DOI 10.1007/s10559-016-9815-4
*Cybernetics and Systems Analysis, Vol. 52, No. 2, March, 2016*

# A METHOD FOR AUTOMATIC CONSTRUCTION OF ONTOLOGICAL KNOWLEDGE BASES. II. AUTOMATIC IDENTIFICATION OF SEMANTIC RELATIONS IN ONTOLOGICAL NETWORKS*

**O. O. Marchenko**

UDC 681.3

**Abstract.** *A semantic-syntactic model of natural language is presented. After the factorization of constructed tensors of the model, vectors representing the semantic-syntactic valence of words and describing the commutative behavior of words in a sentence are generated. A method is developed for computing vectors of the semantic-syntactic valence of concepts in an ontology that form an implicit description of their semantic relations. An algorithm is proposed for extracting explicit semantic relations between ontological concepts from vectors of their semantic-syntactic valence.*

**Keywords:** *automatic knowledge extraction, corpus linguistics, ontology, nonnegative tensor factorization.*

## INTRODUCTION

Factored tensorial linguistic models make it possible to quite successfully and automatically extract linguistic structures such as selectional preferences in sentences [1] and verb subcategorization frames [2] from text corpora that include data on semantic and syntactic properties of relationships (links) between verbs and their arguments, i.e., nouns, in sentences. This implies the possibility of automatic extraction of semantic relations such as theta roles (cases of Fillmore–Gruber) [3] from the obtained latent semantic space. Theta roles in an ontology represent a system of different connections of predicate-argument type between concepts-nodes describing some processes or actions and concepts that are used in schemes of realization of such processes in some role or other, for example, in the role of subject, object, recipient, etc. Thus, theta roles describe the predicate-argument structure of the semantics of semantic concepts-verbs. The definition of semantic connections between concepts in ontologies in the course of processing and analyzing decomposed tensors of text corpora allows one to automate the filling of ontological knowledge bases with content.

In [4], the construction of a semantic-syntactic model of natural language is described on the basis of nonnegative factorization of linguistic tensors collected as a result of frequency analysis of syntactic structures of sentences from large text corpora. Vectors from matrices of factored tensors of the model determine the following communicative properties of words: the type of relations to which belong these words and words with which these relations are established.

We call the vector that belongs to the matrix of a factored linguistic tensor and corresponds to some word the vector of semantic-syntactic valence of this word at the syntactic position of the corresponding matrix.

The main intricacy is that, in constructing tensors of semantic-syntactic word compatibility, lexemes, i.e., words that have different concepts-meanings, are objects of investigation and analysis. The vector representation of the semantic-syntactic valence of any word $w$ determined by the vector corresponding to it in the decomposed tensor matrix is

*First part of this paper was published in No. 1, 2016.

Taras Shevchenko National University of Kyiv, Kyiv, Ukraine, *omarchenko@univ.kiev.ua*. Translated from Kibernetika i Sistemnyi Analiz, No. 2, March–April, 2016, pp. 30–36. Original article submitted July 16, 2015.

  

intrinsically the sum of summand vectors of separate different semantic meanings inherent in the given word $w$, i.e., concepts $Sw_1, Sw_2, \ldots, Sw_t$ in some ontology. Based on the valence vector $(x_1, x_2, \ldots, x_k)$ of the word $w$, it is necessary to obtain summand valence vectors $(x_{11}, x_{12}, \ldots, x_{1k})$, $(x_{21}, x_{22}, \ldots, x_{2k}), \ldots, (x_{t1}, x_{t2}, \ldots, x_{tk})$ for each of its $t$ concepts-meanings. The valence vector of a fixed concept-meaning in an ontology is an implicit description of its semantic relations with other concepts in an ontological knowledge base. The simplest operation of computing the scalar product of valence vectors of two concepts in an ontology can confirm the presence of a semantic connection between them.

A method is developed for the determination of semantic relations between concepts-synsets of WordNet [5]. It is implemented by means of analysis of matrices of the decomposed tensor created during processing the corpora of articles of English Wikipedia and Simple English Wikipedia with splitting vectors of semantic valence of words into component vectors of semantic valence of their different concepts-meanings and with attaching split vectors to the corresponding conceptual nodes of WordNet [6]. The proposed method is tested for the accuracy of partition of vectors of semantic valence of words into summand vectors of semantic valence of concepts-meanings of these words and also for the accuracy of their attachment to WordNet synsets [5]. The main advantage of this method is the complete automation of determining new semantic relations between concepts in an ontological knowledge base in the course of analysis of matrices of the factored tensor of large text corpora. Despite the fact that relations are specified implicitly, i.e., by vectors of semantic valences, it is precisely this form of representation that makes it possible to solve classical computational linguistics problems such as word sense disambiguation, measurement of semantic proximity of words, semantic analysis of texts, etc.

## SPLITTING OF VECTORS OF SEMANTIC-SYNTACTIC VALENCES OF WORDS INTO SUMMAND VALENCE VECTORS OF THEIR DIFFERENT CONCEPTS-MEANINGS

After factoring the matrix $D$ and the three-dimensional tensor $F$ (see Part 1), the following two matrices are respectively obtained: $W$ and $H$ for circular syntagmatic connections and three matrices $X$, $Y$, and $Z$ for predicative linear connections that consist of vectors of length $k$, i.e., the degree of factorization. Each vector from these matrices corresponds to some word or word combination. Vectors describe the semantic-syntactic behavior of a lexeme, namely, the syntactic positions at which the lexeme forms connections with words, the words themselves, and of the types of such connections. As has been already noted, words are ambiguous, i.e., they, as a rule, have several meanings. The vector of a word is the sum of vectors of all its concepts-meanings. One word can have some vectors belonging to different matrices and corresponding to different syntactic positions. The problem of splitting of each of these vectors is independently solved. The developed algorithm of splitting semantic-syntactic valence vectors of a word into a set of vectors of semantic-syntactic valences of all its WordNet meanings-synsets solves the following problem.

A vector of semantic valence $X_w$ of dimension $k$ is given that corresponds to some word $w$ in the matrix $X$ (or in any other matrix out of the mentioned five matrices on which the method similarly operates). Assume that to the noun $w$ corresponds $t$ meanings-synsets in WordNet. It is required to divide $X_w$ into addends $X_{w_1}, X_{w_2}, \ldots, X_{w_t}$ corresponding to these $t$ synsets.

The algorithm of splitting semantic-syntactic valence vectors of words into summand valence vectors of their different concepts-meanings, i.e., WordNet concepts-synsets, is described in [5]. It performs the analysis of each of $k$ concepts-meanings of the vector $X_w$ with the subsequent determination of the concept-meaning among $t$ concepts-meanings of the word $w$ and, respectively, among $t$ synsets of WordNet to which each concrete nonzero concept-meaning of the vector belongs. Thus, the algorithm splits the semantic-syntactic valence vector of the word $w$ into $t$ vectors of its separate concepts-meanings. In this case, the algorithm attaches the obtained vectors to concrete WordNet synsets.

## EXPERIMENTS WITH THE SPLITTING AND ATTACHMENT ALGORITHM

The estimates for the accuracy of operation of the splitting and attachment algorithm on vectors from the matrices $X$, $Y$, $Z$, $W$, and $H$ that specify linear predicative and circular syntagmatic connections in sentences were computed according to the methodology described in [5]. As a result, the following estimates are obtained for the accuracy of

operation of the algorithm of splitting valence vectors of words into summand vectors of their different concepts-meanings and attaching them to WordNet synsets: 93.17% for $X$; 85.93% for $Y$; 87.81% for $Z$; 92.89% for $W$; 90.03% for $H$. These estimates testify to a high performance of the algorithm and perspectives of its use.

Note that a considerable advantage of the proposed method consists of a high degree of automation of each of its operation stages consisting of parsing articles, composing a tensor, nonnegatively factoring the tensor, and splitting vectors of semantic valence of words into vectors of their concepts-meanings and attaching them to the corresponding WordNet synsets.

The specification of semantic relations between concepts-nodes of an ontological graph in implicit form with the help of $k$-dimensional vectors of semantic valences also has the advantage of the universality of representation of semantic connections. After detecting the $(n+1)$th type of connection between existing concepts, the system fixes its presence in its base in vector representation without making an immediate demand of supplementing the list of relations with a new type, its complete description in the ontology, and the specification of the corresponding syntactic pattern for it.

There also is an essential drawback in this representation. In the model obtained, the presence of a linear predicative $\alpha$–$\beta$-connection between concrete three WordNet concepts-nodes $a$, $b$, and $c$ or a circular syntagmatic $\alpha$–$\beta$-connection between the concepts $a$ and $b$ can be easily verified, which is sufficient for a number of algorithms for parsing sentences. However, the algorithms using the search for the shortest paths-chains between nodes of ontologies cannot directly apply this model without the explicit description of relations between concepts-nodes in the form of incident edges. Moreover, the consideration of the replenishment of a semantic network with new relations is an intricate question as a process of enriching ontologies in the conventional understanding without the generation of explicit descriptions of semantic connections between conceptual nodes.

This implies the need for the construction of a method for automatic extraction of explicit semantic connections from semantic-syntactic valence vectors of concepts in an ontology.

## ALGORITHM OF EXTRACTION OF EXPLICIT SEMANTIC CONNECTIONS FROM SEMANTIC-SYNTACTIC VALENCE VECTORS OF CONCEPTS OF AN ONTOLOGY

The idea of the method for extracting explicit semantic connections of predicative-argument or circular syntagmatic type from the vector model of representation of relations between nodes-concepts in an ontology consists of repeated reading of processed texts of a corpus by means of the parsing procedure implemented based on the Cocke–Younger–Kasami algorithm [7] operating on the basis of semantic-syntactic valence vectors of concepts attached to WordNet synsets.

This process can be schematically represented as follows.

The algorithm sequentially parses sentences of texts from a training corpus and constructs control spaces of syntactic structures.

When the algorithm should know whether the construction of a circular syntagmatic connection between words $a$ and $b$ is possible, it transits to WordNet synsets-nodes $\{A_i\}$ referred to by the word $a$ and to synsets-nodes $\{B_i\}$ referred to by the word $b$.

Next, it computes the concepts-meanings of $(W_{a'}, H^{\mathrm{T}}_{b'})$ for words $a' \in \{A_i\}$ and $b' \in \{B_i\}$ if, among them, there are $a''$ and $b''$ for which $(W_{a''}, H^{\mathrm{T}}_{b''}) > T1$ (where $T1$ is some threshold level determined empirically, for example, $T1 \geq 1$), and then a circular syntagmatic connection is formed and a semantic relation of the type "object–property" is established between the concept-synset of $A_k : a'' \in A_k$ and the concept-synset of $B_j : b'' \in B_j$.

The existence of at least one such a pair (namely, $a''$ and $b''$) guarantees that the data for the matrix $D$ are obtained from the same texts and sentences that are processed at the current moment. The corresponding syntactic relation between the words $a$ and $b$ is put in the matrix $D$ and is added to vectors of its factored matrices $W$ and $H$. During splitting these vectors of semantic-syntactic valences, the corresponding concepts-meanings are attached to some two synsets, it is precisely these synsets that will be found by the described algorithm, and this relation will be established between them.

The algorithm assures that the sought-for semantic relation between synsets will be correctly found and identified under the following two conditions:

— during training, the corresponding syntactic link (connection) is correctly written by the Stanford Parser in the subordination tree and parse tree of a sentence and, as a result, the corresponding control space used for training the system is correctly composed, this link is correctly written in the matrix $D$, and, hence, it is surely present in vectors of the factored matrices $W$ and $H$;

— the algorithm of splitting vectors of semantic-syntactic valence has correctly split the vectors from $W$ and $H$ that correspond to this connection and has correctly performed the attachment to correctly determined WordNet nodes-synsets.

The made experiments testify to the high accuracy of operation of the algorithms being used and, hence, to the high reliability of this method of determination of semantic relations between WordNet synsets.

To increase the degree of reliability of the algorithm, an additional test for the presence of similar connections between children and/or parents of given synsets $A_k$ and $B_j$ can be introduced. If these connections exist, i.e., $\exists A'_k$ is a parent/child synset of $A_k$, $\exists B'_j$ that is a parent/child synset of $B_j$, and $\exists a' \in A'_k$ and $\exists b' \in B'_j$ such that $(W_{a'}, H^{\mathrm{T}}_{b'}) \geq T1$, then the probability of construction of a correct semantic connection between the synsets $A_k$ and $B_j$ considerably increases.

Note that the connections between the synsets $A_k$ and $B_j$ will be found immediately if the above conditions are fulfilled. If a connection is not found during training or is found but is incorrectly attached to inappropriate synsets, then additional tests increase the reliability of the operation of the method for determining semantic connections between synsets.

Intrinsically, to repeatedly read texts of a training corpus, a parser-builder of control spaces (see [4]) is used with the only difference that, in [4], to determine the presence of connections between words, the vectors of semantic-syntactic word valences obtained after the factorization of the matrix $D$ and tensor $F$ were used but, in this article, the transition is carried out from words to their concepts-meanings, i.e., WordNet synsets since, in this model, semantic-syntactic valence vectors are attached only to synsets. Thus, the parser "forcedly" passes from words and word combinations to their semantic meanings, and this phase of the semantic analysis is carried out concurrently with parsing.

Let us consider the case when the algorithm reveals more than one pair of synsets $A_k$ and $B_j$ $(a'' \in A_k, b'' \in B_j) : (W_{a''}, H^{\mathrm{T}}_{b''}) > T1$.

The Cocke–Younger–Kasami algorithm is the process of dynamic assembling of all possible versions of the syntactic structure of a sentence. At each level of the process of constructing the control space of a syntactic structure, two structures (two points of control spaces) are merged into one larger structure, i.e., a point that, in some way or other, inherits the lexical meaning from its constituent points. The further attachment at a higher structural level will occur according to vectors of semantic-syntactic valence of the synsets that will be referred to by this new lexical meaning of the obtained structure, i.e., a point of the control space (CS). Therefore, it is possible to preserve unions of all possible pairs of $A_k$ and $B_j$ $(a'' \in A_k, b'' \in B_j) : (W_{a''}, H^{\mathrm{T}}_{b''}) > T1$ in table cells. In the process of generation of the general structure of a sentence by the Cocke–Younger–Kasami algorithm, incorrect versions will be eliminated owing to the impossibility of establishment of semantic-syntactic connections at upper levels of the table and construction of an integral structure. In the case of normal termination, the completely constructed CS structure will contain unions of corresponding correct concepts-meanings, i.e., WordNet synsets, at points with the use of correct semantic-syntactic relations between them. Hence, these relations can be introduced into the semantic WordNet base by adding the corresponding semantic connections between synsets after the completion of the construction of complete and integral CS structure for the sentence.

We now consider a more formal description of the algorithm.

**Input.** The lexico-semantic WordNet base with semantic-syntactic valence vectors attached to its synsets nodes and the input chain of words of a sentence $\omega = a_1 a_2 \ldots a_n \in \Sigma^+$.

**Output.** The table of semantic-syntactic analysis $T$ describing the control space of the syntactic structure of the input sentence; at its points, the space contains word meanings, i.e., WordNet synsets connected by semantic $\alpha$–$\beta$-connections.

**Algorithm**

**Step 1.** Put $t_{i1} = \{A_i \mid A_i$ are synsets that are referred to by $a_i \forall i = 1, \ldots, n\}$. If they are absent (for example, for prepositions, conjunctions, etc.), then, instead of synsets, $t_{i1}$ contains only lexemes that also have own semantic-syntactic valence vectors with the help of which they are connected with semantically significant lexemes at the following stage.

**Step 2.** Assume that $t_{ij}$ are already computed for all $1 \le i \le n$ and all $1 \le j' < j$. Put $t_{ij} = \{A_i$ for some $1 < k \le j$, $\{B_i\} \in t_{ik}$ and $\{C_i\} \in t_{i+k,j-k}$, $\xi(B_i, C_i) = A_i\}$. Since $1 < k \le j$, $k$ and $j-k$ are less than $j$. Thus, $t_{ik}$ and $t_{i+k,j-k}$ are computed before $A_i \Rightarrow^+ a_i a_{i+1} \ldots a_{i+j-1}$.

After this step, it follows from $A_i \in t_{ij}$ that

$$A_i \Rightarrow (B_i, C_i) \Rightarrow^+ (a_i \ldots a_{i+k-1}, C) \Rightarrow \ldots \Rightarrow a_i \ldots a_{i+k-1} a_{i+k} \ldots a_{i+j-1}.$$

**Step 3.** Repeat step 2 until $t_{ij}$ are known for all $1 \le i \le n$ and $1 \le j \le n-i+1$.

We now consider the behavior of the function $\xi(B, C)$. It checks whether points $B$ and $C$ can establish a circular syntagmatic $\alpha$–$\beta$-connection, or the $\alpha$-connection of a linear predicative relation, or $\beta$-connection of a linear predicative relation between themselves.

To determine the possibility of establishment of a circular syntagmatic $\alpha$–$\beta$-connection between points $B$ and $C$, the check is performed by computing the scalar product $(W_B, H_C^{\mathrm{T}})$, where $W_B$ is the vector of the concept-synset that is the semantic meaning of the point $B$ and $H_C$ is the vector of the concept-synset that is the semantic meaning of the point $C$. If $(W_B, H_C^{\mathrm{T}}) \ge T_{\alpha\beta}$, then the function $\xi$ establishes this circular syntagmatic connection, places it at a new point $A$, and computes its new lexical and semantic meaning. This meaning is inherited from the point $B$ as the main point of the pair or, as a result of union, a new concept-meaning is formed ("Black" + "hole" = "Black hole"). The point $A$ assumes the lexical meaning and corresponding synset (or collection of synsets in the case of ambiguities) found by the function $\xi$ as its semantic meaning. To each synset (or synsets) corresponds its (their) semantic-syntactic valence vectors attached to it (them) from different matrices of the tensor language model.

To determine the possibility of establishment of the $\beta$-connection of a linear predicative relation between points, the check is performed by computing the scalar product $(Y_B, Z_C)$, where $Y_B$ is the vector $Y$ of the concept-synset that is the semantic meaning of the point $B$ and $Z_C$ is the vector $Z$ of the concept-synset that is the semantic meaning of the point $C$. If $(Y_B, Z_C) \ge T_\beta$, then the function $\xi$ establishes this $\beta$-connection of a linear predicative relation, places it at a new point $A1$, and computes its lexical and semantic meanings. As a rule, $A1$ inherits both lexical and both semantic meanings from the points $B$ and $C$.

To determine the possibility of establishment of the $\alpha$-connection for a linear predicative relation between points $D$ and $A1$, the check is performed by computing $\sum_{i=1}^{k} X_D[i] * Y_B[i] * Z_C[i]$, where $X_D$ is the vector of the concept-synset that is the semantic meaning of the point $D$; $Y_B$ is the vector of the concept-synset that is the semantic meaning of the point $B$ from $A1$; $Z_C$ is the vector of the concept-synset that is the semantic meaning of the point $C$ from $A1$. If $\sum_{i=1}^{k} X_D[i] * Y_B[i] * Z_C[i] \ge T_P$, then the function $\xi$ establishes the $\alpha$-connection of the linear predicative relation and thereby terminates the complete linear predicative sequence of $\alpha$–$\beta$-connections of the CS of this sentence.

After the algorithm has successfully completed the assembling of the entire control space of a sentence, the completely constructed CS structure at points contains unions of the corresponding correct concepts-meanings, i.e., WordNet synsets, with the use of correct semantic-syntactic relations between them.

Then the stage of traversal of the constructed CS structure of the sentence is performed with transferring the circular semantic connections found between synsets to the WordNet database in the form the corresponding semantic relations between nodes-concepts of the ontology.

TABLE 1. Semantic Cases of Fillmore–Gruber that Describe the Correspondence between Syntactic Patterns and Types of Semantic Relations

| **Argument-Predicate Role Relation** | **Semantic Meaning** | **Syntactic Pattern** | **Example** |
|---|---|---|---|
| Actant or subject | Action initiator, concept of the type of an object | Corresponds to a noun group NG | **Jonn** writes a letter |
| Theme (object) | Object subjected to an action, concept of the type of an object | Corresponds to the noun group NG but at another syntactic position | Jonn writes **a letter** |
| Recipient (a subspecies of an object) | Object in the direction of which an action is performed | Corresponds to noun group PG | Jonn writes a letter **to Mary** |
| Tool | Object with the help of which an action is performed | Corresponds to a prepositional phrase PG with the prepositions "with" and "by" | Jonn writes a letter **with his Parker pen** |
| Style | A manner of execution of an action | It is expressed with the help of adverbs ADV | Jonn writes **easily** |

If, between the CS points containing the synsets $A$ and $B$ as their semantic meanings, a circular syntagmatic $\alpha$–$\beta$-connection is established, then, between $A$ and $B$, the explicit relation "object–property" or "action–property" is written in WordNet.

If, between the CS points containing the synsets $A$ and $B$ as their semantic meanings, the linear predicative $\alpha$-connection is established, then the explicit relation "subject–action" or "actant–action" is written in WordNet between $A$ and $B$.

If, between the CS points containing the synsets $B$ and $C$ as their semantic meanings, the linear predicative $\beta$-connection is established, then, between $B$ and $C$, the explicit relation "action–object" or "action–recipient" or "action–tool," etc. is written in WordNet depending on the use of some syntactic pattern or other to express this $\beta$-connection in the sentence. Theta roles of Fillmore–Gruber [3] (Table 1) describe the correspondence between syntactic patterns and types of semantic relations. Simultaneously with the addition of a collection of semantic connections from one sentence to the ontology, labels reflecting their joint use within the framework of one functional predicate-argument verb scheme are fixed.

After repeatedly processing training text corpora, all implicit semantic connections between synsets expressed by semantic-syntactic valence vectors are explicitly written in WordNet by the addition of the corresponding edges to the graph of the ontology network.

## CONCLUSIONS

This article describes a semantic-syntactic model of natural language; the model is realized with the help of nonnegative factorization of linguistic tensors constructed as a result of frequency analysis of syntactic structures of sentences from training text corpora. Based on the constructed model, an algorithm is developed for the replenishment of ontologies with new semantic relations between nodes-concepts. The method consists of computation of semantic-syntactic valence vectors of ontology concepts and subsequent repeated analysis of texts of training corpora with the use of data structures of the trained model and addition of the found semantic-syntactic relations of predicate-argument type to the ontology database.

**REFERENCES**


1. T. Van de Cruys, “A non-negative tensor factorization model for selectional preference induction,” Journal of Natural Language Engineering, **16**, No. 4, 417–437 (2010).
2. T. Van de Cruys, L. Rimell, T. Poibeau, and A. Korhonen, “Multi-way tensor factorization for unsupervised lexical acquisition,” in: Proc. COLING’2012, Mumbai, India (2012), pp. 2703–2720.
3. C. J. Fillmore, “The case for case,” in: E. Bach and R. T. Harms (eds.), Universals in Linguistic Theory, Holt, Rinehart, and Winston, New York (1968), p. 88.
4. O. O. Marchenko, “A method for automatic construction of ontological knowledge bases. I. Development of a semantic-syntactic model of natural language,” Cybernetics and Systems Analysis, **52**, No. 1, 20–29 (2016).
5. A. V. Anisimov, O. O. Marchenko, and T. G. Vozniuk, “Determining semantic valences of ontology concepts by means of nonnegative factorization of tensors of large text corpora,” Cybernetics and Systems Analysis, **50**, No. 3, 327–337 (2014).
6. G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller, Introduction to WordNet: An Online Lexical Database, http://wordnetcode.princeton.edu/5papers.pdf.
7. D. H. Younger, “Recognition and parsing of context-free languages in time n3,” Information and Control, **10**, No. 2, 189–208 (1967).