Control and Cybernetics

vol. **41** (2012) No. 1

Context search algorithm for lexical knowledge acquisition*†

by

Julian Szymański[1] and Włodzisław Duch[2]

[1]Department of Computer Systems Architecture
Gdańsk University of Technology, Poland
[2]Department of Informatics
Nicolaus Copernicus University, Toruń, Poland
e-mail: julian.szymanski@eti.pg.gda.pl, wduch@is.umk.pl

Abstract: A Context Search algorithm used for lexical knowledge acquisition is presented. Knowledge representation based on psycholinguistic theories of cognitive processes allows for implementation of a computational model of semantic memory in the form of semantic network. Knowledge acquisition using supervised dialog templates have been performed in a word game designed to guess the concept a human user is thinking about. The game that has been implemented on a web server, demonstrates elementary linguistic competencies based on lexical knowledge stored in semantic memory, enabling at the same time acquisition and validation of knowledge. Possible applications of the algorithm in domains of medical diagnosis and information retrieval are sketched.

Keywords: semantic memory, knowledge representation, information retrieval, knowledge acquisition.

1. Introduction

Natural Language Processing (NLP) is still one of the greatest challenges facing artificial intelligence. To understand a text people employ background knowledge, stored in their semantic memory (Tulving, Bower, & Donaldson, 1972; Collins & Loftus, 1975; McClelland & Rogers, 2003). This memory is at the foundations of human linguistic competence, facilitating rich associations that provide meaning to the text being read (Martin & Chao, 2001). Computational models of semantic memory should improve natural language processing,

*Submitted: September 2010; Accepted: October 2011

†This is an extended and amended version of the paper, presented at the 5th Congress of Young IT Scientists (Międzyzdroje, 23-25.IX.2010).

allowing machines to understand basic concepts represented by words. "Understanding" is manifested by the ability to give words correct meaning in the specific context that they appear in, leading to appropriate inferences that follow from the general knowledge of cognitive agent endowed with semantic memory. Statistical approaches to NLP treat text as a sequence of characters, not as words that possess meanings, therefore they achieved rather limited successes. Grammatical approaches are based on artificial constructions imposed on natural language and have not been very successful, either. Only human brains are capable of using language, therefore neurolinguistic approach to NLP is our best chance to develop good algorithms in this area (Duch, Matykiewicz & Pestian, 2008).

Models of semantic memory data structures that may store and use lexical information in a way similar to humans are of great interest in artificial intelligence. Words control behavior, pointing to knowledge stored in the brain, but the big problem is how to construct lexical databases that will reflect this knowledge correctly. Handcrafted machine readable dictionaries, such as WordNet (Miller et al., 1993), have been very useful, but as a general purpose semantic dictionaries they are too limited and have too many deficiencies to be successful in particular applications. In this paper a method for acquiring lexical knowledge in restricted domains through interaction with humans is described. Based on fixed dialog scenarios NLP system communicates with people using simplified form of natural language, using its lexical knowledge already stored in semantic network to modify itself. This interactive self-control process enables the acquisition of common sense knowledge about the relations between language concepts.

Next section describes our approach to knowledge representation for semantic memory, Section 3 - the context search algorithm, Section 4 - a game used to validate usefulness of lexical knowledge, Section 5 introduces active dialogs that serve to acquire new knowledge, and Section 6 contains discussion and plans for future research.

2. Representing knowledge in semantic memory

Psycholinguistics (Gleason & Ratner, 1997) tries to model human cognition using computer models, but without understanding how knowledge is represented in the brain (Pulvermuller, 2003, Duch et al., 2008) only simple experiments may be analyzed. Knowledge representation is one of basic concepts in artificial intelligence, specifying the structures used to store and process information, determining what kind of inferences can be performed (Davis, R., Shrobe & Szolovits, 1993). The most flexible method for expressing knowledge is natural language. It is also the most difficult to formalize, and the problem of knowledge representation for natural language is still unsolved. Natural language computer interfaces and control systems, dialog systems, information retrieval and question answering systems are still at quite a primitive level.

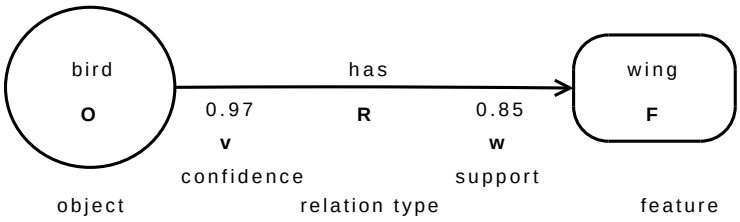

Figure 1. Atom of knowledge vwORF used for semantic memory model implementation

A flexible method to represent some aspects of word meaning is based on triples in the form of object – relation type – feature. This method can be employed for modeling data with first order logic (Guarino & Poli, 1995), currently popular in the form of RDF for ontology implementations (Staab & Studer, 2004). Such triples have also been used for building semantic networks (Sowa, 1991) and machine readable dictionaries (Calzolari, 1984). These triples are used here for implementation of the semantic memory model, but to increase their expressiveness two weights are added, enabling handling of uncertainty and learning process that helps in knowledge acquisition. The weights allow to encode fuzzy knowledge in the sense of fuzzy sets (Zadeh, 1996) and estimate importance of information (in terms of descriptiveness or reliability).

In Fig. 1 the elementary atom (unit) of knowledge vwORF used for implementation of semantic memory is presented. It consists of five elements which can be divided into two groups:

Triples of knowledge:

O – name of object (term), pointing to the concept encoded in semantic network

R – type of relation that binds objects with their features

F – feature that is related to some property of the object.

Weights:

v – confidence, a real number in the $\langle 0, 1\rangle$ range, estimating reliability of knowledge described by the triple. Value of v approaches 1 if strong confirmation of the knowledge expressed by the triple has been observed, but for new knowledge atom it is near 0.

w – support, a real number in the $\langle -1, +1\rangle$ range, estimates how typical is the feature for the object. Using this parameter adjectives such as: „always", „frequent", „seldom", „never" can be expressed, e.g.: for feature *black* associated with term *stork* support is $w = -0.5$ because it is *seldom* true, while feature *white* has $w = 0.9$ because storks are *almost always* white.

In Fig. 1 the utterance „bird has wing" is expressed using vwORF notation. It has high confidence ($v = 0.97$), estimated on the basis of frequent confirma-

tions observed by the system, and also high support ($w = 0.87$) expressing the belief that a bird usually has wing. A single triple is an atom of knowledge, with strong limitations: there is no way to say that a bird has no more than two wings, as can be done in the frame representation. However, more knowledge can be added using additional triples. The set of connected triples provides one possible model of semantic memory, forming a network that represents rich knowledge, denoted here by the ζ symbol.

Expressing knowledge in the form of semantic network ζ is quite natural for humans and may be seen as a reflection of some associations in the brain (Duch et al., 2008). Visual interface allows for easy modification of knowledge content, but such representation is not the most efficient for processing by computers. To enable fast numerical operations semantic network is mapped on a geometrical "semantic space" representation, denoted here as ψ. This is done by link-based representation, with each semantic network node **C** represented by a sparse n-dimensional vector of features **F** linked to it. This feature vector is called here the **C**oncept **D**escription **V**ector, or CDV.

Some features are irrelevant for a given object and thus are left undefined. ζ representation in the form of a graph can be transformed into its matrix representation (ψ). During mapping ζ into ψ selected types of links may provide additional knowledge that could be used to enrich CDV. In our approach we used four types of relations. They allow to introduce elementary inferences based on different ways how CDV are merged:

is_ a relation introduces in ζ the hierarchy of concepts through inheritance of features, contributing to cognitive economy. If relation of *is_ a* type between two objects has been identified, features from CDV of the superior object are passed on to the CDV vector of the inferior object. The v values related to the *is_ a* relation connecting two objects are multiplied by the w values related to each feature that is passed on. This allows features to be passed on down the hierarchy, taking into account confidence of knowledge.

similar – CDV features are copied from the first object to the second, new features have confidence factor v multiplied by the support w value for the first object. Note that if $v = 1$ this relation becomes "*same*", allowing for implementation of semantic memory object equivalence.

excludes – like ***similar*** but the support w value of the feature passed on is multiplied by -1.

entail – allows for making inferences from relations between features F_1 and F_2, adding F_2 feature to all CDV vectors for object where F_1 exists, with the same w value of F_2 as for F_1, and the confidence factor v associated with the relation.

Note that during processing of all above mentioned types of relation (mapping ζ into ψ) if relation between object and feature already exists in ζ then ψ is not modified. Performing the inferences based on processing of these relation types allows CDV vectors to be extended by adding new feature values. Fig. 2

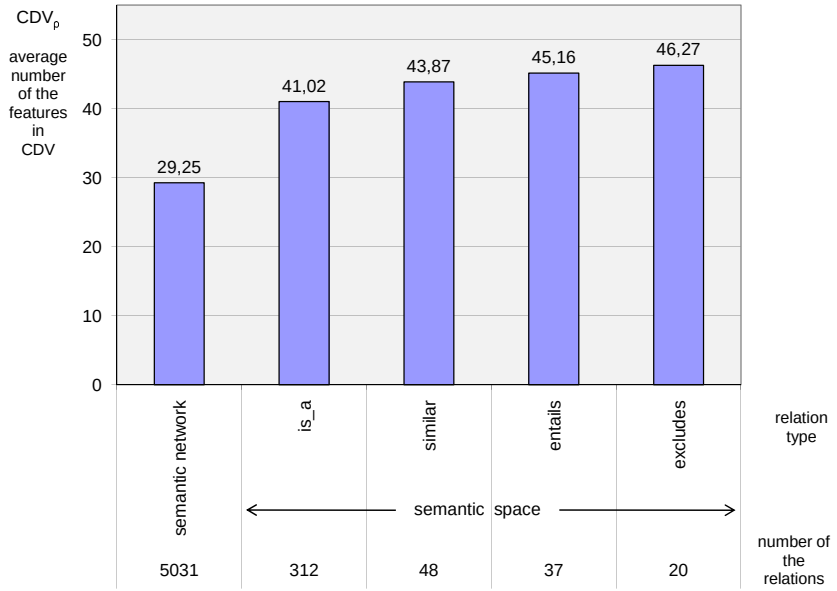

Figure 2. Average number of features in CDVs after adding new types of relations performed during mapping ζ into ψ.

shows an example (described below) illustrating how the processing of a particular relation type while mapping of ζ into ψ influences the average number of features defined in CDV vectors. The initial data stored in the form of semantic network have been constructed for 172 test objects from the animal kingdom domain. 475 initial features have been selected for description of these objects, with relations between them obtained from 3 lexical resources: WordNet (Miller et al., 1993), MediaMIT ConceptNet (Liu & Singh, 2004), and Microsoft MindNet (Vanderwende et al., 2005). Use of three independent resources allows for creating initial semantic network in an automatic way assuring high quality of knowledge stored in the network, with v confidence values set by confirming information in different sources. Relations that appear in only one data source are not used, if they are found in two sources, then confidence factor is $v = 0.5$, and if they appear in all three sources, $v = 0.75$. The confidence factors are changed further as a result of interactions with human users. Knowledge acquisition by aggregating three machine readable dictionaries created 5031 most common relations describing 172 animals with 475 features.

3. Context search algorithm

Semantic Network that stores relations between lexical elements can be useful in many applications. We have successfully applied this representation of

knowledge in text classification (Majewski & Szymański, 2008), where knowledge about relations of words has been used for evaluating text similarity. Semantic space ψ with vectors representing lexical elements allows to perform Context Search algorithm where objects are found referring to their features. This kind of search could be useful when a user does not know or cannot recall the name of the object (as in the *Tip of the Tongue* situations, Burke et al., 1991). Identifying objects by their features is rather common, and in such cases the keyword-based approach is not effective.

To identify objects in the semantic space one should start from specifying values of the most informative features. Given M terms (objects o) in the semantic space ψ spanned by N dimensions (features c) the best feature, in terms of discrimination, should have the highest Information Gain (IG) (Quinlan, 1986). In decision trees nodes are split to reduce entropy over class distribution. Here each object may be treated as a separate class, but also individual objects may be grouped into sets labeled by concepts that are at the higher level in ontology. If this is not the case, the entropy of feature c_j over all terms is calculated as:

$$H(c_j) = -\sum_{i=1}^{M} p_j(o_i) \log p_j(o_i); \qquad p_j(o_i) = |w_{ij}|/M \tag{1}$$

where w_{ij} is the support of the relation between object i and its feature j. Information gain is equal to the change of this entropy resulting from the split of all data after the value of feature c_j is fixed. Best feature has highest information gain, but in a large semantic space ψ frequently several features will have the same entropy. Additional preferences may then be based on term popularity, measured by the frequency of general usage (Hunston, 2001). Probabilities estimated from frequency of searched terms provide preferences that are more focused on a given search domain. In our implementation we use approach based on (1) that seems sufficient to obtain good results. However, providing additional information will influence the effectiveness of a search (measured as the number of questions used during the game). Improvement of this factor is our plan for the future research. It can be done in several ways: first we plan to include additional information about object's search probabilities (mentioned earlier), the second is to introduce information about correlations between features (that now are treated as separated ones).

In the middle of the search session or dialog with the query system a lot of features may already have definite value, either explicitly or due to propagation of values through relations. Some feature values may be correlated with others and these correlations should lead to faster convergence towards object identification.

Asking for the values of several most informative features narrows the set of potential target objects. Admissible answers should be restricted to a small

subset, in the implementation we use following coding: $w_{\text{ANSW}} = 1$ if the answer is "yes ", or -1 if "no", 0 for "don't know", and 0.5 for "frequently", -0.5 for "seldom". These answers are collected in the ANSW vector and used to calculate distances to objects in semantic space. Because knowledge stored in the semantic network has different confidence factors (v), and may be fuzzy (w), CDV and ANSW vectors are used to compute similarity in the following way:

$$d_o = d(\text{CDV}, \text{ANSW}) = \frac{1}{K} \sum_{i=1}^{K} (1 - \text{dist}(\text{CDV}_i, \text{ANSW}_i)) \tag{2}$$

where:

$$\text{dist}(\text{CDV}, \text{ANSW}) = \text{dist}(w_{\text{CDV}}, w_{\text{ANSW}}) = \begin{cases} 0 & \text{, if } w_{\text{ANSW}} = \text{NULL} \\ -\frac{1}{K}|w_{\text{ANSW}}| & \text{, if } v = 0 \\ v|w_{\text{CDV}} - w_{\text{ANSW}}| & \text{, if } v > 0 \end{cases} \tag{3}$$

where k is the number of questions asked by the system, v is the confidence, w_{CDV} is the weight w for CDV relations, and w_{ANSW} is the numerical value assigned to the answer for the question about a given feature. Similarity of the CDV and ANSW vectors is calculated as a sum of differences between user's answers and system knowledge. If the answer is „don't know" the feature is excluded from similarity calculation. Additionally the confidence factor v allows to strengthen these CDV components which are more reliable and weaken the influence of the accidental ones. Although this is quite simple, similarity measure vectors are usually compared by looking either at their Hamming distances or using cosine measures. Surprisingly, visualization of feature vectors representing animal properties using such naive distance measures, with both Kohonen's Self-Organizing Maps (Ritter & Kohonen, 1989) and with multidimensional scaling (MDS) (Duch & Naud, 1996) show similarities that agree with intuition, and form more general categories, like prey birds, domestic birds or large cats (see the MDS sample in Fig. 3). In fact, the MDS map of our vectors shows relations that are very similar to the experimentally derived similarity relations based on human ratings of semantic distances (Ripps & Shoben, 1973). As stated in Ripps and Shoben (1973) "Multidimensional scaling of the ratings suggested that semantic distance could be represented as Euclidean distance in a semantic space". Comparison of text fragments requires more sophisticated approach (Manning & Schutze; Szymański & Duch, 2011).

The minimal distance between ANSW and CDV allows for building a subspace $O(\text{ANSW})$ of objects that have the highest probability to be the target of the search in view of the answers obtained so far. In the k-th step (after k questions) of the context search algorithm this subspace covers objects with minimal distance:

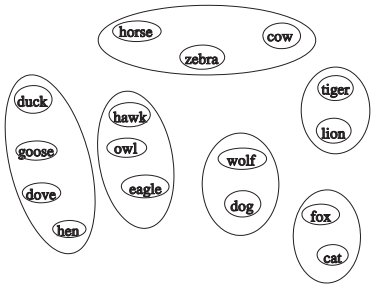

Figure 3. Similarities of vectors representing a few animals displayed using MDS.

$$O(\mathrm{ANSW}_k) = \{o \in O | d_o = \min_i \{d_k(\mathrm{ANSW}, \mathrm{CDV}(o_i))\} \tag{4}$$

where $\mathrm{CDV}(o_i)$ denotes i-th object in subspace O and $d_k(\cdot,\cdot)$ means that the distance is calculated in the subspaces of known answers. Using the minimal distance criteria for building $O(\mathrm{ANSW})$, the subspace in which the searched object lies, should minimize the number of features needed for the search. However, due to the wrong answers, errors in the data, changing targets during search, such an approach could miss some targets and would not contribute to corrections and acquisition of new data, as discussed below.

4. The game of questions

Context search algorithm can be applied in many domains. In fact, this process is similar to active learning, decision making, or trying to diagnose a problem by selecting questions and making additional tests or observations. Consider, for example medical diagnosis where disease should be identified by searching for most distinctive symptoms. In classification problems usually all features are used simultaneously, but in context search they are incrementally added until decision may be taken. This is in agreement with the signal detection theory of perception (Coren, 1994) that is now being extended to human decision making.

The context search algorithm has been tested in medical domain using data from "Diagnostic and Statistical Manual of Mental Disorders" (DSM IV) (DSM, 1994). Faster diagnosis (lower number of steps) was achieved in comparison to the original DSM IV decision tree recommendations. Context search may also improve information retrieval from the Internet (Duch & Szymański, 2008) helping to select a subset of the most relevant pages based on answers to questions generated by the search engine. However, creation of features for large number of unstructured documents indexed by the search engine requires a very large scale semantic network and is computationally very expensive.

The word games are a popular entertainment that relies on human lexical subsystem. They can be based on matching letter combinations (as it is done in scrabble), or they can test user knowledge (as in quizzes). This second group of games has been reserved only for humans, as it requires broad knowledge and deep understanding of semantics. However, in February 2011 natural language processing system called Watson[1], created at IBM, demonstrated great progress in this area. Watson, running on computers having joint power 100 times greater than Deep Blue[2], beat human in the popular *Jeopardy!* quiz. To find the answers Watson is using methods for knowledge extraction parsing a very large textual repository (500GB). In our research we are focused on obtaining common sense knowledge that is obvious for humans. This kind of knowledge is especially hard to obtain in an automatic way because it is rarely found in texts. Such default knowledge is obvious for humans, and is the basis for capturing the meaning of the words. Without it there is no real understanding, just a clever template matching, as the creators of Watson admitted in an interview.

Context search process may also be used in the popular 20-questions word game, where one person is asking questions trying to guess the concept that the opponent has in mind. The game is relatively simple for people, because they have extensive common knowledge about the world, but non-trivial for machines, because success does not depend on computing power but relies on knowledge about the world. Such knowledge may only partially be represented by relations between lexical elements, the ability to make at least shallow inferences is also necessary. Even a few hints in a proper context are sufficient for humans to correctly identify the concept and prepare appropriate answer or action. To achieve similar competence in software good models of the semantic and episodic memories are necessary.

The 20-question game may also be used to test elementary linguistic competencies needed to capture the real meaning of a discourse instead of responding by template matching. Using knowledge encoded in semantic network (vwORF weighted triples are used in the network nodes) computer program tries to guess the concept that the player has in mind. In the present implementation[3] only 5 answers are accepted: *yes/no, seldom/frequently*, and *do not know*. Implementations of this game available on the Internet[4,5,6] are based on learning correlations between questions and target concepts rather than systematic knowledge that may be used in many other applications. For example, it is easy to generate word puzzles in an automatic way using vwORF knowledge representation. In other approaches hard coded questions are used, while our algorithm actively generates most informative questions. Knowledge acquisition is the main bottle-

[1] http://www.ibm.com/innovation/us/watson/what-is-watson/index.html
[2] http://www.research.ibm.com/deepblue/
[3] http://diodor.eti.pg.gda.pl
[4] http://www.20q.net
[5] http://www.braingle.com/games/animal/index.php
[6] http://en.akinator.com/

Figure 4. Avatar used in the implementation of the game under Internet Explorer

neck in expert systems (Cullen & Bryman, 1988), but here large scale machine readable dictionaries have been used to create initial semantic network, and knowledge is validated, corrected and enhanced in human – computer interaction, as discussed in next section. Thus, our approach is aimed at achieving artificial general intelligence (Voss, 2007), rather than creating specialized solutions to different applications.

To make the game of questions more attractive some modifications to the algorithm presented above have been introduced.

1) To avoid frequently repeating the same question and to validate more knowledge atoms features are selected randomly with probability related to their information gain - roulette reproduction algorithm in genetic algorithms works in similar way in quite different context (Goldberg, 1989). This modification makes the search a bit less effective, but in the tests differences have not been significant.

2) Selecting the subspace O(ANSW) of most probable objects using minimal distance d_{min} between ANSW and CDV vectors (equation 2) may miss the target object if distances are large. To prevent this situation, the subspace O(ANSW) is created using the probability given by Boltzmann distribution:

$$p(\Delta d, k) = \left(1 + \exp\left(\frac{\Delta d}{kT}\right)\right)^{-1} \tag{5}$$

where Δd is the increase of CDV and ANSW distance relative to d_{min}, k is the current number of questions asked, and T is constant, set to 0.2 after some experiments. Larger subspace O(ANSW) will lead to more questions that need to be asked but this has been observed only for popular concepts that are

identified in few steps, for longer games larger k (equation 5) makes the search equivalent to d_{min}.

3) Stop condition: search may stop in 3 cases:

- If only one object is left in the $O(\text{ANSW})$ subspace. It happens rarely because knowledge is incomplete and may not be sufficient for unique identification.
- If a limited number of objects is left in the $O(\text{ANSW})$ subspace, heuristic guessing is a good strategy. An object significantly different from other objects in the subspace $O(\text{ANSW})$, i.e.:

$$d_p = \Delta(d_{min+1} - d_{min}) > \text{std}(O(\text{ANSW})) \tag{6}$$

  is a good candidate to question about it directly. Here d_{min} is the minimal distance in the $O(\text{ANSW})$ set between CDV and ANSW, d_{min+1} is the second minimal distance, $\text{std}(O(\text{ANSW}))$ is the standard deviation of the distances in $O(\text{ANSW})$. This heuristic decreases the number of questions considerably but occasionally leads to wrong objects.
- If the maximum number of questions is reached. Allowing for only binary answers 20 questions may in principle allow for distinguishing over one million objects ($2^{20} = 1,048,579$). Thus, this seems to be a reasonable maximum number of the questions allowed.

4) The game may be used on a web page, interacting with talking head (avatar, Fig. 4), an example of HIT (**H**umanised **I**n**T**erface). It is using MS ActiveX technology, therefore full interaction is available only under Internet Explorer. This implementation serves as the testbed for integration of various technologies making the web applications more user-friendly (Szymański, Sarnatowicz & Duch, 2008). Haptek[7] 3D head was integrated with text to speech engine and speech recognition software[8] (available only in console version). Technical problems with such implementations show that HIT man-machine interfaces are still very difficult to use.

5. Knowledge acquisition through Active Dialogs

To verify existing knowledge and acquire new concepts, the context search algorithm (implemented in the word game) may be extended by adding active dialogs, templates of interactions, run in various stages of the game. Currently, three templates are used:

1) If the program has guessed the concept correctly, additional question is asked: *Is that right?* to verify quality of the knowledge stored within semantic network. Using the *yes/no* answers given by the users to this question a precision measure is defined as the number of games that terminated with success divided by the total number of games played, $Q = N_s/N$. Initially $N = 30$ test games for

[7] http://www.haptek.com

[8] MS SpeechAPI http://www.microsoft.com/speech/speech2007/default.mspx

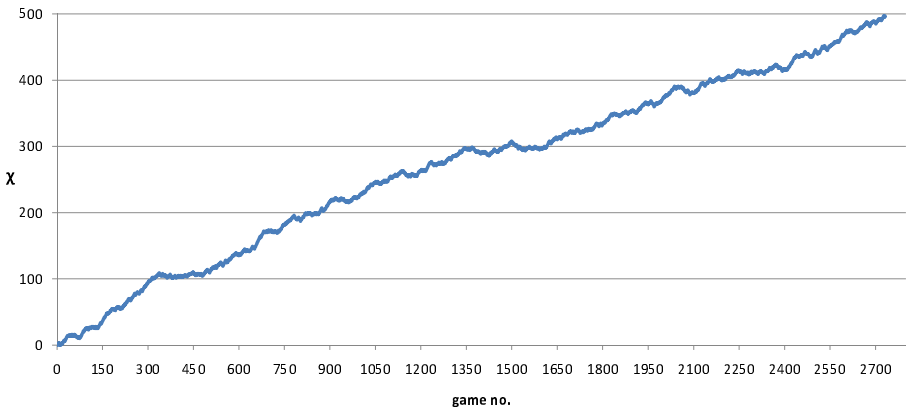

Figure 5. The dynamics of the competence process measured for 2700 games.

concepts from ζ knowledge based are run, selected with probability distribution given by the normalized number of features in their CDV to favor more popular concepts.This gave $Q = 0.70$, indicating that current knowledge is a good start, but there is ample room for improvements.

2) The ANSW vectors are used to enrich CDV vectors of concepts correctly identified by context search algorithm. If some ANSW features are already defined in CDV they modify w weights. Additionally, the program asks: *Tell me something about this <concept>*. Full analysis of the answer requires deep parsing to extract the knowledge in the vwORF form (Szymański et al., 2008), but only limited parsing has been implemented so far. If the concept may be mapped into some ontology, a list of candidate properties may be automatically generated and the user may be asked: are all these facts true? This will bring additional knowledge to semantic network.

3) If the search has not been successful, an additional question is asked: *Sorry, I fail to guess your concept. What was it?* The answer may be either a new concept that is added to the semantic network with the features taken from the ANSW vector, or the existing concept, in which case the reasons for failure have to be analyzed. Usually this is due to incorrect associations between feature and objects. They have to be pointed out to the user and if confirmed: *I expected that this* concept *has this* CDV feature*, but your answer was* ANSW feature*, is this correct?*.

These three templates allow for acquisition and verification of lexical knowledge of the system.

To evaluate the competencies of the system (ability to retrieve proper objects) we introduce χ measure defined in (7) as a sum of K games results:

$$\chi = \sum_{i=1}^{K} RES_i \, , where \; RES = \begin{cases} -1 & \text{, if game fails} \\ 1 & \text{, if game is finished with success.} \end{cases} \quad (7)$$

When the game had terminated with success, χ was increased, otherwise it was decreased by 1. In Fig. 5 we present graph of the game competencies that has been measured for K=2700 games. The games had been performed in limited domain of animal kingdom between the system and human users from the Internet. The increasing trend of the curve indicates that growing number of interactions with the system positively influences its ability to guess animal names that human user thinks about. Note the 0 point at the horizontal axis, denoting the starting time when semantic memory has been initialized with the data from machine readable directories. During the 2700 games our system obtained 147 new objects unknown to it before.

6. Discussion and future work

Semantic Memory as an element of the human cognition process has been a subject of many psycholinguistic theories of language. They provide good inspirations for building computational approximation of that process, but successful implementations of such models require a lot of lexical knowledge. Obtaining the common sense associations between lexical concepts, obvious to humans, is the prerequisite for effective natural language processing needed to approximate, using computational models, processes responsible for language understanding in the brain.

Knowledge representation methods are at the core of artificial intelligence. The weighted triples *vwORF*, proposed in this paper, have been inspired by psycholinguistic theories of human semantic memory. Many projects in Natural Language Processing are too ambitious and in the end fail to provide any useful results. Semantic networks built from the *vwORF* atoms of knowledge converted to a vector space representation for numerical efficiency offer a flexible approach to store and use lexical knowledge. Although such representation does not solve all NLP problems, using it the context search algorithm demonstrated elementary linguistic competence that has not been shown by more sophisticated NLP systems. Implementation of a word game has been used for verifying and acquiring new relations between lexical elements. This goes well beyond simple template matching used in most NLP projects, including chatterbots.

Bootstrapping approach to the problem of automatic lexical knowledge acquisition has been used here, creating initial imperfect semantic space from machine readable dictionaries, and then improving it by interaction with humans using active dialogs. Although in the present implementation only few active dialogs have been used to demonstrate the ability for acquiring common sense knowledge about language concepts, adding more templates should lead to progressively higher linguistic competencies in natural language processing.

This common sense knowledge has been evaluated and corrected in a series of experiments involving human players. This step is frequently missing in construction of lexical databases – consider for example WordNet, a huge effort built without feedback from ordinary users who could complete missing knowledge, stratify it and indicate its more and less important elements. So far our tests have been performed only in a limited domain as a proof of concept rather than real application. The next step is to use context search algorithm on a much larger scale to improve information retrieval from the Wikipedia. Interaction of many volunteers could lead to a large scale semantic network, verified in action during numerous information retrieval sessions. Potential applications range from information retrieval, to natural language computer and robotic interfaces that should give us much more flexible control based on language commands.

Acknowledgement

This work was supported by Polish Committee for Scientific Research grant N N516 432338.

References

Burke, D., MacKay, D.,Worthley, J. and Wade, E. (1991) On the tip of the tongue: What causes word finding failures in young and older adults. *Journal of Memory and Language* **30** (5), 542–579.

Calzolari, N. (1984) Machine-readable dictionaries, lexical data bases and the lexical system. In: *Proceedings of 10th International Conference on Computational Linguistics*. Association for Computational Linguistics, 460–460.

Collins, A., Loftus, E. (1975) A spreading-activation theory of semantic processing. *Psychological Review* **82** (6), 407–428.

Coren, S. and Ward, L.M. (1994) *Sensation and Perception.* Harcourt Brace, Toronto, 4th edition.

Cullen, J., Bryman, A. (1988) The knowledge acquisition bottleneck: time for reassessment? *Expert Systems* **5** (3), 216–225.

Davis, R., Shrobe, H. and Szolovits, P. (1993) What Is a Knowledge Representation? *AI Magazine* **14** (1), 17–33.

DSM (1994) *Diagnostic and Statistical Manual of Mental Disorders.* American Psychiatric Association.

Duch, W., Matykiewicz, P. and Pestian, J. (2008) Neurolinguistic approach to natural language processing with applications to medical text analysis. *Neural Networks* **21** (10), 1500-1510.

Duch, W. and Naud, A. (1996) Multidimensional scaling and Kohonen's self-organizing maps. In: *Proc. of the 2nd Conference "Neural Networks and their Applications"*. Szczyrk, Poland, **I**, 138-143.

Duch, W. and Szymański, J. (2008) Semantic web: Asking the right questions. In: *Proceedings of the 7 International Conference on Information and Management Sciences.* California Polytechnic State University Press.

Gleason, J.B. and Ratner, N. B. (1997) *Psycholinguistics.* Wadsworth Publishing, 2nd edition.

Goldberg, D. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

Guarino, N. and Poli, R. (1995) Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human Computer Studies* **43** (5), 625–640.

Hunston, S. (2001) Word frequencies in written and spoken English: Based on the British national corpus. *Language Awareness* **11** (2), 152–157.

Liu, H. and Singh, P. (2004) ConceptNet. A Practical Commonsense Reasoning Tool-Kit. *BT Technology Journal* **22** (4), 211–226.

Ripps, L.J. and Shoben, E.J. (1973) Semantic distance and the verification of semantic relations. *Journal of Verbal Learning and Verbal Behavior* **12**, 1-20.

Majewski, P. and Szymański, J. (2008) Text categorisation with semantic common sense knowledge: first results. In: *Proceedings of 14th International Conference on Neural Information Processing.* Springer, LNCS, 285-294.

Manning, C. and Schutze, H. (1999) *Foundations of Statistical Natural Language Processing.* Massachusetts Institute of Technology Press.

Martin, A. and Chao, L. (2001) Semantic memory and the brain: structure and processes. *Current Opinion in Neurobiology* **11** (2), 194–201.

McClelland, J. and Rogers, T. (2003) The parallel distributed processing approach to semantic cognition. *Nature Reviews Neuroscience* **4** (4), 310–322.

Miller, G.A., Beckitch, R., Fellbaum, C., Gross, D. and Miller, K. (1993) *Introduction to WordNet: An On-line Lexical Database.* Cognitive Science Laboratory, Princeton University Press.

Pulvermüller, F. (2003) *The Neuroscience of Language. On Brain Circuits of Words and Serial Order.* Cambridge University Press.

Quinlan, J. (1986) Induction of decision trees. *Machine Learning* **1** (1), 81–106.

Ritter, H. and Kohonen, T. (1989) Self-organizing semantic maps. *Biological Cybernetics* **61** (1), 241-254.

Sowa, J. (1991) *Principles of Semantic Networks: Explorations in the Representation of Knowledge.* Morgan Kaufmann, Series in Representation and Reasoning. San Mateo, CA.

Staab, S. and Studer, R. (2004) *Handbook on Ontologies.* Springer Verlag.

Szymański, J. and Duch, W. (2011) Induction of the common-sense hierarchies in lexical data. In: *Proc. of 17th Int. Conf. on Neural Information*

Processing. **LNCS 7063**. Springer, 726–734.

Szymański, J., Sarnatowicz, T. and Duch, W. (2008) Towards avatars with artificial minds: Role of semantic memory. *Journal of Ubiquitous Computing and Intelligence* **2**, 1–11.

Tulving, E., Bower, G. and Donaldson, W. (1972) *Organization of Memory.* New York: Academic Press.

Vanderwende, L., Kacmarcik, G., Suzuki, H., Menezes, A. (2005) MindNet: an automatically created lexical resource. In: *Proceedings of HLT/EMNLP on Interactive Demonstrations.* ACL, Morristown, NJ, USA, 8–19.

Voss, P. (2007) Essentials of General Intelligence: The Direct Path to Artificial General Intelligence. In: B. Goertzel and C. Pennachin, eds., *Artificial General Intelligence.* Series *Cognitive Technologies.* Springer, Berlin-Heidelberg, 131-157.

Zadeh, L. (1996) Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems* **4** (2), 103–111.