Method for Synthesizing the Semantic Kernel of Web Content

Sergey Orekhov, Henadii Malyhon

National Technical University "Kharkiv Polytechnic Institute", Kyrpychova str. 2, Kharkiv, 61002, Ukraine

Abstract

The purpose of forming the semantic kernel of web content is to increase the efficiency of virtual promotion of goods or services [1-2]. It is a message in the virtual promotion marketing channel. To operate with the semantic kernel, a set of research tasks is solved, the first of which is the problem of its synthesis. It consists in the formation of the message itself, adapted to the structure and properties of the virtual promotion channel. The paper proposes the formulation of this problem, the method and algorithm for its solution. The scientific novelty of the work is three facts. This problem is considered for the first time. Also for the first time, a metric for estimating the semantic kernel based on the C-value index is proposed. Then solving this problem allows us to analyze the aging effect of the semantic kernel, which was recently discovered. The proposed algorithm includes two cycles. The first cycle searches for candidates for the semantic kernel among phrases. The second one adds new words to the set of candidates. The final stage compares the composition of the set of candidates with the search queries that exist on the Internet today.

Keywords 1

Semantic kernel, C-value, UML, HTML, Text mining

1. Introduction

The semantic kernel is a message in the marketing channel of virtual promotion [1-2]. Then the synthesis of the semantic kernel is to solve the problem of presenting knowledge about the product in the form of a brief description. This description must be run in the virtual promotion channel. Given the nature of virtual promotion, you can determine the following conditions that must be met when forming a semantic kernel.

First, virtual promotion is based on the maximum use of Internet technologies. The latter require the presentation of any information or data in the form of HTML (XML) tags, JSON constructs or RDF schemas. Sometimes we can find information in the form of program code in Javascript, PHP, Python and others. But in this paper we will start from HTML code as the most universal option using the Javascript programming language and relevant frameworks.

Second, from the point of view of search engine optimization, the semantic kernel is a database of keywords, word forms and morphological forms that most accurately characterize the type of activity, product or service that promotes a given web resource. Therefore, the semantic kernel should be considered as a set of key word forms presented in HTML format. It is likely that these keywords need to be highlighted with special HTML tags [3].

Third, search engines also store various word forms in their database and have data on what and how many word forms users request on the Internet. That is, we must take into account in the semantic kernel of the most popular word forms according to the search engine version.

Fourth, the number of words in the semantic kernel is limited in practice. Users are more likely to enter short search queries that match some semantic kernel. We will assume, using the conditions of

COLINS-2022: 6th International Conference on Computational Linguistics and Intelligent Systems, May 12–13, 2022, Gliwice, Poland
EMAIL: sergey.v.orekhov@gmail.com (S. Orekhov); gmalygon@gmail.com (H. Malyhon)
ORCID: 0000-0002-5040-5861 (S. Orekhov); 0000-0001-5448-2488 (H. Malyhon)

the method of analysis of hierarchies, that the number of keywords is limited by the power of a small sample, i.e. ten positions per kernel.

In general, the task of presenting knowledge is formulated as follows: to formalize some subject area of knowledge using a conceptual scheme. Typically, such a scheme includes a data structure that brings together all relevant classes of objects and the relationships between them, as well as the rules (theorems and constraints) that exist in a given subject area.

We will assume that we have as a data structure - a semantic network, where the vertices are keywords. Such keywords can belong to one of three classes: concept, object or action, as suggested in [4]. Then in the paper it is suggested to consider that the semantic kernel is a semantic network of propositional type [5]. Keywords such as concept and object are considered network nodes. These two classes form the rules that describe the knowledge of the product in the format: what, when, where [6]. That is, what we sell when the product is available and where you can buy it. In the same form it is possible to make out and the description of need of the buyer. Using these assumptions, we build a mathematical model of the semantic kernel as a model of the text [7-8] about the product or need.

2. Problem statement

As input we have a text document *D*, which can be written in any language and presented as an HTML construct or plain text:

$$D = \{s_1,...,s_i,...,s_n\}, \qquad (1)$$

where $s_i = \{w_{i1}, sep'_{i1}..., sep'_{ib}, w_{im}, sep_{i1}..., sep_{ik}\}$ - it is a sentence that is part of document *D*. We believe that there will be *n* sentences in the document.

The i-th sentence consists of words w_{ij} and delimiters sep'_{ib} and sep_{ik}. The number of words in a sentence will be *m*. We have two sets of delimiters sep'_{ib} and sep_{ik}. The first set sep'_{ib} combines the characters that separate the words in the sentence, and the set sep_{ik} is the characters at the end of the sentence.

In our case, we will assume that the set of characters sep_{ik} is unimportant, so we will not take it into account in the future. And the set of characters sep'_{ib} will be transformed into a set of connections between nodes of the semantic network [5]. As shown in [4], we have three types of connections: "is", "is a part", "is a type". Then the set $sep'_{ib} = \{isa, apart, typeof\}$.

Each word w_{ij} has its own morphological paradigm, or rather token. According to [7-8], we consider that a keyword unites all forms of one word and its different meanings according to the dictionary, for example, the Ukrainian language. It makes no sense to take into account the part of speech for each word in our case, so we will further consider the concept of the lemma of the word wf_{ij}. Lemma is a normal or infinitive form of a word recorded in a dictionary:

$$wf_{nf} = \{w, case, n\},\ wf_{nf} \in Wf\ , \qquad (2)$$

where *n* is the singular and *case* is the nominative case of the word.

Then let there be a function of normalization or lemmatization of words: $DNF : Wf \rightarrow Wf_{nf}$. Thus, the semantic kernel will include words (2) from document *D*, which were lemmatized from sentences (1).

Let's move on to identifying candidates (words) that we will include in the semantic kernel. Candidate is a word or phrase that meets the criteria and is potentially a term of a specific subject area, or rather describes a product or need that it covers. Let *P* be a set of candidates for the semantic kernel. The elements of this set are words w_i and phrases p_j:

$$P = \{w_1,...,w_o, p_1,..., p_r\},$$

where o is the number of candidates for words, and r is the number of candidates for phrases. Next, define the structure of the phrase:

$$p_j = \{w_{j1},..., w_{jc} \big| c = \overline{1,z}\},$$

where j is the number of the phrase, $j = \overline{1,r}$, but c is the number of the word in the phrase p_j, z is the maximum number of words in the phrase. It usually is equal to ten, that is $z \le 10$. According to empirical data, it is advisable to consider. Although the most likely limitation is $z \le 5$.

In the paper based on typical models of texts without taking into account the properties of the language in which the document D is composed, we describe the construction of the semantic kernel synthesis function as an algorithm that includes two stages:

$$DP : D \to P. \tag{3}$$

In the first stage using function (3) the set P of candidates for the kernel from document D is formed. In the second stage the final list of words or phrases inside the kernel is formed by filtering and ranking. Since the power of the set (1) can be any, we are dealing with large text data. It is advisable to use soft calculations or mathematical statistics to process them [9-11].

In the paper we propose to use the statistical metric M in order to compare the element of future kernel:

$$F_m : P \to M, M \in R, M \ge 0.$$

Having all the prerequisites, we can formally formulate the problem of semantic kernel synthesis.

Given: $TD = \{d_1,..., d_{DOC}\}$ - a text body that describes the product and (or) the need that it covers. *DOC* is the number of web content documents submitted for kernel synthesis. It is necessary to find a set of pairs: $T = \{(p, m)_V\}$. Each pair is a word or phrase and the value of a statistical metric. V is the number of words or phrases in the *TD* text box.

The set T must be ranked in descending order of the metric M. Thus, the closer to the beginning in the final set T is the candidate p, the more likely it is that it is part of the semantic kernel. This fact does not contradict the similar approach outlined in [4].

Consider a metric that allows us to evaluate a word or phrase. We will take the *C-value* method as a basis [7-8]. This method is based on the use of such statistical metrics as the frequency of phrases in the text. But the *C-value* metric also takes into account the length and nesting of the candidate.

Nested terms are a concept that is in the input text separately or as part of other concepts [7-8]. Then this metric is calculated by the following formula:

$$M = \begin{cases} \log_2 |a| \cdot f(a), uninvested \\ \log_2 |a| (f(a) - \dfrac{1}{P(T_a)} \sum\limits_{b \in T_a} f(b)), invested \end{cases}. \tag{4}$$

where a - candidate to the kernel, $|a|$ - length or number of words, $f(a)$ - frequency of appearance of the candidate a, T_a - set of candidates that include the word a, $P(T_a)$ - number of candidates to T_a, $\sum f(b)$ - sum of frequencies of appearance of candidates $b \in T_a$ that include a. That is, a is a nested candidate in the phrase b.

The formula makes it possible to draw the following conclusion. The longer the term a, the greater the value of its metrics. This has the following meaning. Longer terms in the text will be less than short ones. Accordingly, the probability of occurrence of the term b in the number of f positions is less than the probability of the occurrence of the term a in the number of f times, provided that $|a| < |b|$. For the same reason, it can be concluded that the phrase b is more likely to be a term than a. In addition, this method is designed with the fact that the higher the number of terms T_a that include a, the greater the degree of independence a.

Thus, the solution of the problem of semantic kernel synthesis describes the construction of word-metric pairs. We present a method of obtaining such pairs.

3. Method

We verbally describe the method of semantic kernel synthesis as an improvement of the algorithm, which was proposed in [4]. The *TD* text body is supplied as input. Within the developed method there are two cycles. The first cycle operates with sentences as phrases. The second cycle works with selected sentences and selects words from them.

Consider the first cycle and its algorithm.

Step 1. We take for processing the i-th document of the *TD* case. From all sentences of the document D_i, $i=\overline{1,DOC}$ we define words w_{ij} and delimiters of type sep'_{ib}, and delimiters of type sep_{ik} and we delete sep_{ik} completely.

Step 2. Construct table 1, which accumulates a list of candidates for the semantic kernel, i.e. the set *P* is formed. As candidates, we enter all the sentences D_i, $i=\overline{1,DOC}$.

Table 1

Candidates-phrases for entry into the semantic kernel of web content

№	Candidate (phrase)	Frequency	C-value	Candidate type
-	-	-	-	-

For each candidate, the frequency of occurrence in the text box is estimated. Next, go to step 1 to process the next document D_{i+1}. Follow these two steps to complete Table 1 fully with all the sentences available in the text box.

Step 3. Perform the calculation of statistical metrics (4) for each candidate in table 1. Next, we sort the rows of the table in descending order of the C-value metric. Taking into account the conditions defined above for the text model, we choose $|P_D|$ the first terms as candidates for the semantic kernel. The power of the set P_D is defined as the initial condition before the start of the first cycle. The paper proposes to set this value to ten.

Step 4. For each candidate from table 1 set its type according to the concept of "4P". To do this, answer three questions: what, where, when. The "what" question describes the name of the product or service. The question "where" is the geography of a good or service, that is, any word that describes a place on a map. But the question of "when" is responsible for words that tell us about time or time intervals. In the simplest case, the semantic kernel includes three words: product name, place of sale and time interval. This completes the first cycle and we have the set P_D, which consists only of phrases $P_D=\{p_1,...,p_r\}$.

Consider the second cycle. Its purpose is to add to the set P_W individual words $\{w_1,...,w_o\}$ that are part of the semantic kernel. We believe that $P=P_W\cup P_D$. The following steps are suggested.

Step 1. We take for processing the i-th document of the *TD* case. From all sentences of the document D_i , $i=\overline{1,DOC}$, we define words w_{ij} and delimiters of type sep'_{ib}, and delimiters of type sep_{ik} and we delete sep_{ik} completely.

Step 2. Construct table 2, which accumulates a list of candidates for the semantic kernel, i.e. the second part of the set *P* is formed - P_W. As candidates, we enter all the words w_{ij}. Determine the frequency of occurrence of the word in the text corpus *TD* as a whole.

Table 2
Candidates-words for entry into the semantic kernel of web content

№	Candidate (word)	Frequency	C-value	Candidate type	$\lvert T_a \rvert$ value
-	-	-	-		-

Step 3. Perform the calculation of statistical metrics (4) for each candidate in table 2. Next, we sort the rows of the table in descending order of the C-value metric. Taking into account the conditions defined above for the text model, we choose the first terms $\lvert P_W \rvert$ as candidates for the semantic kernel. We also calculate the value $\lvert T_a \rvert$ as the number of occurrences of words from set P_W to set of phrases P_D. And again we will re-sort table 2 on condition of increase in value $\lvert T_a \rvert$.

Thus, we have two sets of candidates to enter the semantic kernel, completing the second cycle.

The final stage. The resulting set $P = P_W \cup P_D$ is lemmatized. The set of phrases P_D is represented in the form of a semantic network according to the algorithm presented in [4]. Then to the semantic kernel we choose those phrases in which the maximum number of words of candidates from the set P_W on one side and which are in the filter of the search engine in the first place.

It is also advisable to use metric (4) to evaluate the findings of the search engine. It will also allow the formation of "candidate-metric" pairs for a more accurate choice of components of the semantic core.

The resulting set P'_D will be the semantic kernel proved by a set P_W and search engine web service. Consider the algorithmic representation of the method proposed in the dissertation

4. Algorithm

Using the UML [13-15], an action diagram was prepared to describe the algorithm that implements the method of synthesis of the semantic kernel of the web content - Figure 1. The diagram shows two cycles of forming: first a subset of phrases and then a subset of words that are part of the semantic kernel.

The question of typing words and phrases in automatic mode is open. Such a filter can be implemented, for example, using the approach described in [6], where the authors of this paper was directly involved.

The next problem is with the search engine. All accumulated candidates from the set P_D must be submitted to the web service input to calculate the metric value (4).

To build a semantic network, the method described in [4] should be used. It allows us to generate a graph in automated mode.

The final choice of the components of the semantic kernel is based on the values of the metric (4) given in Table 2, as well as on the basis of calculating the rules of the type "is a". The last indicator allows us to estimate the number of rules in which a word or phrase that is a candidate for entry into the semantic kernel.

We believe that the more rules in which a word or phrase is involved, the higher its value for the semantic kernel [4].

5. IT solution

According to the classical theory of software systems design [16-17] at the first stage it is necessary to form a set of potential software components that will be part of the information system. To do this, we need to create software requirements for future software components.

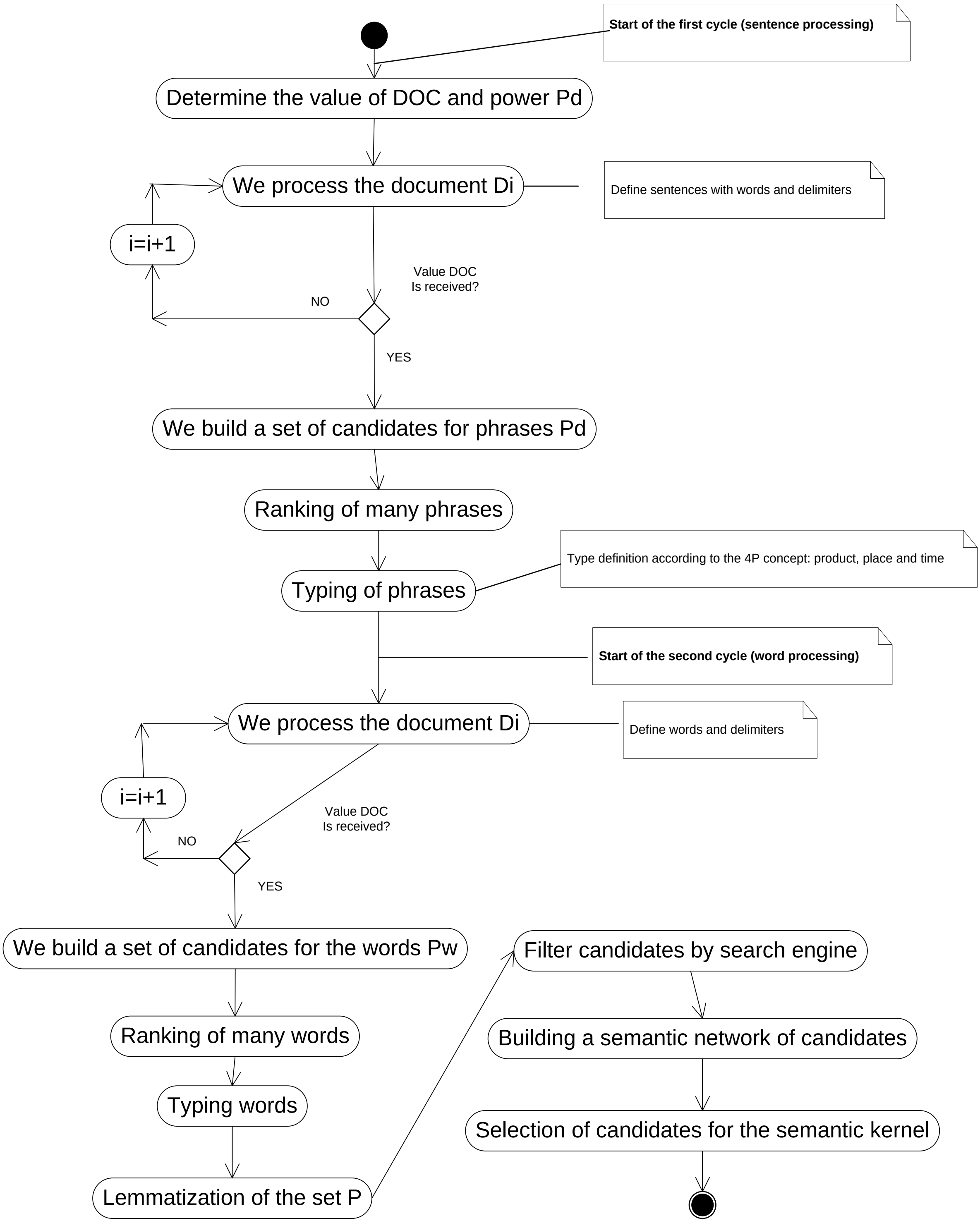
Start of the first cycle (sentence processing)
Determine the value of DOC and power Pd
We process the document Di
Define sentences with words and delimiters
i=i+1
Value DOC
Is received?
NO
YES
We build a set of candidates for phrases Pd
Ranking of many phrases
Type definition according to the 4P concept: product, place and time
Typing of phrases
Start of the second cycle (word processing)
We process the document Di
Define words and delimiters
i=i+1
Value DOC
Is received?
NO
YES
We build a set of candidates for the words Pw
Ranking of many words
Typing words
Lemmatization of the set P
Filter candidates by search engine
Building a semantic network of candidates
Selection of candidates for the semantic kernel

Figure 1: Activity diagram

We take into account the action diagrams for algorithmic support of the proposed methods for solving the problem, shown in Figures 1.

According to the SWEBOK concept [17], we have six quality factors that fix non-functional requirements. We will use the following approach in the analysis of quality factors (Figure 2). It is necessary to divide these factors into two groups according to two basic roles in the software development lifecycle: user and developer (Figure 2). The first group includes such factors as functionality, reliability and ease of use. The second group will include efficiency, maintainability and tolerability. It is impossible to fulfill all six factors, so it is necessary to focus on at least two main factors from one and the other group of factors. The most important for the potential user of this information technology are such factors as functionality and portability. To enhance the quality of the software, a description should be added according to the usability factor.

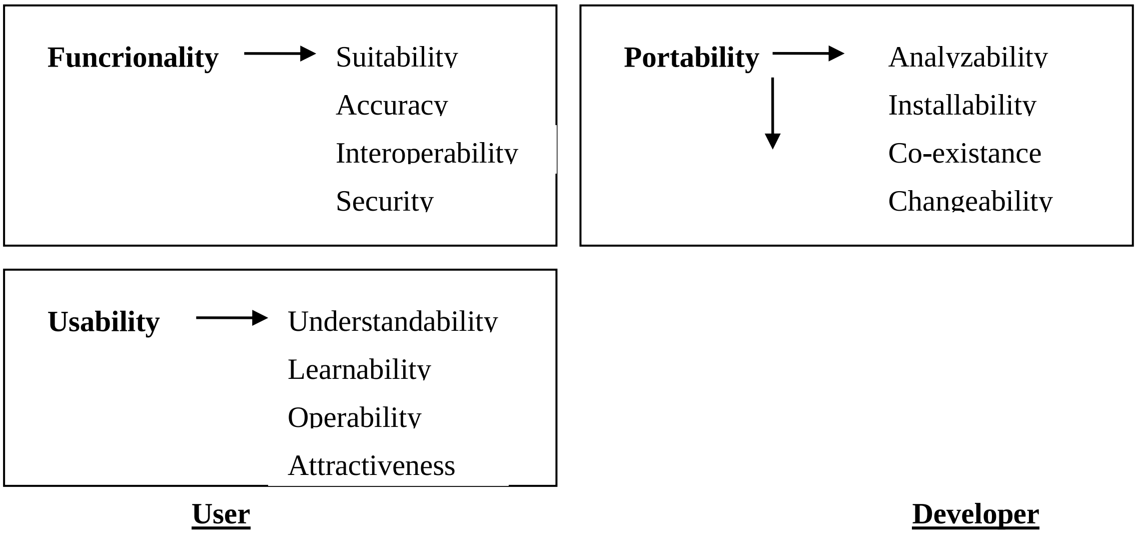

Figure 2: Factor of software quality

The functionality of the projected information technology is based on the idea of implementing a cycle of situational management, which is triggered by the synthesis of set of semantic kernels. Next, having the first semantic kernel, we can evaluate its effectiveness. According to this assessment, it is decided whether or not to change this semantic kernel in the promotion channel. If a change is required, the next kernel is selected from the set available or a new set of kernels is generated. The possibility of completing the cycle according to a given set of stop criteria is also analyzed.

Thus, the choice of these three factors is due to the fact that information technology from the point of view of the end user should cover a maximum of actions (functions) to solve the problem of situational management. This technology must be effectively integrated into existing information environments both at the data level and at the software component level. Also, such integration should be clear, workable and convenient, i.e. easy to use. However, other factors for this case of software systems design are of little importance. For example, the efficiency factor is primarily unimportant from the point of view of time behavior, because the decision-making time is at least thirty days. The reliability of this technology is due to the reliability of the metric values of the semantic core, which does not depend on the requirements for the software itself. From the point of view of the maintainability factor, no further analysis of the technology is required to change it. Such technology either has an effective effect on the promotion of goods or not. That is, in the case of a negative effect, this technology is simply no longer used.

The paper proposes to form the following software components: semantic kernel and semantic network – figures 3-4. All components are being designed according to the algorithm of problem solving.

The semantic kernel component (Figure 3) is designed according to the MVC template [18-19]. The elements of this component are all components of the algorithm for the synthesis of the semantic kernel (Figure 1): sentence, words, delimiters and HTML content. Also it is necessary to present HTML content or semantic kernel by semantic network (Figure 4). The controller class is also used to start the synthesis procedure. Visualization of the synthesis result is carried out by methods of the KernelView class.

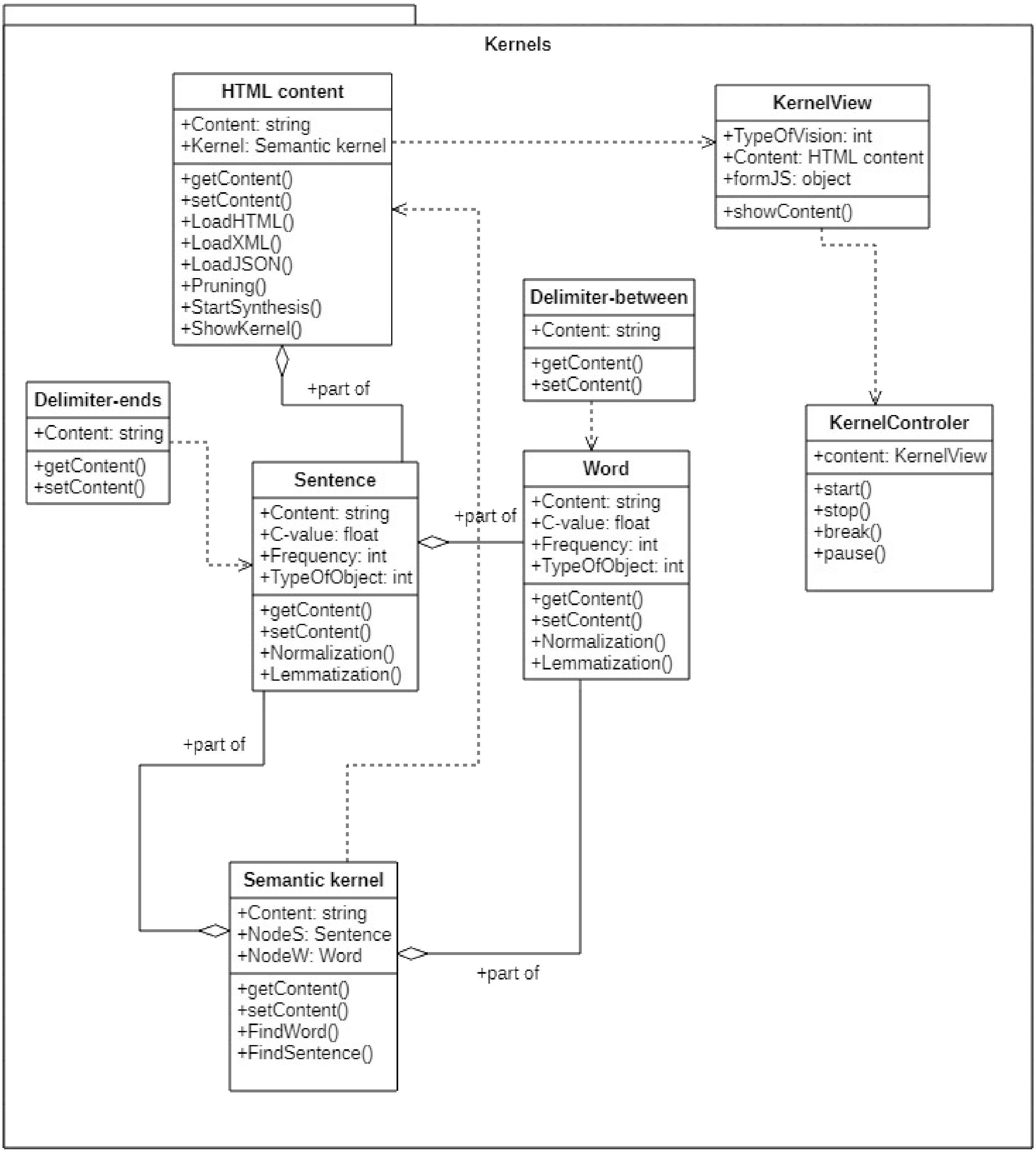

Figure 3: Program package – semantic kernel

The second component is needed to describe the classes for the software implementation of the method of presenting phrases from web content in the form of a semantic network. This makes it possible to evaluate the semantic kernels generated in terms of artificial intelligence methods, in particular in terms of the number of rules.

In order to define program realization of algorithm the sequence diagram was proposed – figure 5.

This diagram (figure 5) shows the order in which different class methods are called in software packages by the end user, but exactly according to the synthesis algorithm.

The sequence of calls describes two cycles of synthesis of semantic nuclei, which can be implemented both separately and sequentially. The first cycle searches for semantic nuclei in the form of phrases, and the second cycle searches for candidate words. Experiments with the semantic cores of various websites have shown that it is advisable to strengthen the cycle of searching for phrases with a cycle of searching for individual words. And check the result with the help of search engine web services, such as Google API.

6. Results

The problem of semantic kernel synthesis and the method of its solution considered in this section is a scientific novelty. At present, there are only verbal descriptions of this problem [3, 20-21], and especially methods of its solution [4]. A distinctive feature of the proposed approach is that the presented statement of the problem is based on the model of text, as well as individual stages, in particular typing, on syntactic models of grammar of the immediate components.

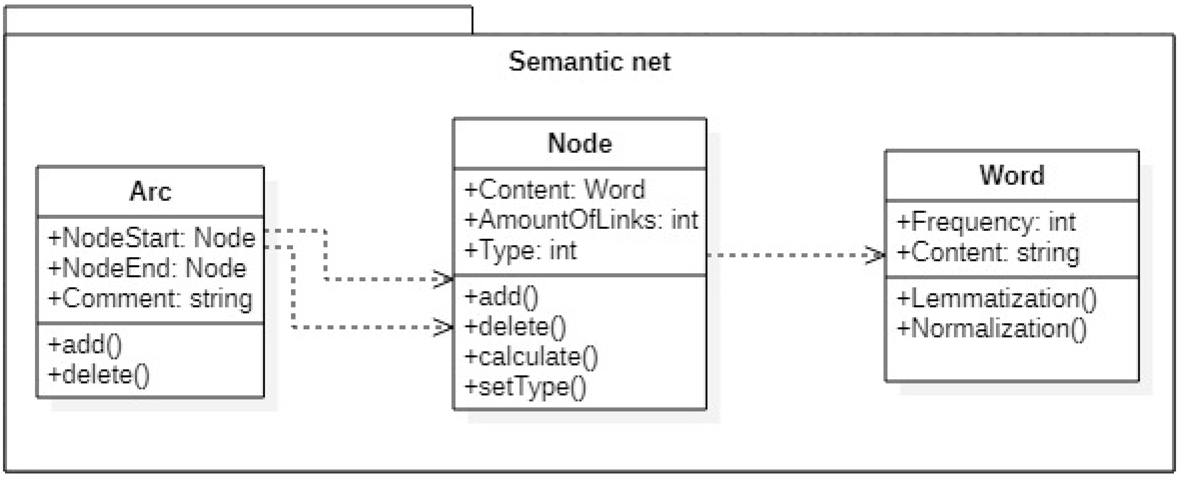

Figure 4: Program package – semantic net

The semantic network is also used to finally select the semantic kernel components to estimate the number of rules that connect the kernel elements and an additional filter based on the search server database.

The last point is especially important because modern verbal approaches to the formation of the nucleus are based on this procedure [3]. But the very use of the principles of systems analysis allows us to operate with the term "synthesis" and solve this problem.

In addition, the proposed algorithm immediately forms at least several variants of the semantic kernel, so when you change the kernel to a new one, you can choose the next option. This approach guarantees the application of the ideology of situational management of the semantic kernel, when the transition from one situation (from one kernel) to another. That is, the set P'_D includes possible variants of the semantic kernel.

7. Conclusions

Among the obtained results of scientific novelty are the following:

1. For the first time the problem of synthesis of the semantic kernel of web content is formulated and the method of its solution from the standpoint of system analysis and methods of artificial intelligence is described.
2. For the first time the metrics for estimation of a semantic kernel on the basis of estimations of the text case of web content are offered.
3. The method of presenting the semantic kernel as a semantic network was further developed

The direction of further research will be the implementation of this algorithm based on Javascript libraries [22]. In particular, it is planned to implement a semantic kernel synthesis algorithm based on the NodeS library [23].

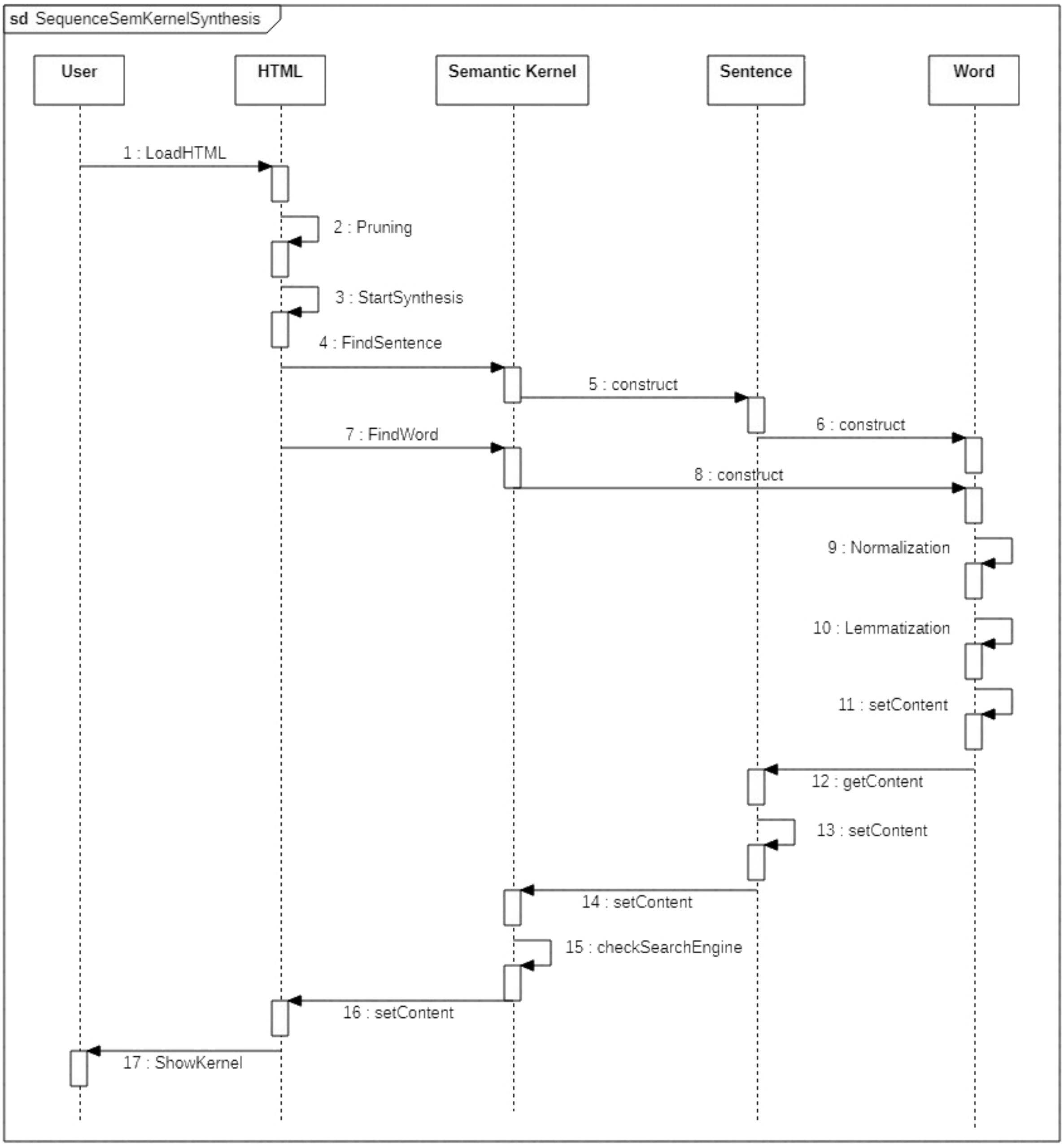

Figure 5: Sequence diagram

8. References

[1] S. Orekhov. Technology of virtual product promotion. Volume 3 of Computer Systems and Information Technologies, 2021, pp. 52-58.

[2] S. Orekhov, H. Malyhon. Virtual promotion knowledge management technology. Bulletin of the National Technical University KhPI. Series: System analysis, control and information technology. Volume 1(3) of Collection of Scientific papers. NTU KPI, Kharkiv, 2020, pp.74–79.

[3] J. Rowley. Understanding digital content marketing. Volume 24 (5-6) of Journal of Marketing Management, 2008, pp. 517–540.

[4] M. Godlevsky, S. Orekhov, E. Orekhova. Theoretical Fundamentals of Search Engine Optimization Based on Machine Learning. Volume 1844 of CEUR Workshop Proceedings, 2017, pp. 23–32.

[5] J. Dean. Big Data, Data Mining, and Machine Learning. Value Creation for Business Leaders and Practitioners, John Wiley & Sons Inc., USA, 2014.

[6] S. Orekhov, H. Malyhon, T. Goncharenko, I. Liutenko. Using Internet News Flows as Marketing Data Component. Volume 2604 of CEUR Workshop Proceedings, 2020, pp. 358–373.

[7] C.C. Aggarwal, C.X. Zhai. A survey of text classification algorithms. Mining Text Data, Springer Science-Business Media, LLC, 2012, pp. 163–222.

[8] A. Khan, B. Baharudin, L. Lee, K. Khairullah. A Review of Machine Learning Algorithms for Text-Documents Classification. Volume 1(1) of Journal of advances in information technology, 2010, pp. 4–20.

[9] B.R. Prasad, S. Agarwal. Comparative Study of Big Data Computing and Storage Tools: A Review. Volume 9(1) of International Journal of Database Theory and Application, 2016, pp. 45–66.

[10] F. Abuqabita, R. Al-Omoush, J. Alwidian. A Comparative Study on Big Data Analytics Frameworks, Data Resources and Challenges. Volume 13(7) of Modern Applied Science, 2019, pp. 1–14.

[11] S. Alkatheri, S.A. Abbas, A. S. Muazzam. A Comparative Study of Big Data Frameworks. Volume 17(1) of International Journal of Computer Science and Information Security, 2019, pp. 66–73.

[12] J. Nereu, A. Almeida, J. Bernardino. Big Data Analytics: A Preliminary Study of Open Source Platforms, Proceedings of ICSOFT, 2017, pp. 435–440.

[13] A. W. Scheer, M. Nuttgens. ARIS architecture and reference models for business process management. Business process management, Springer, Berlin, 2000.

[14] IFEF0: Integration Definition for Function Modeling, National Institute of Standards and Technology, Gaithersburg, 1993.

[15] B. Rumpe. Agile modeling with UML. Springer, Germany, 2017.

[16] J. Ousterhout. A philosophy of software design. Yaknyam Press, USA, 2018.

[17] P. Bourque, R. Fairley. SWEBOK. Guide to the Software Engineering Body of Knowledge. Version 3.0, IEEE Computer Society, 2019.

[18] E. Gamma, R. Helm, R. Johnson, J. Vlissides. Design Patterns. Elements of Reusable Object-Oriented Software, Addison-Wesley, USA, 1995.

[19] T. Winters, T. Manshreck, H. Wright. Software Engineering at Google. Google LLC, USA, 2020.

[20] U. Sharma, K.S. Thakur. A Study on Digital Marketing and its Impact on Consumers Purchase. Volume 29(3) of International Journal of Advanced Science and Technology, 2020, pp. 13096 – 13110.

[21] J. García, D. Lizcano, C. Ramos, N. Matos. Digital Marketing Actions That Achieve a Better Attraction and Loyalty of Users: An Analytical Study. Volume 11(130) of Future Internet, 2019, pp. 1-16.

[22] C. Heilmann. Beginning JavaScript with DOM Scripting and Ajax: From Novice to Professional, Apress, USA, 2006.

[23] ReactJS. Notes for Professionals. Stack Overflow, USA, 2019.