MODEL OF LEXICOGRAPHICAL DATABASE: STRUCTURE, BASIC FUNCTIONALITY, IMPLEMENTATION

Olga Nevzorova, Farid Salimov

Abstract: *In the paper we describe the model of lexicographical database and Russian-Tatar lexicographical database data model.*

Keywords: *lexicographical database, linguistic resource, grammatical model.*

ACM Classification Keywords: *H.3 INFORMATION STORAGE AND RETRIEVAL H.3.1 Content Analysis and Indexing - Linguistic processing*

Introduction

Lexicography is the branch of linguistics that deals with the theory and practice of compiling the dictionaries. Theoretical lexicography develops typologies of dictionaries. The dictionaries may differ in form, content and by applied purposes. We can distinguish the following basic functions of dictionaries:

- registration of objective data about the world (encyclopedic dictionaries and defining dictionaries);
- structuring of language content (thesauruses);
- normalization of language as a mean of facilitating communication (standard and terminological dictionaries);
- lexical systematization for language learning (training dictionaries);
- translation from one language to another (bilingual and multilingual dictionaries);
- auxiliary operations.

The development of the dictionary is performed in accordance with a specific concept, which is reflected in the structure or vocabulary or explicitly explained in the introductory article. The use of different parameters in specific lexicographical dictionary is determined by the language specifics, lexicographical traditions, type and purpose of the dictionary and also viewpoints of lexicographers.

The development of new information technologies is contributed to the formation of computational lexicography. The main tasks of computational lexicography are linked with theory and practice of electronic dictionaries. Electronic dictionaries are software products that store dictionary information in a structured way. The development of electronic dictionaries is being performed by appropriate software tools and is being supported by specialized models and methods of text processing. We can distinguish the following types of electronic dictionaries:

complete electronic equivalents of printed dictionaries;

computational lexicographical systems and lexicographical databases based on appropriate lexicographical models.

The electronic version of dictionaries creates new opportunities for the presentation of lexical material including the possibility of partial presentation according to different criteria (different projections of the dictionary). The electronic version allows to apply different linguistic technologies (morphological and syntactic analysis, full-text search, speech recognition and speech synthesis, etc.) to the content of articles. Lexical entry can contain a variety of grammatical and semantic features, phonetic information, lists of translations, list associated with the key entries (collocations, standard phrases, etc.), text comments, synonymic and antonymic rows . Examples of successful commercial solutions in the field of computational lexicography are electronic dictionaries of “Lingvo” family of ABBYY company.

Lexicographical database and lexicographical systems

Technology development of modern information systems are closely linked with the technology of database development. The theory of database is a fully established discipline, based on formal approaches, methods and technologies of creating and maintaining the structured data, as well as finding information in large distributed datasets. To create and maintain a database (updates, access to records on request and issue records to the user) a database management system (DBMS) is used. Description of the database at the conceptual level is a generalized view of the data in terms of subject area (the application developer, user, or an external information system). The data model defines the rules of data structuring in the database. The data model includes a set of principles and methods for describing data and methods for manipulating data. According to the type used by the data model there are three common classes of databases: hierarchical, network and relational. By functionality, one can be identified as operational and reference information databases. The last one includes catalogs of electronic libraries, electronic dictionaries, statistical databases, etc. These systems are used to support core business processes and does not require changes to existing records. Operational databases are designed for more control of various technological processes. In this case, data are not only retrieved from the database, but data are changed (added), including the result of this usage. In the field of possible applications one can distinguish between universal and specialized (or problem-oriented) systems.

Lexicographical database (LDB) contains data on different levels of linguistic units (from morphemes to the text) and a variety of information about these units. One can specify the following applications for the LDB:

- supporting of operation of various automated systems associated with the processing of text and speech (information systems, expert systems, training systems, speech analysis, machine translation, etc.);
- computerized lexicographical tools for development of dictionaries of different types (training dictionaries, translation dictionaries etc.)
- automation of activities of researchers: linguists, language teachers and other professionals.

Structuring information in a database is a tool to automate the process of search of information in large data. Searching of information in the context of a database is considered as a sequence of operations to find objects with certain properties. One of the major problems is the problem of constructing an effective search in the databases. In [Ryzhov, 2004] the models of strict and fuzzy databases are introduced and the models of strict and fuzzy queries to the databases are developed. According to this classification, a strict database deals with a set of records whose values uniquely are interpreted by users. The strict request is regarded as a logical

expression with the logical terms that are expressed by the usual means of set theory To define request, it is possible to enumerate the values of attributes or objects or to indicate the change of parameters and attributes and to associate data of pair "attribute-value" with logical connectives.

Fuzzy query, in contrast to the strict query, may contain logical terms with fuzzy values. For example, the values of the attribute "size" may be "small", "very little", "big", etc. In this case, the boundary between the values of a fuzzy query is not defined, and each user has their own idea about these limits. This is the difference between a strict and fuzzy databases: the attributes of the fuzzy ones may be fuzzy values. Searching for a strict request in strict databases is well designed. To solve the problem of effective search one must be able to construct the predicates using first-order predicate logic, which take true values on the set of data interested. Execution time optimization should be carried out under constructing the query. Query is focused on searching the set of data that satisfy certain conditions. It is important to choose a formalized description language of the target set under constructing such queries. For this purpose, different tasks use different means: the language of predicate logic, the language of regular expressions, etc. For example, conjunctive normal form (CNF) as the basic formula for writing a query is used in the Russian National Corpus [Ruscorpora]. The user selects the required characteristics from a set of attributes on understandable language under constructing the query. The system based on the selected values, builds a formula of query. In particular, one can search a set of lexical items for a given set of grammatical or semantic attributes. The structure of the base relations has important role under performance the query. When designing a database it is necessary to consider the structure and types of possible queries. For fuzzy queries the problem of describing the boundaries of the fuzzy variables (for example, what value of size can be interpreted as a "big") arises. In [Ryzhov, 2004] the approach to the formation of such requests involving the theory of fuzzy sets is discussed.

In [Shirokov, 1998] the concept of lexicographical system is described. The lexicographical system is regarded as an information environment in which lexicographical models are implemented. Special cases (or implementations) of lexicographical systems are the systems of description language (both computational and traditional systems). The dictionary as an abstract lexicographical system must have a structure that includes at least two essential parts - the left part (registry) and right part (interpretation part). The vocabulary has interpretation unlike word list. But the dictionary has a deeper structure, which appears in the structure of the left and right parts of the dictionary as a whole and its entries, as well as in the structure of links between dictionary entries or links between different parts of dictionary. Thus, the dictionary is a special kind of text, which is a description of the vocabulary in a systematic and structured way. Each dictionary entry begins with a dictionary headword. The set of headwords forms the vocabulary (or the left part of the dictionary). The choice of vocabulary depends on the purpose of a dictionary.

Vocabulary may consist of the following language units:

- phonemes;
- morphemes (prefixes, roots, suffixes, etc.) - for a dictionary of morphemes, grammatical dictionaries, dictionaries of word formation;
- lexemes - for the majority of dictionaries (monolingual, spelling, etc.);
- word forms - for grammatical dictionaries, dictionaries of rhymes, etc.;
- collocations - for phraseological dictionaries, dictionaries of idioms, dictionaries of clichés.

The right part of the vocabulary is used to explain the headword. Right part zones are designed for each dictionary individually. The right part may contain a list of synonyms of the word, the translation of words (for

dictionaries of foreign words), the interpretations of the concept, which describes the given word, and different applications (graphs, charts, figures, etc.). The set of all entries forms corpus of the vocabulary.

The structure of the dictionary entry can be quite complicated and has a large number of structural elements. Since the structure of the article defines a system of basic relations for the database, it is important formal description of such a structure. A formal approach to the description of the various linguistic objects helps to build the algorithms for the selection of objects based on the verification of certain conditions and predicates, which are defined in the model description.

Models of lexicographical systems and lexicographical databases can be considered as information systems and models for describing data and methods for manipulating data are very important. In this case the main lexicographical work associated with the isolation and descriptions of lexical units are performed by professional linguists although their professional activities may be supported by a set of specialized software tools. However, it should be clearly understood that the selected material requires special linguistic formalization for the effective integration of database technology. The next section will discuss a model describing the data and methods for performing searches on the developed data model, made in the design of the Russian-Tatar lexicographical database.

Russian-Tatar lexicographical database data model

At the moment, Russian-Tatar lexicographical database is being developed at Research Institute for Applied Semiotics at Tatarstan Academy of Science. Main aim of this project is developing baseline resource for linguistics software used for knowledge intensive science projects, such as Tatar language corpus and parallel corpus, machine translation and others. Developed solutions should have effective functionality, which supports storing and searching of multiparameter linguistic definitions. An important part of this is to implement extensions for inclusion of new languages (firstly Turkic languages with similar structures).

This linguistic resource can be characterized as having very fine levels of specification of linguistic markup and models internal relationships inside of and between Tatar and Russian languages lexical systems. Database consists of interlinked components, corresponding to Russian and Tatar languages. Each of these components have independent internal structure, caused by specifics of the language in question and linked on lexical equivalency level. Each language component is represented by grammar and semantic models.

Process of designing Russian-Tatar lexicographical database was geared towards solving several major problems:

- Building data definition schema based upon linguistic models;
- Development of coding scheme for encoding linguistic data to avoid duplication of source data;
- Development of effective search mechanisms for accessing data.

LDB structure is defined by theoretical linguistic models that are being used, also by formalization level of such models and by an ability of these models to be expressed in the database level logic.

Information retrieval tasks are comprised of several important problems, such as searching for grammatical attributes of a given word form (direct search problem), and reverse problem of searching for a set of lexical

tokens, whose grammatical attributes include those that supplied by user. Solving direct search problem is equivalent to solving morphological analysis problem over set of word forms, which are defined by dictionary.

Special properties of Tatar language defined a series of decisions, made in process of designing representation for its grammatical model (T-component). One of Tatar languages defining characteristics is separation of any word form into root and affix parts, with affixational part defining word form's morphologic attributes. All lexical tokens of Tatar language can be divided into four morphological types, according to which affixational morphemes can linked with base morphemes of specific type. Same morphological attribute can be characterized using different allomorphs, which can joined into morphological category classes according to their morphological and morphonological types. Because of that information about word form was divided into two parts: dictionary of base morpehemes and dictionary of inflexional suffixes. Dictionary of base morphemes stores information about lemmas and morphological and morphonological types. Dictionary of inflexional suffixes contains possible chains of affixes which are linked with dictionary of base morphemes based on types mentioned earlier. In theory, those chains can have infinite length because of duplicate morphemes. Research of the statistical data has shown that in Tatar language word forms with more than 5 affixes make up less than 1% of word forms encountered in texts. Because of that property, length of chains in dictionary of inflexional suffixes is capped at 5 allomorphs. Also linked with this dictionary is table of morphological categories, which contains information about rules for building affixational chains, in addition to decoded values of corresponding morphological attributes.

The figure 1 depicts part of database structure, linked to T-component. T_Base table contains information about lemmas, T_Okon describes possible affixal chains, which in turn are divided into morphological categories (M_Posled table). Attributes WG and FORMA describe morphological and morphonological types of lemma accordingly. Relationship between T_Base and T_Okon, defined by WG and FORMA attributes, is M:M.

In the case of database being constructed according to such principles, implementation of direct search of morphological attributes is rather trivial: you just need to split word form into root and inflexional suffix and retrieve morphological attributes based on suffix alone.

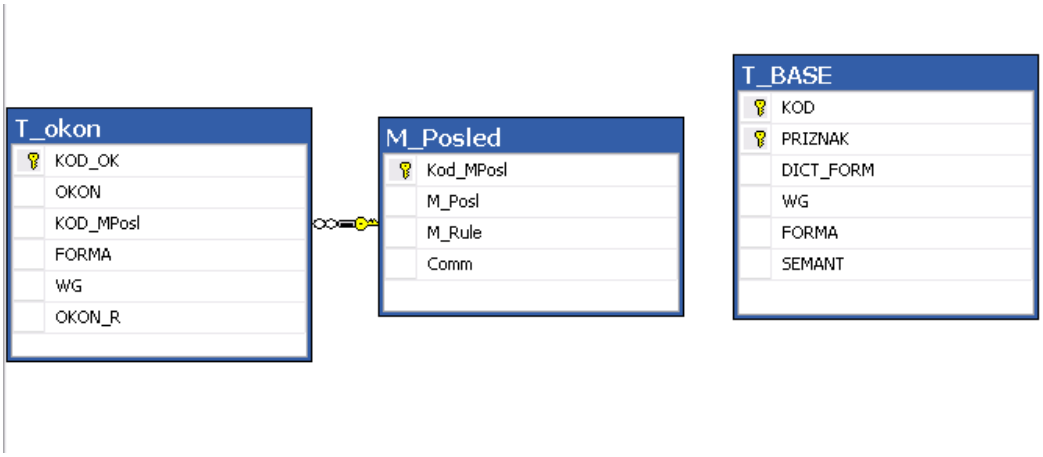

Fig. 1. The fragment of lexicographical database structure

Complications arise if you try to implement reverse search algorithm along same lines as a direct one. Full list if of grammatical attributes can be quite extensive, it's elements interlinked with others according to specific hierarchical relationships, each lexical token with its own set of morphological attributes. Storing full set of

attributes in database using standard RDBMS practices (each attribute has its own field) can be quite ineffective, mainly because in majority of cases values of attributes won't be equal and so resulting database will be sparse and riddled with empty values. In such cases it becomes necessary to use specially designed encoding methods adjusted to search techniques used. Simplest solution is to use encode information using characteristic vectors, consisting of 0 and 1 depending on membership of specific attribute within set that is being encoded. Set of attribute values is ordered according to specific condition and each affixal chain has its own characteristic vector. Retrieving set of word forms with specified set of attributes is implemented as matching resulting vector $\overline{b}$ with rows in table of morphological categories. If condition $\overline{b} \le \overline{a}$ is fulfilled, whereas $\overline{a}$ is characteristic vector for a sought morphological category, then search is stopped. Considering that for binary vectors $\overline{b} \le \overline{a}$ is equivalent to $\overline{b} = \overline{a} \,\&\, \overline{b}$, whole process can be implemented using only bitwise operations. Described approach has added benefit of accommodating further expansion of grammatical parameters set: arbitrary number of additional bits can be reserved in advance and adding data about new parameter can be as simple as switching 0 to 1 in appropriate position. Another approach is to split characteristic vector into parts $\overline{a_i}$ and $\overline{b_i}$, $i = \overline{1,k}$ with modified predicate $\&_{i=1}^{k} ((\overline{a_i} \,\&\, \overline{b_i}) = \overline{b_i})$. Main disadvantage of using bit vectors to encode attribute data is that each attributes position is fixed in accordance with enumeration order. It matters because of "recursive" inflectional suffixes in Tatar language which require keeping track of grammatical attribute's repetition factor in word form. For example, consider attributive affix - *Dagi* and possessive affix – *Nyky*. These affixes can be joined multiple times with root part of the word, potentially creating "cyclic" word form of theoretically infinite length (in Tatar *urman+nar+ym+dagi+lar+ym+dagi* … «…those in those that in my forests.. »). In such cases, it is necessary to reserve sufficient space in characteristic vector for repetition of attributes in affixational chain.

Another approach for encoding morphological data treats morphological attributes as formulas of propositional calculus. As mentioned earlier, sets of attributes forms a set of ordered structures, conjunctive normal forms can be used as an expressions. Each conjunction can be mapped to a definite grammatical characteristic and, being comprised of disjunctions, evaluates a set of values of respective attribute. Using this approach to encoding reduces searching to a comparison between two different formulas. Each formula A implements some function f_A, defined on some subset of grammatical attributes. Let's define that formula A covers formula B $(A \supseteq B)$ if for any subset of grammatical attributes over which $f_B = 1$ also true that $f_A = 1$. If we use conjunctive normal form, it is easy to introduce conditions for verifying coverage of a function by another. Assume that *B* is a formula, created from a targeted request from a user. Executing a search against this formula will result in set of word forms, whose linked formulas cover *B.*

Described approach is flawed because formula attribute breaks 1NF atomicity requirement. This requires checking coverage relationship using in-database defined functions. Checking process can be sped up by introducing linear ordering on conjunctions and disjunctions using weight function (which uses frequency distribution of an attribute in all formulas) over set of grammatical attributes and terminating comparison process if coverage breach is discovered.

At the moment second approach is implemented in the database. Encoded CNF of attributes is stored in *M_Posl* field of *M_Posled* table. Linguistic database is implemented using MS SQL Server 2005. Tests on database

consisting of 4000 lexical tokens have shown that query with formulas that include about 50 grammatical attributes takes 3ms to complete on 2.8GHz processor with 512MB RAM and internal caching enabled.

In conclusion we'd like to note that proposed approach to encoding grammatical components is language-independent and was used in processing Russian component regardless of completely different database schema. This approach can also be used for searching word forms with specified set of grammatical attributes.

Conclusion

Development of database for lexicographical models can be difficult task, requiring both comprehensive practical and theoretical knowledge of database systems with knowledge about lexicographical models and specific features of language structures. As a result, trade-offs in design process are inevitable, such as breaking atomicity requirement and introducing additional value processing procedures, which significantly complicates development process.

Design of structures and functional features of LSD is based on the specific problems of lexicographical data processing. Models database is focused primarily on the direct retrieval of data (find the characteristics of a given object), while at the same time, the task of goal-driven search (find all of the objects with given features) is very important for linguistic research. Combining in one data model effective solutions to the forward and reverse search is a difficult task.

The article is suggested quite efficient solutions of the above problems for the project Russian-Tatar lexicographical lexicographical database. In the future, we will plan to develop given LSD in the expansion of the semantic descriptions of vocabulary, as well as expanding to other Turkic languages, based on the description of grammatical formalisms proposed models.

Acknowledgements

The work was supported by the Russian Scientific Foundation for the Humanities, project No 11-14-16024a / B.

Bibliography

[Ryzhov, 2004] Ryzhov A.P. Modeli poiska informacii v nechetkoj srede. M. MGU, 2004. In Russian.
[Ruscorpora] Ruscorpora [Electronic resource] http:\\www.ruscorpora.ru/
[Shirokov, 1998] Shirokov V.A. Informacijna teorija leksikografichnih sistem. Kiïv, Dovira, 1998. – 331p. In Ukrainian.

Authors' Information

Olga Nevzorova *– Vice-director of Research Institute of Applied Semiotics of Tatarstan Academy of Sciences, Kazan Federal University. P.O. Box: 420111, Bauman str., Kazan, Russia; e-mail: onevzoro@gmail.com*
Major Fields of Scientific Research: Natural language processing, Artificial intelligence
Farid Salimov *– Senior Scientific Researcher of Research Institute of Applied Semiotics of Tatarstan Academy of Sciences, Kazan Federal University. P.O. Box: 420111, Bauman str., Kazan, Russia; e-mail: Farid.Salimov@ksu.ru*
Major Fields of Scientific Research: Software technologies, Multi-dimensional information systems