The approach to unification of data extracted from social networks

Vadim Moshkin[a] and Nadezhda Yarushkina[a]

[a] *Ulyanovsk State Technical university, Severny Venets str., Ulyanovsk, 432027, Russian Federation*

Abstract

The work presents an ontological model for the unification of data profiles of different social networks. This model avoids data redundancy by including contextual information in annotations to ontology relations. In addition, an approach to information retrieval using syntagmatic patterns in the formation of a database tree of posts of social network users is proposed. The article also presents the results of experiments with data from the social network Facebook confirming the effectiveness of the proposed models and algorithms.

Keywords 1

Ontology, social network, syntagmatic pattern, knowledge base

1. Introduction

Analysis of social data currently plays a significant role in many areas and requires appropriate tools. Existing solutions use social networks as a tool for collecting large amounts of important information. These solutions are based on tools and technologies for working with big data.

First of all, specialists from research centers and companies around the world use data from social networks to model social, economic, political and other processes from the personal to the state level to develop mechanisms for influencing these processes, search for the necessary knowledge, and create analytical and business applications and services.

Several methods have been implemented for obtaining representative sets of social networks users: Currently, the most effective methods for collecting information from social networks are the following approaches:

- width traversal method [1];
- the "forest fire" method [2];
- Metropolis-Hastings method [3].

Social networks are a powerful marketing research tool, as users voluntarily publish information about themselves, their views, interests and preferences. Currently, most social media aggregators analyze data related to specific brands. Aggregators collect information about actions on the company's page in social networks, look for brand mentions and help make business decisions based on this data.

There is also a set of tools that provide data collection from profiles of a particular user. These software systems are designed to analyze brand engagement and popularity (through the rating of the relevant community or user profile).

The most popular social media analysis tools:

- YouScan [4] - the first system for professional monitoring of Russian-language social media. YouScan tracks the mentions of brands, products, competitors in blogs, forums, social networks (Facebook, VKontakte, Twitter, YouTube) and presents the monitoring results in a convenient analytical interface with teamwork functions.

Russian Advances in Fuzzy Systems and Soft Computing: selected contributions to the 8-th International Conference on Fuzzy Systems, Soft Computing and Intelligent Technologies (FSSCIT-2020), June 29 – July 1, 2020, Smolensk, Russia
EMAIL: v.moshkin@mail.com; jng@ulstu.ru.
ORCID: 0000-0002-9258-4909; 0000-0002-5718-8732.

- ForsMedia [5] - a system for extracting structured and unstructured information about existing and potential customers from social networks.
- BrandAnalytics [6] - a software system that tracks brand mentions in social networks, blogs, forums, review sites, instant messengers and online media.
- Feedot [7] and Hootsuit [8] - services that provide the user with their profiles in various social networks.

Social network aggregators and similar software services store information obtained from various social networks, mainly in a relational database. The relational database architecture is optimal when the data objects do not have diverse and multiple relationships. Therefore, storing data, for example, the Facebook network with 1 billion users and 10 billion relations between them, in a relational database is not effective in optimizing the data search space.

The graph model of the knowledge base is effective when applying complex and flexible queries. In this regard, the task of developing models, algorithms and software tools for storing knowledge of structurally complex information from social networks is relevant.

2. A model for the unification of data extracted from different social networks

The main problem of collecting data from different social networks is the structural difference in the form of data storage. Therefore, the main task is to develop unification algorithms for the extracted data, adapted to the features of the presentation of structured and unstructured knowledge in each media resource [9].

The extraction and unification of data from social network profiles, according to the developed approach, includes several stages:

1. The selection of a set of social networks from which information will be extracted.
2. Design and development of methods for extracting information from each social network. Methods depend on the availability of APIs, access rights and data access policies for each social network.
3. Search for profiles of one person in various social networks.
4. Automated data collection from social network profiles.
5. Unification of the extracted data to a single model.
6. Translation of data into subject ontology.

Formally, the ontology model of social network profiles is:

$$O^{SN} = \{N^{SN}, R^{SN}, F^{SN}\},$$

where N^{SN} is the set of nodes (objects and classes) of the ontology;
R^{SN} is the set of ontology relations, $R^{SN} \in N^{SN} \times N^{SN}$;
F^{SN} is the set of ontology interpretation functions (axioms);

$$N^{SN} = N^{B} \cup N^{COM} \cup N^{DOM};$$

where $N^{B} = \{n_1^{B}, n_2^{B}, \dots, n_m^{B}\}$ – nodal objects are users of the social network
$N^{COM} = \{n_1^{COM}, n_2^{COM}, \dots, n_l^{COM}\}$ - internal objects are the essence of social networks.

The translation of the elements of various social networks into the elements of the set N^{COM} is presented in table 1.

Table 1

Translation of elements of social networks into objects of the ontological model

N^{COM}	VKontakte	Twitter	Instagram	Facebook	Ok.ru
Social Network	URL	URL	URL	URL	URL
Group	Group	-	-	Group	Group
Post	Post	Twit	Photo	Post	Post
Comment	Comment	Comment	Comment	Comment	Comment
Attachment	Attachments	Attachments	Tags, links	Attachments	Attachments

- $N^{DOM} = \{n_1^{DOM}, n_2^{DOM}, \dots, n_k^{DOM}\} -$ are objects of the material world: military unit, school, university, city, state, music group, book, etc.).

Relation Types:

$$R^{SN} = R^{OP} \cup R^{DTP} \cup R^{CONT}$$

- $R^{OP} = \{r_1^{OP}, r_2^{OP}, \dots, r_s^{OP}\}$ are Object Properties (hasFriend, hasFollower etc.), i.e. relations between objects of ontology;
- $R^{DTP} = \{r_1^{DTP}, r_2^{DTP}, \dots, r_h^{DTP}\}$ are DataType Properties, i.e. relations between ontology objects and built-in type values (Boolean, String, Number). Examples of relations of the proposed model are presented in table 2.
-

Table 2

Some relations of the ontological model for representing social network data

№	Profile field	Domain	Relation	Range
			Datatype Properties	
1	Name	User	hasName	String
2	Last name	User	hasLastname	String
3	Date of birth	User	hasDateOfBirth	Date
			Object Properties	
4	School	User	wentToSchool	School
5	City	User	livesIn	City
6	Audio	User	hasAudio	Audio
7	Audio writer	Audio	hasAuthor	Person/ User
8	Post	User/ Group	hasPost	Post
9	Has friend	User	hasFriend	User
10	Has follower	User	hasFollower	User

- R^{CONT} are annotation properties that define the context.

Two types of context were identified within the proposed model:

- R^{CSN} is the annotation relation in which social network data is stored.
- R^{T} is the annotation relation which stores the time period during which this relationship was relevant. Wherein

$$(\forall r_i \in R^{OP}, R^{DTP}), \exists\, r_i^{CONT} \in R^{CONT}, r_i^{CONT} = \{\, r_i^{T}, r_i^{CSN}\}.$$

Schematically, the consideration of the temporary context and the context of the data source is shown in Figure 1.

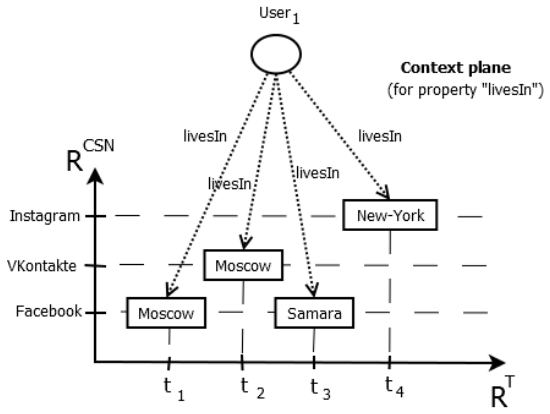

Figure 1: The temporary context and the context of the data source

Historicity of data is maintained through the use of a temporary context. There is no data redundancy when the information from the profiles of one person in different social networks coincides (Fig. 2) due to the storage of contextual information in annotations to relations.

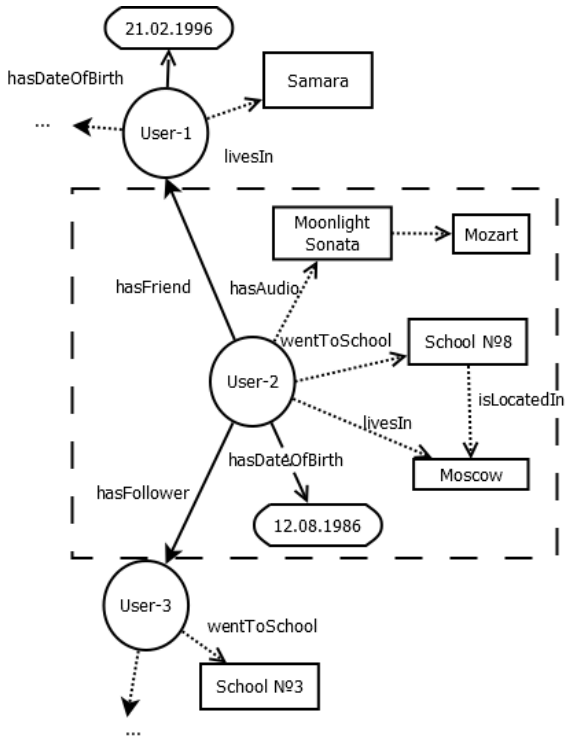

Figure 2: A fragment of the ontology of data from social networks

Selected objects and ontology classes store data downloaded from most existing social networks. The unified ontological representation of data increases the efficiency of processing, analysis and data retrieval.

Separately, when extracting and unifying data from social networks, text elements such as posts and comments are processed. The analysis of unstructured information helps to determine the semantics and sentiment of the groups and profiles of users of social networks. A semantic structuring of text resources is necessary for the effective search and analysis of these resources by extracting semantic trees from large fragments of texts.

3. The algorithm for extracting the semantic tree from the text resources of social networks

3.1 Building a parse tree

The extraction of knowledge from unstructured resources is aimed at finding information that describes a certain area of knowledge defined by the data structure. The semantic tree is a formal model of the subject area, in the form of a graph of terms and semantic relationships and summarizes the hierarchical data structure [10].

Extracting a semantic graph from text data from social networks simplifies the search process in large data packages. Hence, to construct a semantic graph of sentences of a specific text fragment, it is necessary:

- Extract a parse tree from each sentence of a text fragment;
- Merge parse trees;
- Translate syntactic graph into semantic (ontology).

Currently, there are several tools for parsing texts in natural language, for example, [11] [12] [13] [14]. As part of this project, the tools developed as part of the AOT [15] project were modified. A semantic graph can be formally represented as a directed graph:

$$G^{Sem} = (W^{Sem}, E^{Sem}),$$

where $W^{Sem} = \{W_1^{Sem}, W_2^{Sem}, \dots, W_w^{Sem}\}$ is the set of nodes of the semantic graph. Each node of a semantic graph is a linguistic unit obtained by translating nodes of a syntactic graph into a semantic one;

$E^{Sem} = \{W_i^{Sem}, W_k^{Sem}\}$ is the set of arcs of the semantic graph in which $W_i^{Sem}, W_k^{Sem} \in W^{Sem}$.

The result of the parsing is the selection of syntactic groups and fragments. These groups are extracted from unstructured text resources using syntax rules and the construction of a syntactic graph [16].

A feature of this approach is the presentation of many syntax rules in the form of an NLP ontology [17]. Formally a NLP-ontology is:

$$O^{NLP} = (M^{NLP}, N^{NLP}),$$

where M^{NLP} is the morphological characteristics of the M^{NLP} NLP-ontology O^{NLP}, defining the characteristics of the group relative to the groups in which it belongs;

N^{NLP} is the set of rules for constructing syntax groups. Rules are written in SWRL notation [18]. The result of applying the rules is the parse tree G^{Synt}:

$$G^{Synt} = (W^{Synt}, E^{Synt}),$$

where $W^{Synt} = \{W_1^{Synt}, W_2^{Synt}, \dots, W_w^{Synt}\}$ is the set of nodes of the parse tree, which can be represented as terms or syntactic groups;

$E^{Synt} = \{W_i^{Synt}, W_k^{Synt}\}$ is the set of arcs of the parse tree in which $W_i^{Synt}, W_k^{Synt} \in W^{Synt}$.

From here, the syntax group is determined by the following parameters:

- The type of the syntax group (for example, HOMO_ADJ - is a group of homogeneous adjectives).
- The main syntactic subgroup (for example, for the group type "NOUN-NUM" the main group is a noun).
- Grams of the syntax group are the morphological characteristics of the M^{NLP} NLP ontology O^{NLP}.

One rule N_i^{NLP} NLP ontology O^{NLP} forms one type of group. Each i -th rule N_i^{NLP} of the NLP ontologyO^{NLP} receives many morphological features as input $M_{w_i^{Synt}}^{NLP} \subseteq M^{NLP}$. If the set of morphological characters $M_{w_i^{Synt}}^{NLP}$ corresponds to the set of rules $M_{N_i^{NLP}}^{NLP} \subseteq M^{NLP}$, the group is assigned a specific type specified in the i-th rule N_i^{NLP} of the NLP ontology O^{NLP}.

The rules of the NLP-ontology O^{NLP} are applied in the order determined by the expert linguist and combine the input group with the groups located to its right. The order of syntactic rules corresponds to the order of building groups: from smaller to large.

The syntax rule N_i^{NLP} of the NLP-ontology O^{NLP} operates with a limited number of objects. The main objects are:

- Formed set of groups to which you want to add a new group.
- The set of morphological features $M_{w_i^{Synt}}^{NLP} \subseteq M^{NLP}$ is about the i-th term w_i^{Synt} of the analyzed text fragment.

Each rule applies to every w_i^{Synt} term from left to right. Each rule for a newly built group indicates its main group, a list of grammes (taken from the list of grammes of the main word), type.

Let us give an example of the operation of the algorithm for extracting a syntactic graph using the example of a post offer of one of the communities of the social network Facebook: *"Still need to use Javascript to get full control over the interface".* The resulting parse tree is shown in Figure 3.

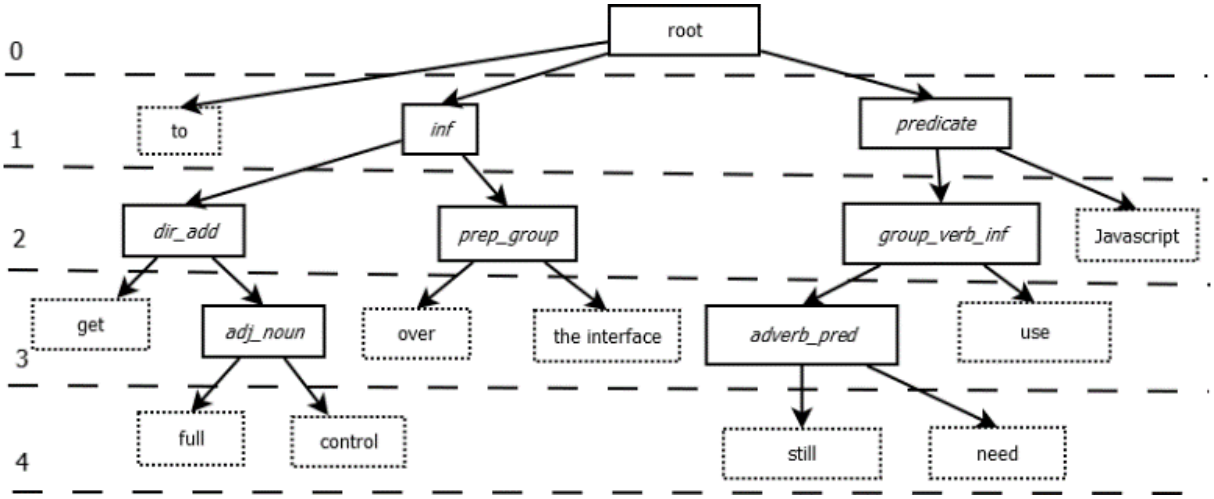

Figure 3: Example parse tree

3.2 Translation of a parse tree into a semantic tree

The function of translating a parse tree to a semantic tree is:

$$F^{Sem}: \{W_{li}^{Synt}, P_j\} \rightarrow \{W^{SEM}, E^{SEM}\}$$

where W_{li}^{Synt} – is the i– th node of the l-th parse tree level.

P_j–is the j-th rule for determining nodes of a parse tree that will be translated into nodes and relations of a semantic graph. Formally, the rule is:

$$\left(W_1^{Synt}, W_2^{Synt}, \dots, W_k^{Synt}\right) \rightarrow \{W^{SEM}, E^{SEM}\}. k = \overline{1, K},$$

where W_k^{Synt} is the k-th linguistic unit of the rule corresponding to the node of the semantic graph. For the rule to work, it is necessary that all linguistic units included in it are involved;

K is the number of linguistic units in the rule; W^{SEM} is the set of nodes and E^{SEM} is the relation of the semantic graph.

The application of the rules for translating nodes of a parse tree into a semantic graph [19] allows you to get the graph shown in Figure 4.

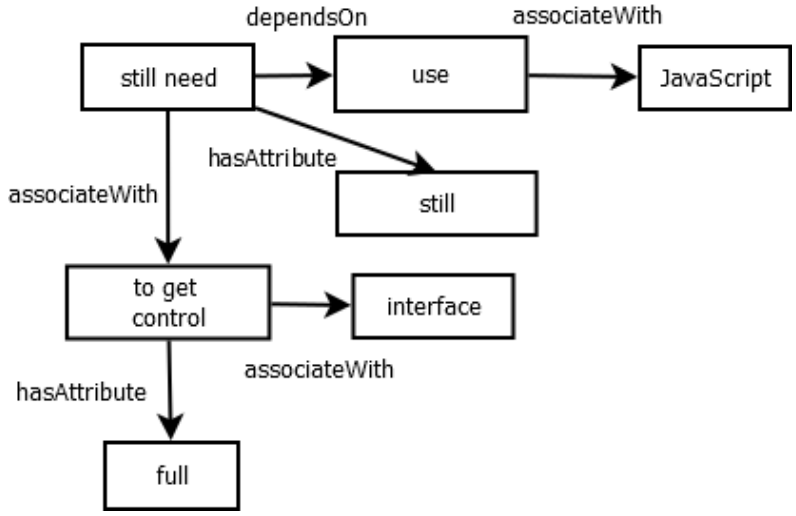

Figure 4: An example of the obtained semantic graph

Thus, the proposed algorithm integrates the linguistic and semantic approach and structures text resources, increasing the efficiency of the search in text fragments on the NL.

4. The approach to the search in the graph knowledge base using the mechanism of syntagmatic patterns

Search in text fragments (posts, comments) of social network profiles can be carried out as a result of translation of text arrays into a tree view of the user posts database (UPD). Formally, the structure of the UPD tree is

$$T^{UCD} = (P^{UCD}, TD, R^{UCD})$$

where $P^{UCD} = \{P_1^{UCD}, P_2^{UCD}, \dots, P_n^{UCD}\}$ is a set of syntagmatic patterns. A syntagmatic pattern is a combination of several words (n-grams, n> 1), united by the principle of semantic-grammatical-phonetic compatibility.

$TD = \{TD_1, TD_2, \dots, TD_n\}$ is a set of text data (the contents of the UPD);

$R^{UCD} = \{R^P, R^{TD}\}$ is a set of relations of the UPD tree:

$R^P = \{R_1^P, R_2^P, \dots, R_n^P\}$ is the set of relations between the patterns (internal nodes) of the UPD tree;

$R^{TD} = \{R_1^{TD}, R_2^{TD}, \dots, R_n^{TD}\}$ is the set of relations between the internal and terminal nodes of the UPD tree (syntagmatic patterns and text fragments).

The internal nodes of the UPD tree contain a syntagmatic pattern as a label. Terminal nodes contain textual data from which a response template for a search query is extracted. An example of the UPD tree is shown in Figure 5.

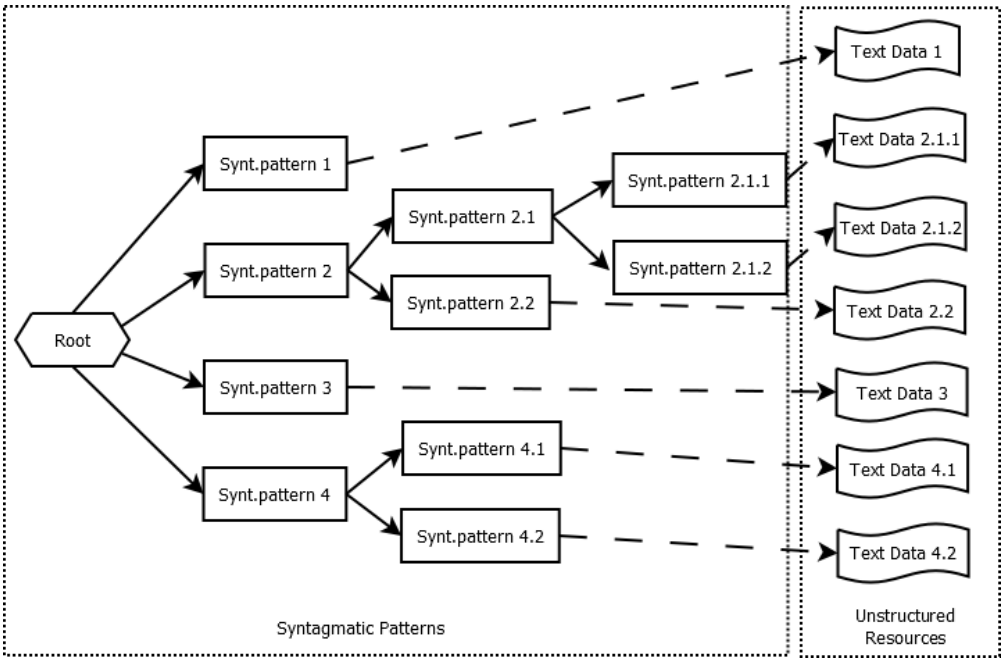

Figure 5: Social Network User Database Base Tree

Closer to the root element of the tree are more general syntagmatic patterns. Closer to the terminal nodes of the tree are more accurate syntagmatic patterns. Thus, the structure of the UPD tree allows you to find the necessary terminal nodes at the request of the user (if the answer to such a request exists in the UPD tree).

Using syntagmatic patterns as a unit of search in the knowledge base instead of individual terms allows you to semantically expand the scope of the search. For example, the syntagmatic pattern "software * development" will allow you to find sentences containing the following - grams:

- “software development”;
- “development of mobile software”;
- “development of an expert software system”, etc.

The UPD tree helps you find answers to user requests. First you need to find the desired terminal tree node. The labels of internal nodes are used to find the most relevant terminal node. Each internal

node of the UPD is marked with a syntagmatic pattern. The search algorithm for a user request includes the following steps:

- Search for the relevant terminal node of the tree;
- Search for relevant sentences from text fragments (TF) associated with the selected terminal node. The most relevant text fragment will be the answer to the search query.

Algorithm diagrams are presented in Figure 6.

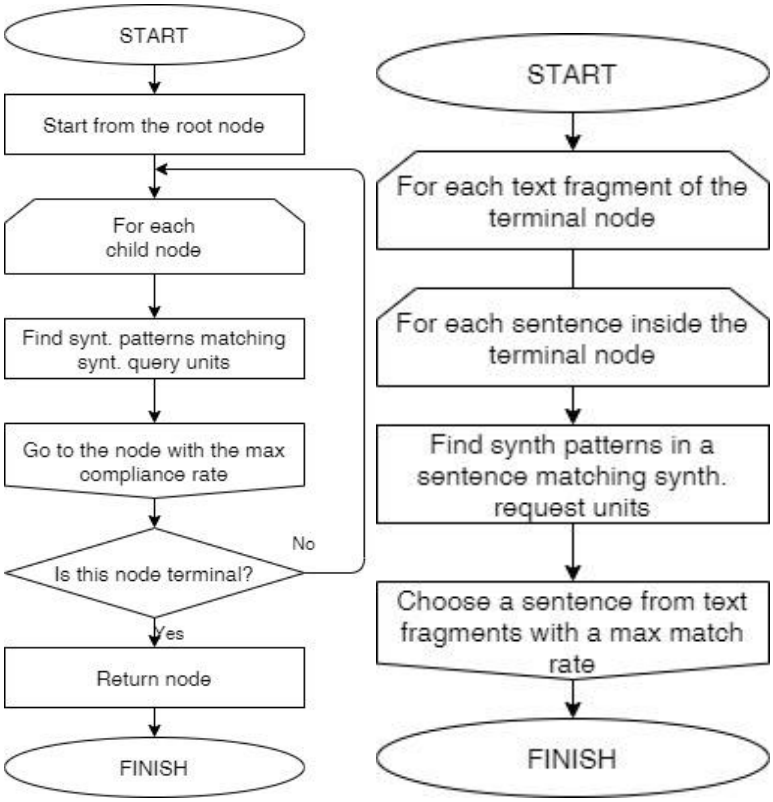

Figure 6: Scheme for finding the answer to the user request graph UPD

Thus, this algorithm organizes the search for the most relevant answer to a user's request.

5. The results of the experiments

A series of experiments was carried out confirming the effectiveness of the proposed models and algorithms in constructing a single knowledge base.

We used data from a live feed of users of the Facebook social network to build a UPD tree. The experiment included the following steps:

- A set of 1050 English-language profiles on the social network Facebook was randomly selected for the formation of training and test sets.
- The data of profiles and publications of selected users for the last month were uploaded to the developed information system for further analysis.
- Only user profiles with at least 10 English-language text publications for the last month were selected. 314 profiles were selected with a total of 5,744 publications.
- The UPD tree was automatically extracted for each publication.
- 50 free-form questions in a natural language were formulated and answers were searched in the extracted data using two algorithms:

1. developed algorithm based on the mechanism of syntagmatic patterns
2. keyword search algorithm.

Both algorithms can produce several sentences as answers. Each answer is evaluated by an expert: true or false.

The answers are conditionally divided into “long” (up to 250 characters) and “short” (up to 50 characters). The main metrics used in evaluating the effectiveness of search engines are accuracy, completeness, F-measure and average mutual rating. In the framework of this study, the F-measure indicator was used:

$$F_1 = 2 \times \frac{P \times R}{P + R},$$

where P is a measure of accuracy (P, precision), R is a measure of completeness (R, recall) [20]

Examples of the question and the answers received: " How does Dragon 2 dock to the ISS?". The answer, according to the algorithm of syntagmatic patterns, is for the user of Ruben Aldrete Jr.: “T*his is a game with the actual interface that Astronaut Doug Hurley will use to dock the Dragon V2 capsule to the ISS*.”.

The results of the experiments are shown in table 3 and figure 7.

Table 3

The results of experiments to find answers to user queries in graph UPD

Algorithm	Answer type	True positive	False positive	False negative
Synth Pattern Based Algorithm	Short answers	37	12	6
	Long answers	64	4	12
Keyword Search	Short answers	40	9	3
	Long answers	60	24	16

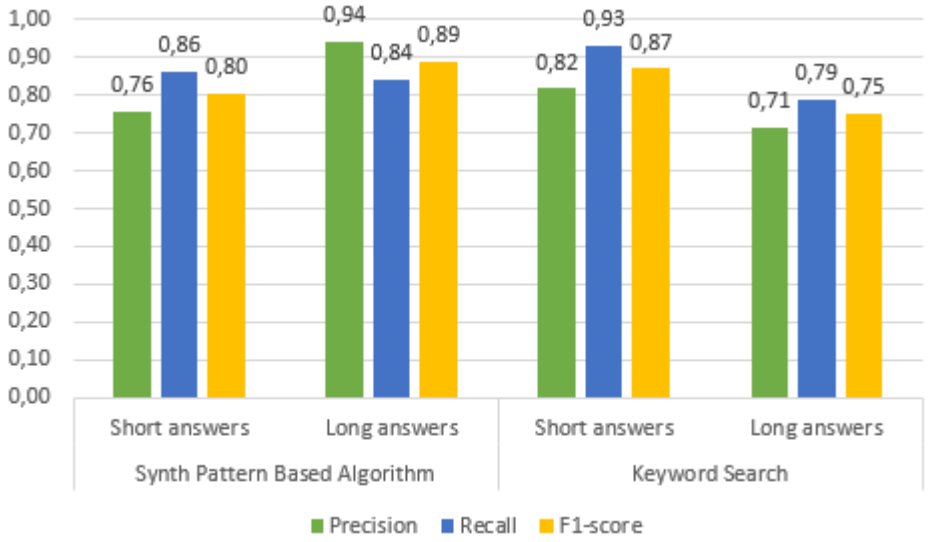

Figure 7: The results of experiments to find answers to user requests in graph UPD

As can be seen from the experimental results, the effectiveness of the use of keyword search and the algorithm based on syntagmatic patterns are almost identical when searching for short answers (F_{synt} = 0.8, $F_{keywords}$ = 0.87). This is explained by the lack of contextual information contained in the short sentence, and therefore, single-word terms are extracted as syntagmatic patterns from text fragments.

When searching for “long” answers, the developed algorithm found more correct answers (F_{synt} = 0.89, $F_{keywords}$ = 0.75) due to the selection of more complex semantic nodes from sentences that better define the meaning of the text fragment.

6. Conclusion

Thus, within the framework of this project, approaches to the formation of a unified knowledge base were developed. The knowledge base is formed by extracting structured and unstructured information from user profiles of social networks.

An ontological model for unifying user data of various social networks helps to avoid data redundancy by using graph structures and including contextual information in annotations to ontology relations. This approach is effective when matching information from the profiles of one person in different social networks. This approach is also effective when historical data need to be considered.

The approach to the formation of a semantic tree from text fragments using the integration of syntactic rules and knowledge engineering methods allows further merging of the obtained semantic trees into a single subject knowledge base of a specific information resource.

The developed approach to the search for information using syntagmatic patterns has shown its effectiveness in the search for long answers to the question posed.

In the future, it is planned to introduce fuzziness into the structure of the knowledge base (using the FuzzyOWL [21] notation) when solving the problem of fuzzy interpretation of search results in the database.

7. Acknowledgements

This study was supported by Foundation for Assistance to Small Innovative Enterprises in Science and Technology (contract No. 60GS1CTS10-D5/56043 dated 06.02.2020, "Development, technical implementation and testing of a prototype platform for the formation of a social portrait of an applicant based on intelligent data search in social networks using the principles of knowledge engineering") the Russian Foundation for Basic Research (Grant No. 18-47-730035).

8. References

[1] M. Najork, J. L. Wiener, Breadth-first crawling yields high-quality pages, Proceedings of the 10th international conference on the World Wide Web, ACM, 2001, 114-118.
[2] J. Leskovec, C. Faloutsos, Sampling from large graphs, Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006 , 631-636.
[3] M. Gjoka et al., Practical recommendations on crawling online social networks // Selected Areas in Communications, IEEE Journal on, 29, 9 (2011) 1872-1892.
[4] YouScan. URL: https://youscan.io.
[5] ForsMedia. URL: http://www.fors.ru/business-solutions/forsmedia.
[6] BrandAnalytics. URL: https: //br-analytics.ru.
[7] Feedot. URL: http://feedot.com.
[8] Hootsuite. URL: https://hootsuite.com.
[9] A. Filippov, V. Moshkin, N. Yarushkina, Development of a Software for the Semantic Analysis of Social Media Content. In: Recent Research in Control Engineering and Decision Making. ICIT. Studies in Systems, Decision and Control, Springer, 199 (2019) 421-432.
[10] Kasprzik, R. Yoshinaka, Distributional Learning of Simple Context-Free Tree Grammars., Algorithmic Learning Theory - 22nd International Conference, ALT 2011, 398-412. doi:10.1007/978-3-642-24412-4_31.
[11] K. Shu, S. Aziz, V. L. Huynh, D. Warrick, M. Marcolli, Syntactic Phylogenetic Trees. In: Kouneiher J. (eds) Foundations of Mathematics and Physics One Century After Hilbert. Springer, Cham, 2018, 417-441.
[12] M. A. Artyomov, A. N. Vladimirov, K. E. Seleznev, Review of natural text analysis systems in Russian, Bulletin of Voronezh State University. Series: System Analysis and Information Technology, 2 (2013) 189-194.
[13] G. Tomassetti, A Guide to Parsing: Algorithms and Terminology URL: https://tomassetti.me/guide-parsing-algorithms-terminology.

[14] D. Jurafsky, J. H. Martin, Constituency Parsing, Speech and Language Processing, 2019, URL: https://web.stanford.edu/~jurafsky/slp3/13.pdf.
[15] Automatic word processing. URL: http://aot.ru.
[16] N. N. Leontiev, On the status of knowledge in automatic text understanding systems, Computational Linguistics and Computational ONTOLOGIES./ Proceedings of the XVIII Joint Conference “Internet and Modern Society” (IMS-2015). 2015, 104-115.
[17] Estival, C. Nowak, A. Zschorn, Towards Ontology-based Natural Language Processing, Proceeedings of the Workshop on NLP and XML (NLPXML-2004): RDF/RDFS and OWL in Language Technology, Association for Computational Linguistics, 2004, 59-66.
[18] Semantic Web Rule Language. URL: https://www.w3.org/Submission/SWRL.
[19] N. Yarushkina, V. Moshkin, A. Filippov, I. Dyakov, The Approach to Extracting Semantic Trees from Texts to Build an Ontology from Wiki-Resources. In: Proceedings of the Third International Scientific Conference “Intelligent Information Technologies for Industry” (IITI’18), Advances in Intelligent Systems and Computing, Springer, 127-137.
[20] K. M. Ting, Precision and Recall. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA, 2011, https://doi.org/10.1007/978-0-387-30164-8.
[21] F. Bobillo, U. Straccia, Fuzzy ontology representation using OWL 2. International Journal of Approximate Reasoning, 52 (2011) 1073-1094.