Towards extracting relations from unstructured data through natural language semantics

Diana Trandabăț[1,2]

[1] Faculty of Computer Science, University "Al. I. Cuza" of Iasi, Romania
[2] Institute of Computer Science, Romanian Academy
dtrandabat@info.uaic.ro

Abstract. Semantics has always been considered the hidden treasure of texts, accessible only to humans. Artificial intelligence struggles to enrich machines with human features, therefore accessing this treasure and sharing it with computers is one of the main challenges that the natural language domain faces nowadays. This paper represents a further step in this direction, by proposing an automatic approach to extract information from texts on the web by using semantic role labeling.

Keywords: artificial intelligence, natural language parsing, semantic roles, machine learning for semantic labeling.

1 Introduction

Attracted by the potential applications, more and more researchers from the artificial intelligence field submerged into the natural language processing domain. Thus, one further step in the human-computer interaction is the use of human languages instead of some pre-defined expressions. In order to teach a computer to understand a human speech, language models need to be specified and created from human knowledge. While still far from decoding political speeches, computer scientists, electrical engineers and linguists have all joined efforts in making the language easier to be learned by machines.

Our approach uses semantic role analysis in order to establish the roles that entities have in different contexts, and what are the temporal, modal or local constraint that determine or restrict an event to take place. A semantic role represents the relationship between a predicate and an argument. Semantic parsing, by identifying and classifying the semantic entities in context and the relations between them, has great potential on its downstream applications, such as text summarization or machine translation.

In this paper we propose a system which, starting from an input entity, extracts web pages found on a Google search for a particular entity, selects the snippets that contain the input entity, and then performs semantic role labeling to extract the relations between the entity and its context. Thus, our system creates a contextual map by

identifying the role an entity plays in different contexts, as well as the roles played by words frequently co-occurring with the input entity.

The motivation behind the work presented in this paper is the need to create a map of structured context related to a specific entity (e.g. a company or product name, an event, etc.). Through this map, the concepts that are usually in relation to the searched input entity are highlighted, together with their specific role (which can be of type Cause, Effect, Location, Time, etc.), thus providing a good material for social analyses, market research or other marketing purposes.

The paper is structured in 5 sections. After an introduction in the field of semantic role analysis, we briefly present the current work in Section 2. Section 3 introduces the overall application, describing the intermediary steps, while section 4 presents our approach for a Semantic Role Labeling system and evaluates it. The final section draws the conclusions of this paper and discusses further envisaged developments.

2 Semantic Role Analysis

Natural language processing is a key component of artificial intelligence. All content elements of a language are seen as predicates, i.e. expressions which designate events, properties of, or relations between, entities. The predication represents the mechanism that allows entities to instantiate properties, actions, attributes and states. More precisely, the linking between a phenomenon and individuals is known as predication. Predicates are not treated as isolated elements, but as structures, named predicate frames or semantic frames. Fillmore in [7] defined six semantic roles: Agent, Instrument, Dative, Factive, Object and Location, also called deep cases. His later work on lexical semantics led to the conviction that a small fixed set of deep case roles was not sufficient to characterize the complementation properties of lexical items, therefore he added Experiencer, Comitative, Location, Path, Source, Goal and Temporal, and then other cases. This ultimately led to the theory of Frame Semantics [6], which later evolved into the FrameNet project [1].

The semantic relations can be exemplified within the Commercial Transaction scenario, whose actors include a buyer, a seller, goods, and money. Among the large set of semantically related predicates, linked to this frame, we can mention buy, sell, pay, spend, cost, and charge, each of which indexes or evokes different aspects of the frame.

In the last decades, hand-tagged corpora that encode such information for the English language were developed (VerbNet [13], FrameNet and PropBank [18]). For other languages, such as German, Spanish, and Japanese, semantic roles resources are being developed. For Romanian, [23] has started to automatically build such a resource.

For role semantics to become relevant for language technology, robust and accurate methods for automatic semantic role assignment are needed. Automatic Labeling of Semantic Roles is defined as identifying frame elements within a sentence and tag them with appropriate semantic roles given a sentence, a target word and its frame [14]. Most general formulation of the Semantic Role Labeling (SRL) problem sup-

posed determining a labeling on (usually but not always contiguous) substrings (phrases) of the sentence s, given a predicate p.

In recent years, a number of studies, such as [3] and [8], have investigated this task on the FrameNet corpus. Role assignment has generally been modeled as a classification task: A statistical model is trained on manually annotated data and later assigns a role label out of a fixed set to every constituent in new, unlabelled sentences. The work on SRL has included a broad spectrum of probabilistic and machine-learning approaches to the task, from probability estimation [8], through decision trees [22] and support vector machines [20], to memory-based learning [15]. While using different statistical frameworks, most studies have largely converged on a common set of features to base their decisions on, namely syntactic information (path from predicate to constituent, phrasal type of constituent) and lexical information (head word of the constituent, predicate).

As for extracting relations, worthnoticing is the recent work done by the Watson group at IBM on relationship extraction and snippets evaluation with applications to question answering [21, 27].

3 Towards Semantically Interpreting Texts

The goal of the application we propose is to extract the context in which an entity occurs in web documents, together with the relations that the searched entity establishes with frequently co-occurring words. The basic architecture of our application contains a series of specialized modules, as follow:

3.1 User Input Acquiring Module

This module prompt the user for an input entity, usually representing a company or product name, an event name, a person, etc. A drop down list of the most frequent searched entities is offered as suggestion.

3.2 Web Page Retrieval Module

This module extract from the web the first n (in our tests n=200) web pages found on a Google search for the input entity. Google search options restricting the web search can be applied, such as selecting articles from newspapers, blogs or in a specific language.

3.3 Snippet Extraction Module

Using the Google snippet suggestion and some simple heuristics, the paragraphs containing the input entity are selected. A simple anaphora resolution method, based on a set of reference rules, is applied to the web pages, in order to link all entities to their referees.

The anaphoric system we used is a basic rule-based one, focusing on named entity anaphoric relations. Thus, we developed a rule-based system that performs the following actions:

- identifies a subset of a named entity with the full named entity, if it appears as such in the same text. For instance, Caesar is identified with Julius Caesar if both entities appear in the same text. Similarly, the President of Romania and the President are considered anaphoric relations of the same entity, if they appear in a narrow word window in the text.
- solves acronyms using a gazetteer we have initially built over the Internet, and which is continuously growing in size. For instance, *United States of America* and *USA* are co-references.
- searches for different addressing modalities and matches the ones that are similar. For instance, *John Smith* is co-referenced with *Mr. Smith*, and *Mary and John Smith* is co-referenced with *The Smiths*, or *The Smith Family*.
- solve pronominal anaphora in a simplistic way. Thus, if a pronoun (i.e. *she*, *he*, *him*, *his* etc.) is found in the text, and in the preceding sentence an entity is found, then we create an anaphoric link between the pronoun and its antecedent. A similar rule exists for companies, where the pronoun *it* may be linked to *the Insurance Company*, for instance.

3.4 Snippet Cleaning Module

After relevant paragraphs (containing the input word) are extracted, functional words such as (and, so, a, etc.) are eliminated from these paragraphs, since, being statistically too frequent for any kind of texts, they do not convey any useful information for semantic role labeling. A list of these functional words, created by using word frequencies in large corpora, is used.

3.5 Semantic Role Labeling Module

This module performs semantic role labeling on the obtained paragraphs, in order to identify the role the entity in question and the related entities plays. It will be described in details in the following section and evaluated in Section 6.

3.6 Module for the Creation of a Map of Concepts

This modules works in two steps: first, it extracts from the semantic role analysis the relations between the searched entity and the neighbor entities, creating a list of relations. Secondly, it generalizes the concepts that are found to be in relation with the searched entity across all extracted paragraphs, using the WordNet [5] hierarchy.

The next section presents the core module of this architecture, the semantic role labeling system, developed by training a set of supervised machine learning algorithms for several languages.

4 Our Semantic Role Labeling Approach

In order to detach semantic information from texts, we built a supervised SRL system, which we named PASRL – Platform for Adjustable Semantic Role Labeling. Several machine learning techniques and feature sets have been tested using the algorithms implemented in the Weka toolkit [26]. We used 12 of the most common machine learning algorithm that are available in Weka, such as decision trees, SVMs, memory-based learners, etc. A full description of the machine learning algorithm from Weka used for PASRL can be found in [25]. Each learning algorithm is trained with a set of features, the performances of the different obtained models are compared, and the best one is selected. The output of PASRL is a Semantic Role Labeling System which can be used to annotate new texts.

Using training data preprocessed with syntactic and dependency information, PASRL is composed of two main sub-systems: a Predicate Prediction module and an Argument Prediction module.

4.1 Predicate Prediction module

The first module of the semantic role labeling system is the predicate prediction module. The Predicate Prediction module system is composed out of three sub-modules:

- *Predicate Identification* this module takes the syntactic analyzed sentence and decides which of its verbs and nouns are predicational (can be predicates), thus for which ones semantic roles need to be identified;
- *Predicate Sense Identification* – once the predicates for a sentence are marked, each predicate need to be disambiguated since, for a given predicate, different sense may demand different types of semantic roles;
- *Joint Predicate and Predicate Sense Identification* – jointly identifies the predicates and their senses (the two above sub-modules).

Predicate Identification Task

Our semantic role labeling system uses the PropBank annotation of semantic roles. Since predicational words are not just verbs, beside PropBank [18] for the verbal frames, NomBank [20] is also used for nouns. Using syntactic annotation, consisting of marked dependency relations, and also the resources PropBank and NomBank, the system first tries to identify the words in the sentence that can behave as semantic predicates, and for which semantic roles need to be found and annotated (the Predicate Identification module). This module relies mainly on the external resources, thus the verbs that are in PropBank (have semantic frame annotation) are likely to be semantic predicates, those which aren't, are not predicational verbs, thus cannot have semantic arguments. For example, the verb to be has no annotation in PropBank, since it is a state and not an action, predicational, verb. Similarly, the NomBank is used to sort nouns that can behave as predicates from those that cannot have semantic arguments.

The predicate identification program transforms the annotated input into training instances for the ML algorithms in order to identify which nouns or verbs from the input are predicates. For each noun or verb in the sentence, an instance is created with a set of features from a syntax-based vector space, inspired from the features usually used for Semantic Role Labeling [14], and a binary class label (the candidate word is or not a predicate for the considered sentence). The features used for the Predicate Prediction are detailed in [24].The output of this module is the input file, where each verb or noun that behaves as a predicate is annotated.

After the predicates from the input sentence are identified, the next module is successively applied for all the predicates found in the sentence, in order to identify for all of them all and only their arguments. For example, for the sentence:

`The assignment of semantic roles depends on the number of predicates.`

two predicates are identified by the Predicate Identification module (assignment and depend). The next module will be applied two times, once for the identification of the sense and semantic arguments of the assignment predicate, and once for the depend predicate.

The output will therefore provide the two annotations:

`[The assignment of semantic roles]`$_{\text{ARG0}}$ `[depends]`$_{\text{TARGET}}$ `[on the number of predicates]`$_{\text{ARG1}}$`.`
`[The assignment]`$_{\text{TARGET}}$ `[of semantic roles]`$_{\text{ARG1}}$ `depends on the number of predicates.`

Predicate Sense Identification Module

Since this module considers that the predicational words are already identified, a binary feature is_predicate is extracted from the training file and added to the feature set created for the Predicate Identification task, and Weka classifiers are again ran to classify each predicate with its PropBank/NomBank role set. This module is needed in order to select the types of semantic roles specific to the sense the predicate has. The sense annotation in PropBank and NomBank is similar to some extent to the sense annotation from WordNet, with the observation that the classification in sense classes (role sets in PropBank's terminology) is centered less on the different meanings of the predicational word, and more on the difference between the sets of semantic roles that two senses may have. The senses and role sets in PropBank for a particular predicate are usually subsumed by WordNet senses, since the latter has a finer sense distinction.

Joint Predicate and Predicate Sense Identification

Instead of running the Predicate Identification and the Predicate Sense Identification processes successively, we tested running them simultaneously, using the same features presented above and as class the predicate role set similar to the one used for Predicate Sense Identification.

4.2 Argument Prediction Module

After running the first module of the semantic role labeling system (either pipelined or joint), the Argument Identification task is called in order to assign to each syntactic constituent of a verb / noun its corresponding semantic role. The instances in this case are not only the nouns and verbs, but every word in the sentence.

The Argument Prediction module performs argument prediction, based on the dependency relations previously annotated with the MaltParser [16] and the Predicate Prediction output. The input of this module contains syntactic information (part of speech and syntactic dependencies), predicate and predicate role set, and in the output each syntactic dependent of the verb is labeled with its corresponding role. This module uses as external resources PropBank and NomBank frame files and a list of frequencies of the assignment of different semantic roles in the training corpus. The features used are described in [24].

5 Evaluating PASRL

An evaluation metric for semantic roles have been proposed within CoNLL shared task on semantic role labeling [14]. The semantic frames are evaluated by reducing them to semantic dependencies from the predicate to all its individual arguments. These dependencies are labeled with the labels of the corresponding arguments. Additionally, a semantic dependency from each predicate to a virtual ROOT node is created. The latter dependencies are labeled with the predicate senses.

The evaluation of the PASRL performance was computed using 10-fold cross-validation on the training set. For each task, PASRL evaluates all the machine learning algorithms used against the gold-annotated corpus, and the best performing algorithm is saved in a configuration file. The evaluation was performed considering the number of correctly classified labels and correctly identified predicates.

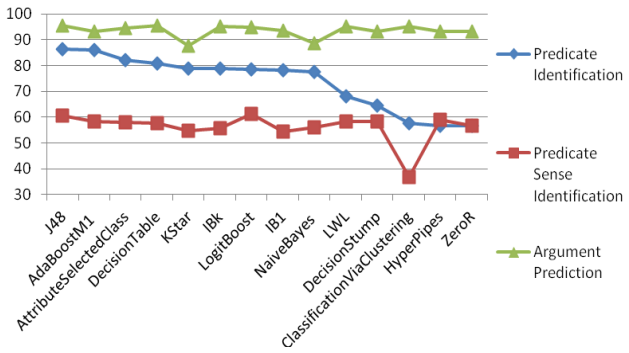

Fig. 1. Performance of the analyzed machine learning algorithms for the SRL tasks performed by PASRL

Fig. 1 presents an overview of the performance of different machine learning algorithms for each of the SRL tasks for English: Predicate Predication (using the Predicate Identification and the Predicate Sense Identification modules, not the joint learning module) and Argument Prediction. One can clearly see that in the Argument Prediction task the algorithms give best results, and the Predicate Sense Identification task is the most difficult task to model. Detailed results for the Predicate Prediction task and its sub-tasks are presented below.

For the Predicate Identification task, running the classifiers with the default weights of Weka for the English dataset, their results ranged from 86% correctly identified predicates to 56%. The algorithm that performs best is the J48 classifier (decision tree) and the one that performs worst is the simple ZeroR classifier (see Table 1).

Table 1. Top 5 ML algorithms evaluated with 10-fold cross-validation for English, for the Predicate Identification task

ML Algorithm	**10-fold cross validation**
J48	86.323
AdaBoostM1	86.016
AttributeSelectedClass	82.169
DecisionTable	80.921
KStar	78.97

Using boosting techniques with J48 as base classifier could improve further the module's performance. Changing the default weights of the classifiers can modify their performances, but we believe that the hierarchy will not change substantially. However, this remains a direction to address in a further work.

The best performing model (J48 in this case, which is a decision tree learning method) is saved and will be used when the Predicate Identification module will be called from the configuration file for annotating an unlabeled text.

The results for the Predicate Sense Identification Task are considerably worse than the ones for Predicate Identification task (see table 2), even if the results report an evaluation performed on a gold annotated input file. Therefore, the actual results are expected to be even worse. However, we notice that J48 is still among the best algorithms, and memory based algorithms generally perform badly on this subtask.

Table 2. Top 5 ML algorithms for the Predicate Sense Identification task for English

ML Algorithm	**10-fold cross validation**
LogitBoost	61.085
J48	60.641
HyperPipes	58.868
AdaBoostM1	58.322
DecisionTable	57.667

Instead of running the Predicate Identification and the Predicate Sense Identification processes successively, we tested running them simultaneously, using the same features presented above. Fig. 2 shows the difference in the performance obtained by using the pipelined Predicate Identification and Predicate Sense Identification module, as compared to the module that jointly learn Predicate and Predicate Sense Identification. One can notice that, although the results are significantly worse for the joint learning task, the algorithms that perform best for the pipelined task have still good performance in the joint task.

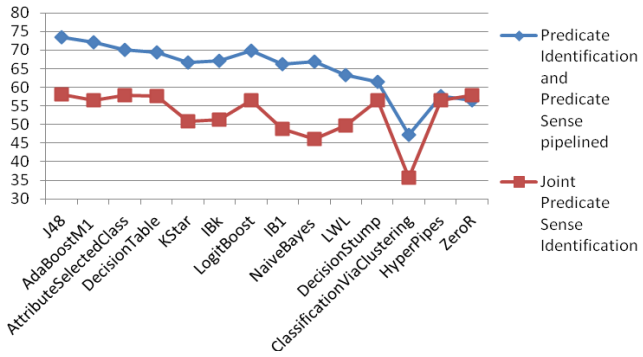

Fig. 2. Performance of the Predicate Prediction system using (1) the pipelined Predicate Identification and Predicate Sense Identification module tans (2) the joint Predicate and Predicate Sense Identification

When considering the 5 best classifiers for all the sub-tasks trained on the whole training data, evaluated using 10-fold cross-validation, we can observe that J48 and Decision Tables are among the 5 best algorithms for different languages, which suggests that some algorithms may perform better than others for Semantic role Labeling.

6 PASRL in Multilingual Context

PASRL was developed using training sets for different languages: English, German, Chinese, Czech and Japanese, provided for research purposes by the CoNLL 2009 shared task. The training data consisting of manually annotated treebanks such as the Penn Treebank for English, the Prague Dependency Treebank for Czech and similar treebanks for Chinese, German and Japanese languages, enriched with semantic relations. This allows for multilingual comparison to be performed. Thus, one can notice that the best performing algorithms score differently for different languages.

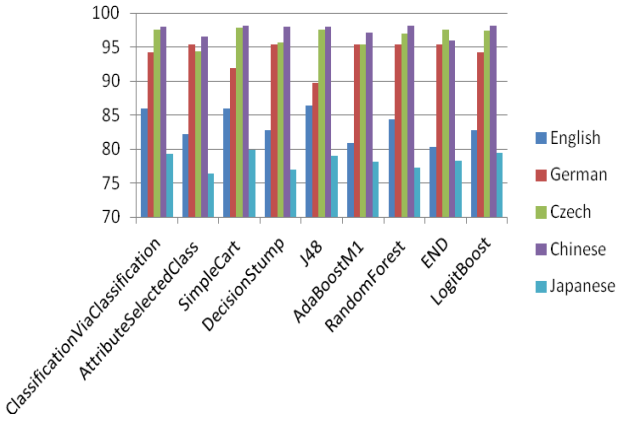

Fig. 3. Performance of the analyzed machine learning algorithms for the Predicate Identification task for different languages

For instance, as figure 3 shows, the best performances for Predicate Identification for Czech, German and Chinese range around 97-98%, while for English the best performance is 86% and for Japanese only almost 80%.

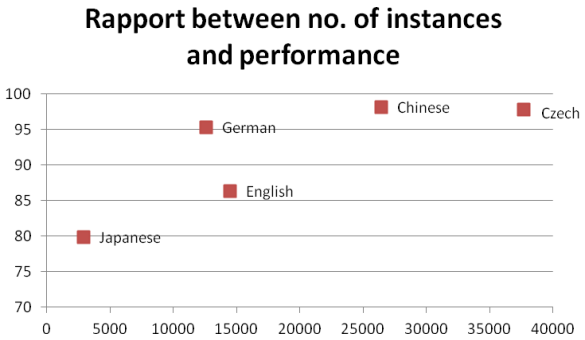

Fig. 4. Rapport between the number of instances and the performance of the Predicate Identification module in different languages

Since the size of the training corpus is similar (see figure 4 for an overview of the number of training instances), one possible explanation could be the different level of inflection, the free vs. fixed word order, or the position of the verb with respect to the other elements of the sentence (SVO vs. SOV language type), or a combination thereof.

7 Conclusions

This paper presented a semantic role labeling system developed using supervised machine learning algorithms from the Weka framework. This system is used in an application that monitors the contexts in which a specific entity appears in web texts and the relations it has with other co-occurring concepts. The developed SRL platform can be used for different languages, provided that a training corpus annotated with semantic roles is available. After testing several classifiers on different subproblems of the SRL task (Predicate Identification, Predicate Sense Identification, Predicate and Sense Identification, Argument Prediction), the proposed system chooses the algorithm with the greatest performance and returns a Semantic Role Labeling System (a sequence of trained models to run on new data).

The envisaged application of our system is in the marketing field, where the related concepts our system offers can be used to monitor the reaction of the consumer to different changes in a company's marketing policies.

Acknowledgments This work was supported by a grant of the Romanian National Authority for Scientific Research, CNCS – UEFISCDI, project number PN-II-RU-PD-2011-3-0292.

References

1. Baker G., Collin F., Fillmore, Charles J., and Lowe, John B. "The Berkeley FrameNet project". *In Proceedings of COLING-ACL*, Montreal, Canada. 1998.
2. Budanitsky Alexander and Graeme Hirst. "Evaluating wordnet-based measures of semantic distance". *Computational Linguistics*, 32(1):1347, 2006.
3. Chen J. and O. Rambow. "Use of deep linguistic features for the recognition and labeling of semantic arguments". In *Proceedings of EMNLP 2003*.
4. Daelemans, W., Zavrel, J., K. van der Sloot and A. van den Bosch. *TiMBL: Tilburg Memory Based Learner*, version 5.1, Reference Guide. Technical report, ILK Technical Report Series 04-02, 2003.
5. Fellbaum Christiane (1998, ed.) WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.
6. Fillmore Charles J. "Frame semantics", în *Linguistics in the Morning Calm*, Hanshin Publishing, Seoul , 1982, 111-137.5
7. Fillmore Charles J. "The case for case". In Bach and Harms, editors, *Universals in Linguistic Theory*, pages 1-88. Holt, Rinehart, and Winston, New York, 1968.
8. Gildea Daniel and Daniel Jurafsky. "Automatic labeling of semantic roles". *Computational Linguistics*, 28(3):245-288, 2002.
9. Hacioglu Kadri and Wayne Ward. "Target word detection and semantic role chunking using support vector machines". *In Proc. of HLT/NAACL-03*, 2003
10. Hacioglu Kadri. "Semantic role labeling using dependency trees". In *Proceedings of the 20th international conference on Computational Linguistics COLING'04*, Morristown, NJ, USA, 2004
11. Klein Dan and Christopher D. Manning. "Accurate unlexicalized parsing". *In Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pages 423-430, 2003.

12. Kudo Taku. *Machine Learning and Data Mining Approaches to Practical Natural Language Processing*. PhD thesis, School of Information Science, Nara Institute of Science and Technology, 2003.
13. Levin B. and M. Rappaport Hovav. *Argument Realization. Research Surveys in Linguistics Series.* Cambridge University Press, Cambridge, UK, 2005.
14. Marquez Lluis, Xavier Carreras, Kenneth C. Litkowski, and Suzanne Stevenson. "Semantic role labeling: An introduction to the Special Issue". *Computational Linguistics*, 34(2):145-159, 2008.
15. Morante Roser, Walter Daelemans, and Vincent Van Asch. "A combined memory-based semantic role labeler of English". In *Proceedings of CoNLL*, pp 208-212, Manchester, UK, 2008.
16. Nivre J. "An efficient algorithm for projective dependency parsing". In *Proc. of the 8th International Workshop on Parsing Technologies* (IWPT 03), pp. 149-160, 2003.
17. Pado Sebastian and Mirella Lapata. "Dependency-based construction of semantic space models". *Computational Linguistics*, 33(2), 2007.
18. Palmer Martha, Daniel Gildea, and Paul Kingsbury. "The proposition bank: An annotated corpus of semantic roles". *Computational Linguistics*, 31(1):71-106, 2005.
19. Pennacchiotti Marco, Diego De Cao, Paolo Marocco, and Roberto Basili. "Towards a Vector Space Model for FrameNet-like Resources". In *Proceedings of LREC'08*, Marrakech, Morocco 2008.
20. Pradhan Sameer, Kadri Hacioglu, Valeri Krugler, Wayne Ward, James H. Martin, and Daniel Jurafsky. "Support vector learning for semantic argument classification". *Machine Learning Journal*, 60(13):11-39, 2005.
21. Schlaefer,N., J Chu-Carroll, E Nyberg, J Fan, W Zadrozny, D Ferrucci, "Statistical source expansion for question answering", Proceedings of the 20th ACM international conference on Information and and Knowledge Management (CIKM) 2011.
22. Surdeanu M., S. Harabagiu, J. Williams, and P. Aarseth. "Using predicate-argument structures for information extraction". In *Proceedings of ACL2003*, pp 8-15, Tokyo, 2003.
23. Trandabăţ D. "Towards automatic cross-lingual transfer of semantic annotation", in *6e Rencontres Jeunes Chercheurs en Recherche d'Information (RJCRI) 2011*, ISBN 978-2-35768-024-1, pp. 403-408, 16-18 March, Avignon, France, 2011.
24. Trandabăţ D. "Mining Romanian texts for semantic knowledge", in *Proceedings of Intelligent Systems and Design Application Conference*, ISDA2011, Cordoba, Spain, ISSN: 2164-7143, ISBN: 978-1-4577-1676-8, pp. 1062-1066, 2011.
25. Trandabăţ Diana. *Natural language processing using semantic frames*. PhD Thesis, 2010, http://students.info.uaic.ro/
~dtrandabat/thesis.pdf.
26. Witten Ian H. şi Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques* (Second Edition). Morgan Kaufmann, 2005.
27. Chu-Carroll, J., J Fan, N Schlaefer, W Zadrozny 2012 "Textual resource acquisition and engineering" - IBM Journal of Research and Development, vol. 56 3.4: IBM, pp. 4–1, 2012.