization is shown in Figure 6.

4 CONCLUSIONS AND FUTURE WORK

The approach for automated construction of concept taxonomies presented in this paper allows to construct more representative taxonomies, because it is based on the results obtained from World Wide Web (WWW), which is currently the largest pool of knowledge in the world, rather than from some text corpora. Furthermore, the construction of taxonomies is performed significantly quicker than using a manual construction method and requires little expert knowledge on the subject domain of the constructed taxonomy. The accuracy of the automatically constructed taxonomy is satisfactory considering the semantic ambiguities in the subject domain, the experiment was performed in. On the other hand, polisemy becomes a serious problem when the results obtained from the search engine are only based on the keyword's presence or absence. However, this problem is also present when constructing concept taxonomies manually.

The reliability of the constructed taxonomy can be increased by narrowing search queries to more reliable sub-webs of WWW (such as Wikipedia encyclopaedia), searching in documents that are expected to contain more formal knowledge (such as PDF which is a common format for presenting scientific papers in WWW), or constraining the search space by adding additional information on the subject domain of the taxonomy. However, the final evaluation of the automatically constructed taxonomy still should be left to the experts.

Future work will focus on the extension of our framework for predicting other (non-taxonomical) relationships such as meronomy (*has-a*) aiming for generation of richer domain ontologies. The second research direction will focus on the improvement of reliability of automatically constructed taxonomies by extending the framework with an intelligent agent that will search the web for the most reliable sub-webs of WWW as a source of knowledge for domain taxonomy/ontology construction.

REFERENCES

Berners-Lee, T., Hendler, J., Lassila, O., 2001. The Semantic Web. *Scientific American*, May 2001, pp. 29-37.

Cimiano, P., Hotho, A., Staab, S., 2004. Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. *Proc. of European*

Conf. on Artificial Intelligence ECAI 2004, pp. 435-439.

Clerkin, P., Cunningham P., Hayes, C., 2001. Ontology Discovery for the Semantic Web using Hierarchical Clustering. *Proc. of the Semantic Web Mining Workshop at ECML/PKDD 2001*, Freiburg, Germany.

Cristianini, N., Shawe-Taylor, J. 2000. *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press.

Cui, G., Lu, Q., Li, W., Chen, Y., 2008. Corpus Exploitation from Wikipedia for Ontology Construction. *Proc. of 6th Int. Language Resources and Evaluation Conference LREC 2008*, Marrakech, Morocco, 28-30 May.

Daille, B., 1996. Study and implementation of combined techniques for automatic extraction of terminology. In Resnick, P., Klavans, J. (eds.), *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*, MIT Press, Cambridge, MA.

Damaševičius, R., 2009. Ontology of Domain Analysis Concepts in Software System Design Domain. In Papadopoulos, G.A., *et al.* (eds.), *Information System Development: Design and Development*. Springer.

Damaševičius, R., Štuikys, V., Toldinas, E., 2008. Domain Ontology-Based Generative Component Design Using Feature Diagrams and Meta-Programming Techniques. *Proc. of 2nd European Conference on Software Architecture ECSA 2008.* LNCS 5292, pp. 338-341. Springer-Verlag, 2008.

Davulcu, H., Vadrevu, S., Nagarajan, S., 2003. OntoMiner: Bootstrapping and Populating Ontologies From Domain Specific Web Sites. *First Int. Workshop on Semantic Web and Databases*, Berlin, Germany.

Degeratu, M., Hatzivassiloglou, V., 2002. Building automatically a business registration ontology. *Proc. of the 2nd National Conf. on Digital Government Research*. DG.O.

Etzioni, O., Cafarella, M., Downey, D., Popescu, A.-M., Shaked, T., Soderl, S., Weld, D.S., Yates, E., 2004. Methods for domain-independent information extraction from the web: An experimental comparison. *Proc. of AAAI Conference*, pp. 391-398.

Fellbaum, C., 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Fernández-López, M., Gómez-Pérez, A., Juristo, N., 1997. Methontology: From Ontological Art Towards Ontological Engineering. *Spring Symposium on Ontological Engineering of AAAI*. Stanford University, CA, USA.

Finkelstein-Landau, M., Morin, E., 1999. Extracting Semantic Relationships between Terms: Supervised vs Unsupervised Methods. *Proc. of Int. Workshop on Ontological Engineering on the Global Information Infrastructure*, Dagstuhl Castle, Germany.

Fromkin, V., Rodman, R., 2006. *Introduction to Language*. Wadsworth Publishing, 8 edition.

Joachims, T., 2008. SVMlight: Support Vector Machine. Web site: http://svmlight.joachims.org/

Kashyap, V., Ramakrishnan, C., Thomas, C., Sheth, A., 2005. TaxaMiner: an experimentation framework for automated taxonomy bootstrapping. *Int. Journal of Web and Grid Services*, 1(2), pp. 240-266.

Maedche, A., Staab S., 2004. Ontology Learning. In Staab, S., Studer R. (Eds.), *Handbook on Ontologies*. International Handbooks on Information Systems. Springer, pp. 173-190.

Maedche, E., Staab, S., 2000. Discovering conceptual relations from text. *Proc. of 13th European Conf. on Artificial Intelligence, ECAI-2000*. IOS Press, pp. 321-325.

Nakayama, K., 2008. Extracting Structured Knowledge for Semantic Web by Mining Wikipedia. *Proc. of the 7th Int. Semantic Web Conference (ISWC 2008)*, Karlsruhe, Germany, October 28.

OWL. Web Ontology Language. W3C. Web site: http://www.w3c.org/TR/owl-features/.

Ponzetto, S.P., Strube, M., 2007. Deriving a large scale taxonomy from Wikipedia. *Proc. of the 22nd Conf. on the Advancement of Artificial Intelligence*, Vancouver, B.C., Canada, 22-26 July 2007, pp. 1440–1445.

Potrich, A., Pianta, E., 2008. L-ISA: Learning Domain Specific Isa-Relations from the Web. *Proc. of 6th Int. Language Resources and Evaluation Conference LREC 2008*, Marrakech, Morocco, 28-30 May.

Protégé 2.1. Web site: http://protege.stanford.edu/

Roitman, H., Gal, A., 2006. OntoBuilder: Fully Automatic Extraction and Consolidation of Ontologies from Web Sources Using Sequence Semantics. In *Current Trends in Database Technology*, Munich, Germany, March 26-31. Springer LNCS 4254, pp. 573-576.

Sánchez, D., Moreno, A., 2004. Automatic Generation of Taxonomies from the WWW. *Proc. of the 5 th Int. Conf. on Practical Aspects of Knowledge Management (PAKM 2004)*. LNAI. Springer, pp. 208-219.

Sanderson, M., Croft, B., 1999. Deriving concept hierarchies from text. *Proc. of 22nd Annual Int. ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 206-213. SIGIR.

Sombatsrisomboon, R., Matsuo, Y., Ishizuka, M., 2003. Acquisition of hypernyms and hyponyms from the WWW. *Proc. of 2nd Int. Workshop on Active Mining*.

Štuikys, V., Damaševičius, R., Brauklytė, I., Limanauskienė, V., 2008. Exploration of Learning Object Ontologies Using Feature Diagrams. *Proc. of World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA 2008),* pp. 2144-2154. Chesapeake, VA: AACE.

Suryanto, H., Compton, P. 2000. Learning classification taxonomies from a classification knowledge based system. *Proc. of ECAI'2000 Workshop on Ontology Learning OL'2000*, Berlin, Germany, August 25. CEUR Workshop Proceedings 31, pp. 1-6.

Welty, C.A., Guarino, N., 2001. Supporting ontological analysis of taxonomic relationships. *Data Knowledge Engineering*, 39 (1), pp. 51-74, 2001.

Wikipedia, 2008. List of colors. Web site: http://en.wikipedia.org/wiki/List_of_colors

Zhang, D., 2004. Web taxonomy integration using support vector machines. *Proc. of the 13th Int. Conf. on World Wide Web WWW '04*, pp. 472-481. ACM Press.