ECONTECHMOD. AN INTERNATIONAL QUARTERLY JOURNAL – 2017. Vol. 6. No. 2. 27–32

Partial semantic parsing of sentences by means of grammatically augmented ontology and weighted affix context-free grammar

M. Davydov[1], O. Lozynska[2], V. Pasichnyk[3]

[1]Lviv Polytechnic National University; e-mail: maksym.v.davydov@lpnu.ua
[2]Lviv Polytechnic National University; e-mail: olha.v.lozynska@lpnu.ua
[3]Lviv Polytechnic National University; e-mail: Volodymyr.V.Pasichnyk@lpnu.ua

Received February 08.2017: accepted May 27.2017

Abstract. In spite of the fact that modern statistical and neural net based tools for parsing natural language texts supersede classical approaches there are still areas where generative grammars are used. These are areas where collection of universal parallel corpuses is still in the progress. National sign languages are among them. Ontologies and common sense databases play valuable role in parsing and translation of such languages. Grammatically augmented ontology (GAO) is an ontology extension that links phrases to their meaning. The link is established via special expressions that connect phrase meaning to grammatical and semantical attributes of words that constitute it. The article introduces a new approach to sentence parsing that is based on integration of ontology relations into productions of weighted affix context-free grammar (WACFG). For that reason a new parser for WACFG grammar was developed inspired by works of C.H.A. Koster. Basic properties of WACFG are discussed and the algorithm for selection and convertion of GAO expressions into the set of WACFG productions is provided. The proposed algorithm turned out to be feasible in the context of parsing and translating Ukrainian Spoken and Ukrainian Sign language. The developed approach for mixed semantical and syntactical sentence parsing was tested on the database of sentences from Ukrainian fairy tail by Ivan Franko “Fox Mykyta” where 92 % of sentences were correctly parsed.

Key words: Grammatically augmented ontology, weighted affix context free grammar, semantic parsing, syntactic parsing, template productions.

INTRODUCTION

Modern computer applications require reliable and quick semantical parsing of text. There applications are text summarization, information retrieval, human-computer interaction, question answering systems, expert systems, etc.

Complete semantic parsing is not yet possible due to the lack of required knowledge databases, thus the partial semantic parsing is considered. The main goal is to obtain the most probable semantic relations in a sentence and to describe other relations as syntactic. Such an approach leads to better sentence representation and translation than approaches that use only syntactic relations.

Most modern sentence parsers consist of a dictionary, morphological parser, database of grammar rules, and a module for syntactic and semantic analysis. The quality of syntactic analysis strongly depends of the quality of semantic analysis and word sense disambiguation. It is known that pure syntactic analysis is ambiguous when semantics is not known [1].

Known methods of parsing sentences, such as [2], perform word sense disambiguation and semantics parsing steps after syntactic analysis. It is worth pointing out that generating all possible sentence parsing trees before semantic analysis can lead to significant increase in number of possible syntax parsing trees that have to be considered afterwards. In contrast, the proposed method integrates semantic analysis into the syntax parsing algorithm to increase overall productivity. The most important speed up was achieved for languages that lack prepositions and have limited number of grammar forms. In these languages syntactic parsing mainly relies on semantic text understanding. Ukrainian Sign Language is one of the examples of such a language.

Sentence parsing is done by means of the weighted affix grammar over a finite lattice (WAGFL) that benefits from stochastic context-free grammar (SCFG) [3] and affix grammar over a finite lattice (AGFL) developed by C. H. A. Koster [4].

THE ANALYSIS OF RECENT RESEARCHES AND PUBLICATIONS

Syntactic sentence parsing. The approach based on generative grammars is one of the most studied among many methods of parsing sentences. Extended affix grammar (EAG) [5] and stochastic context-free grammar are fundamental extensions to it that make possible fusion of machine learning and rule-based systems.

Fast and reliable algorithm for parsing natural language texts utilizing affix grammar over finite lattice was developed by C.H.A. Koster. The algorithm imposes restrictions on the set of productions and attributes to make parsing computationally inexpensive but still leave it expressive enough for parsing most of natural sentence structures.

This paper uses grammar close to the approach introduced by C.H.A. Koster, but differs in the way affixes and productions are used to integrate semantic productions into the context-free grammar.

Mixed semantic and syntactic parsing. A well-known approach to parsing of sentences is based on head-driven or “lexicalized” parsing [6]. In this approach all partial phrases obtained in bottom-up sentence parsing are marked with special word that describes their main meaning. These words are used to estimate production probability based on corpus statistics. For example the phrase “cyan car” is marked with a word “car” and probability of a phrase “to have a cyan car” is determined by probability of construction “to have a car” that is more relevant that probability of construction “to have cyan car”. In our approach we use productions generated from ontology instead of using some special symbol that selects main meaning of the phrase. This helps to introduce word categories into the process of sentence parsing but limits the set of semantical phrases to the set declared by ontologies. The use of mixed approach that benefits from ontologies and head-driven grammar can be a subject of further studies.

Semantic parsing. Semantic parsing of sentences is a subject of studies in the field of artificial intelligence. Predicate logic [7] and sub-domain driven parsing [8] are two different approaches for semantic parsing. An approach based on predicate logic potentially can be used for any subject area, but in fact it is limited because most expressions of the language are not predicates but rather agreed-upon rules as was stated by Wittgenstein, Ludwig [9].

Latent semantic analysis [10] is a statistical approach that discovers text topic by exploring the set of its terms. This approach can be used to distinguish the meaning of scientific terms that are assumed to be used uniformly and in a single meaning. However, common words often contain homographs that lead to ambiguity in text parsing and understanding. Moreover, the use of idioms makes the meaning completely different from the meaning of words that constitute idiomatic phrase.

Syntactic sentence parsing based on semantic relations. Several sentences parsing methods utilize semantic relations between words as a complement to syntactic rules. The crucial part of them is a measure of semantic relationship between words. Such a measure can be established using statistics [11], ontologies [12], tensor factorization [13] or mixed methods [14]. The parsing is done using spanning trees [15], branchings [16] or trained PCFGs [17]. Weighting pairwise word dependencies makes it difficult to distinguish a group established expressions that consist of more than two words. A method based on adjusting the probability of generative grammar rules with accordance to semantic word relations is limited to the given set of grammar rules and does not benefit from the use of ontologies. Our approach uses ontologies for creation of new productions rather than adjusting weights of existing ones.

The use of ontologies in sentence parsing. The idea of ontology integration into the process of sentence parsing is not new. In [18] the generation of productions from ontologies for LTAG grammar parser was studied. In the article by Faten Kharbat [19] the WordNet ontology [20] is utilized to be the syntactic guide along with the Transition Network Grammar that helps to get better translation from English to Arabic language. The approach based on rich ontologies was used by Murat Temizsoy and Ilyas Cicekli [21] for Turkish language parsing. The approach uses ontologies to improve text meaning representation model for parsed sentences.

However, the problem of integrating hypernymy/ hyponymy relations into the process of sentence parsing was not previously studied. This article introduces a new method that extends the system of productions using ontology relations. These relations are expressions of GAO and hypernymy/hyponymy relations.

OBJECTIVES

The main objective of the research is to develop sentence parsing algorithm that benefits from the use of GAO expressions. The main part of the paper introduces major concepts of weighted affix context-free grammar, describes how ontology relations are transformed info the set of productions, outlines experimental results and provides an example of parsing sentences by means of the proposed method.

THE MAIN RESULTS OF THE RESERCH

Weighted affix grammar over a finite lattice. The developed parser is based on weighted affix grammar over a finite lattice. This grammar extends symbols of generative grammar with affixes that are used to decrease the number of productions required to describe a language.

Weighted affix grammar over a finite lattice G is defined as a 5-tuple (T,V,S,D,P), where T is a set of all terminal symbols, V is a set of all symbols, S is a starting symbol $S \in V \setminus T$, D is a set of disjoint affix domains $D=\{D_i\}$, $i=\overline{1,|D|}$ each domain D_i represents a set of affixes $A(D_i)$; P is a set of template and regular productions.

Regular productions are written as $\left\langle \left(V\times 2^A\right)^* \overset{w}{\to} \left(V\times 2^A\right)^* \right\rangle$, where A is a set of all affixes $A = A(D) = \bigcup_{D_j \in D} A(D_j)$, 2^A denotes power set of A, and $\left(V\times 2^A\right)^*$ denotes all non-empty strings of attributed symbols $s_1 s_2 \ldots s_k$, $k>0$, $s_j=(v_j, A_j)$, $(v_j, A_j) \in V\times 2^A$; $w \in R^+$ is a multiplicative weight of the production. The weight symbol can be omitted if it equals to one.

The template production is written as $\left(V, D_{inh}, A_{set}\right)^* \overset{w}{\to} \left(V, D_{uni}, A_{req}\right)^*$, where $D_{inh} \subset D$ is a set of domains which affixes are inherited, $D_{uni} \subset D$ is a set of domains where affixes that should be common for symbols in the right part of the production, $A_{set} \subset A$ is a set of affixes that are added to symbols in the left part of the production and $A_{req} \subset A$ is a set of affixes required by symbols in the right part of the production.

Terminal symbols $t_i \in T$ have void set of attributes. They represent words of sentence being parsed. For example, the word "student" can be a male or female singular noun until it is known from the context. In terms of regular productions it is written as:

$$(noun, \{a_{FEMALE}, a_{SINGULAR}, a_{STUDENT}\}) \to (student, \varnothing),$$
$$(noun, \{a_{MALE}, a_{SINGULAR}, a_{STUDENT}\}) \to (student, \varnothing),$$

and a template form is

$$(noun, \{a_{FEMALE}, a_{MALE}, a_{SINGULAR}, a_{STUDENT}\}) \to$$
$$\to (student, \varnothing)$$

that represents both cases provided above. Productions that generate terminal symbols are added by a morphological parser. The morphological parser generates one production for every meaning of the word. The weight of the production represents admissibility of this meaning in the parsed text.

Extending the set of productions with ontology relations. The grammar augmented ontology was introduced in [22]. Along with relations that are common to ontology databases (hyponymy, hypernymy, meronymy, holonymy) GAO contains relations that link synsets to expressions with associated grammatical attributes.

In order to benefit from ontology knowledge new productions were added to generative grammar. The addition of ontology productions into the generative grammar extends the set of semantic attributes. Each synset of ontology was treated as semantic attribute. For the purpose of efficiency semantic attributes and corresponding productions were added only for hierarchies that contained words that were present in the sentence.

Each expression in GAO is a tuple of grammatically augmented ontology terms:

$$e = \lfloor\langle(\sigma_1, A_{\sigma 1})(\sigma_2, A_{\sigma 2})\ldots(\sigma^*, A_\sigma^*)\ldots(\sigma_K, A_{\sigma K})\rangle\rfloor,$$

where σ_i is a synset that narrows the set of words that can appear in the given position of the expression e and $A_{\sigma i}$ is a set of grammar attributes the word is required to possess; σ^* is a head word of the expression. WordNet ontology defines hypernymy relation that was utilized to check if the word can can be a part of GAO expression.

Let $\Lambda = \{noun, verb, adjective, adverb, \ldots\}$ be a set of all lexical categories and $D_\Lambda : \Lambda \to 2^D$ be a mapping from lexical category to the set of its attribute domains.

For example, in Ukrainian attribute domains of noun are gender, number and case. Thereby:

$$D_\Lambda(noun) = \{D_{GENDER}, D_{NUMBER}, D_{CASUS}\}.$$

Each word w in the sentence can have several interpretations $\omega_1, \omega_2, \ldots, \omega_r \in \Omega$, where Ω is a dictionary of all interpretations. Each interpretation ω uniquely defines its semantic attribute $SemAttr(\omega)$, lexical category $LexCat(\omega)$ and the set of grammatical attributes $GrAttr(\omega)$.

The word in the sentence may become a head of some phrase. For example, word "car" may become the head of a noun phrase "green car". Since "green car" is semantically a kind of car the productions should be defined that let to treat phrases as if they are semantically represented by head word. For this reason the mapping $PhraseType : \omega \to V$ is defined that maps from word interpretation into grammar symbol that represents type of phrases based on this word (in English they are adverb phrase(AdvP), adjective phrase(AP), noun phrase(NP), preposition phrase(PP), or verb phrase(VP)).

The algorithm that adds new productions to syntactic parser is defined as follows.

Algorithm $addProductions(s)$

Input. Sentence s.

Output. Set P of productions that should be added to the grammar for parsing of sentence s.

for each word w_i in sentence s

  for each interpretation ω of word w_i

   Let $p := SemAttr(\omega)$, $PT := PhraseType(\omega)$.

   Add production

   $(LexCat(\omega), GrAttr(\omega) \cup \{SemAttr(\omega)\}) \to (w_i, \varnothing)$

   and template production

   $(PT, D_\Lambda(PT), H(p)) \to (PT, D_\Lambda(PT), \{p\})$, where $H(p)$ is a set of all hypernyms inherited by synset p.

   for each expression

   $e = \lfloor\langle(\sigma_1, A_{\sigma 1})(\sigma_2, A_{\sigma 2})\ldots(\sigma^*, A_\sigma^*)\ldots(\sigma_K, A_{\sigma K})\rangle\rfloor$, where

   $\sigma^* \in H(p) \cup \{p\}$ add template production

$$\begin{aligned}(PT, D_\Lambda(PT), \{\sigma^*\}) &\overset{1+\Delta w}{\to} (PhraseType(\sigma_1), \varnothing, A_{\sigma 1} \cup \{\sigma_1\}), \ldots, \\ &(PhraseType(\sigma^*), D_\Lambda(PT), A_\sigma^* \cup \{\sigma^*\}), \ldots, \\ &(PhraseType(\sigma_K), \varnothing, A_{\sigma K} \cup \{\sigma_K\})\end{aligned} \quad (7)$$

    end for

  end for

end for

Productions that are generated from ontology expression have larger weight than simple syntactic productions in order to dominate over them. Additional weight Δw in expression (7) is devised from the admissibility of the expression in the given context or text topic.

In the real-life applications expressions can lack several words because they can be devised from the context. In this case the process of conversion of expression (7) to normal form can be modified to decrease additional weight Δw in case where some of

$\sigma_1,..,\sigma_K$ words are omitted. An algorithm of this conversion is outside of the scope of this article.

The developed algorithm for parsing of sentences [23] was tested on a fairy tail by Ivan Franko "Fox Mykyta" where 92 % of sentences were correctly parsed using ontology rules developed manually. In order to illustrate how the proposed method works consider 2 sentences "My father bought several candies at the shop" and "My father bought several candies at the table".

The first step of the algorithm adds morphological productions for all words from these sentences:

$$(pronoun, \{a_{POSESSIVE}, a_{MY}\}) \to (my, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON\,3}, a_{FATHER-PARENT}\}) \to (father, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON\,3}, a_{FATHER-FOUNDER}\}) \to (father, \varnothing)$$
$$(verb, \{a_{PERSON\,1}, a_{PERSON\,2}, a_{PERSON\,3}, a_{SINGULAR}, a_{PLURAL}, a_{PAST}, a_{PAST\ PARTICIPLE}, a_{BUY-PURCHASE}\}) \to (bought, \varnothing)$$
$$(verb, \{a_{PERSON1}, a_{PERSON\,2}, a_{PERSON3}, a_{SINGULAR}, a_{PLURAL}, a_{PAST}, a_{PAST\ PARTICIPLE}, a_{BUY-BRIBE}\}) \to (bought, \varnothing)$$
$$(verb, \{a_{PERSON1}, a_{PERSON\,2}, a_{PERSON3}, a_{SINGULAR}, a_{PLURAL}, a_{PAST}, a_{PAST\ PARTICIPLE}, a_{BUY-BELIEVE}\}) \to (bought, \varnothing)$$
$$(adjective, \{a_{BOUGHT}\}) \to (bought, \varnothing)$$
$$(noun, \{a_{PLURAL}, a_{SEVERAL}\}) \to (several, \varnothing)$$
$$(numeraladjective, \{a_{PLURAL}, a_{SEVERAL}\}) \to (several, \varnothing)$$
$$(noun, \{a_{PLURAL}, a_{PERSON\,3}, a_{CANDY}\}) \to (candies, \varnothing)$$
$$(preposition, \{a_{PLACE}, a_{AT}\}) \to (at, \varnothing)$$
$$(definiteArticle, \{a_{THE}\}) \to (the, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON\,3}, a_{SHOP}\}) \to (shop, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON\,3}, a_{TABLE}\}) \to (table, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON3}, a_{TABLE-ARRAY}\}) \to (table, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON3}, a_{TABLE-MESA}\}) \to (table, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON\ 3}, a_{TABLE\ -PEOPLE}\}) \to (table, \varnothing)$$
$$(noun, \{a_{SINGULAR}, a_{PERSON\,3}, a_{TABLE-MEAL}\}) \to (table, \varnothing)$$

The next step adds all productions that describe hypernymy relations. The productions that were generated from WordNet [24] are depicted in Fig. 1.

Expressions from GAO for the verbal synset "buy" are transformed into productions of the grammar as depicted in Fig. 2.

After all of productions are set up the parser is called to parse the sentences.

The result of paring the sentences is depicted on Fig. 3. The weight of the first parse if higher because one expression was used instead of using regular syntactic productions.

$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{OBJECT}, a_{WHOLE}, a_{LIVING\,THING}, a_{ORGANISM}, a_{PERSON}, a_{RELATIVE}, a_{ANCESTOR}, a_{PARENT}, a_{GENITOR}, a_{PROGENITOR}\}) \to (NP, D_A(NP), \{a_{FATHER-PARENT}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{OBJECT}, a_{WHOLE}, a_{LIVING\,THING}, a_{ORGANISM}, a_{PERSON}, a_{CREATOR}, a_{ORIGINATOR}\}) \to (NP, D_A(NP), \{a_{FATHER-FOUNDER}\})$$
$$(VP, D_A(VP), \{a_{ACQUIRE}\}) \to (VP, D_A(VP), \{a_{BUY-PURCHASE}\})$$
$$(VP, D_A(VP), \{a_{TRANSFER}, a_{PAY}, a_{GIVE}\}) \to (VP, D_A(VP), \{a_{BUY-BRIBE}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{MATTER}, a_{SUBSTANCE}, a_{FOOD}, a_{NUTRIMENT}, a_{DAINTY}, a_{SWEET}\}) \to (NP, D_A(NP), \{a_{CANDY}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{OBJECT}, a_{WHOLE}, a_{ARTIFACT}, a_{CONSTRUCTION}, a_{ESTABLISHMENT}, a_{PLACE\ OF\ BUSINESS}, a_{RETAILSTORE}\}) \to (NP, D_A(NP), \{a_{SHOP}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{OBJECT}, a_{WHOLE}, a_{ARTIFACT}, a_{INSTRUMENTALITY}, a_{FURNISHING}, a_{FURNITURE}\}) \to (NP, D_A(NP), \{a_{TABLE}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{ABSTRACTION}, a_{GROUP}, a_{ARRANGEMENT}, a_{ARRAY}\}) \to (NP, D_A(NP), \{a_{TABLE-ARRAY}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{OBJECT}, a_{GEOLOGICALFORMATION}, a_{NATURALELEVATION}, a_{HIGHLAND}, a_{TABLELAND}\}) \to (NP, D_A(NP), \{a_{TABLE-MESA}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{ABSTRACTION}, a_{GROUP}, a_{SOCIAL\ GROUP}, a_{GATHERING}\}) \to (NP, D_A(NP), \{a_{TABLE-PEOPLE}\})$$
$$(NP, D_A(NP), \{a_{ENTITY}, a_{PHYSYCALENTITY}, a_{MATTER}, a_{SUBSTANCE}, a_{FOOD}, a_{FARE}, a_{TABLELAND}\}) \to (NP, D_A(NP), \{a_{TABLE-MEAL}\})$$

Fig. 1. The productions that were generated from WordNet

$$(VP, D_A(VP), \{a_{BUY-PURCHASE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-PURCHASE}\})(NP, \varnothing, a_{ENTITY})(preposition, \varnothing, \{a_{AT}\})(NP, \varnothing, a_{RETAIL\ STORE})$$

$$(VP, D_A(VP), \{a_{BUY-PURCHASE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-PURCHASE}\})(NP, \varnothing, a_{ENTITY})(preposition, \varnothing, \{a_{FROM}\})(NP, \varnothing, a_{SOCIAL\ GROUP})$$

$$(VP, D_A(VP), \{a_{BUY-PURCHASE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-PURCHASE}\})(NP, \varnothing, \{a_{ENTITY}\})(preposition, \varnothing, \{a_{FROM}\})(NP, \varnothing, \{a_{PERSON}\})$$

$$(VP, D_A(VP), \{a_{BUY-BRIBE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-BRIBE}\})(NP, \varnothing, \{a_{PERSON}\})$$

$$(VP, D_A(VP), \{a_{BUY-BRIBE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-BRIBE}\})(NP, \varnothing, \{a_{SOCIAL\ GROUP}\})$$

$$(VP, D_A(VP), \{a_{BUY-BELIEVE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-BELIEVE}\})(DS, \varnothing, \{a_{FACT}\})$$

$$(VP, D_A(VP), \{a_{BUY-BELIEVE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-BELIEVE}\})(conj, \varnothing, \{a_{THAT}\})(DS, \varnothing, \{a_{FACT}\})$$

$$(VP, D_A(VP), \{a_{BUY-BELIEVE}\}) \xrightarrow{1.1} (VP, D_A(VP), \{a_{BUY-BELIEVE}\})(conj, \varnothing, \{a_{THAT}\})(NP, \varnothing, \{a_{FACT}\})$$

Fig. 2. The productions of the grammar for the verbal synset “buy”

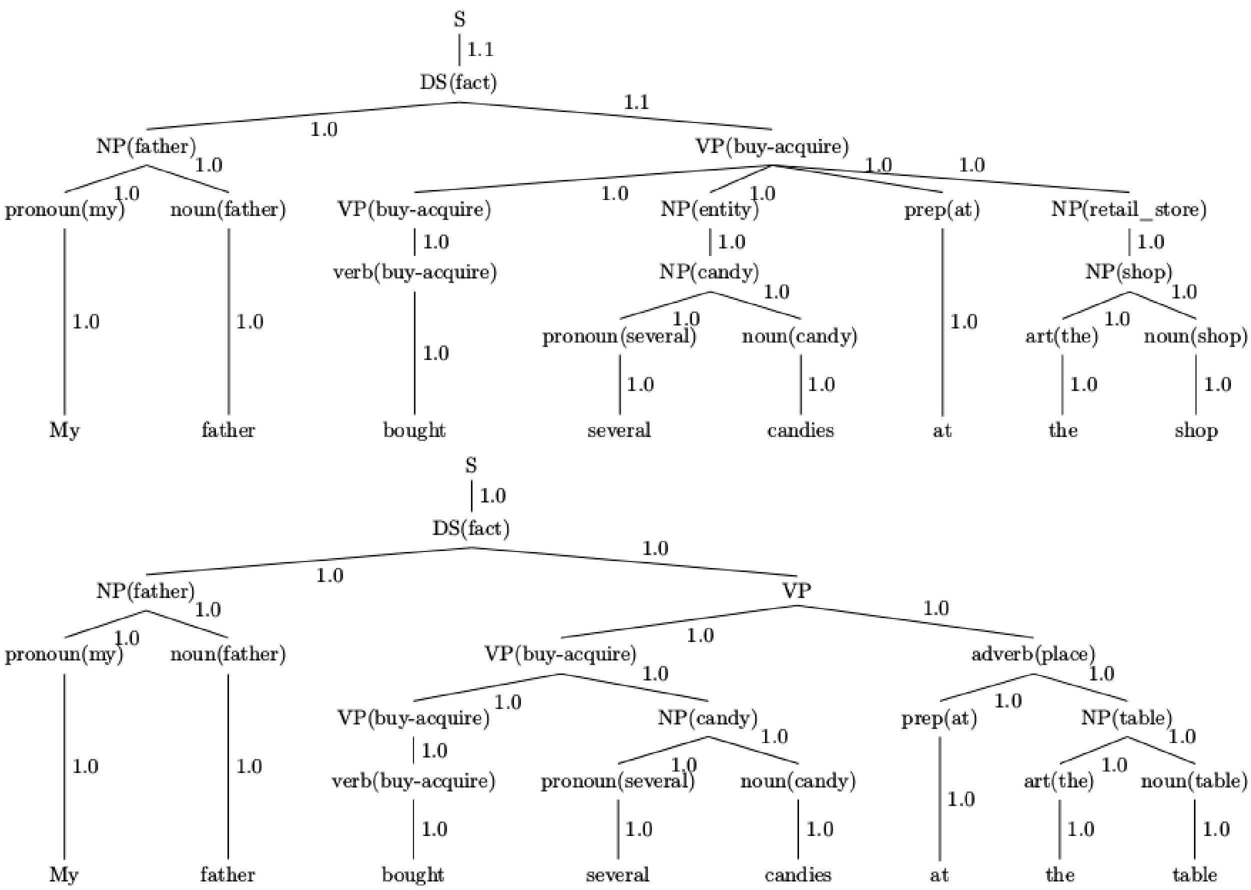

Fig. 3. The result of parsing sentences “My father bought several candies at the table” and “My father bought several candies at the shop”

CONCLUSIONS

The developed approach for mixed semantical and syntactical sentence parsing has shown promising results on fairy tail by Ivan Franko "Fox Mykyta" where 92 % of sentences were correctly parsed by means of the proposed method. The main drawback of the method is the necessity to develop GAO expressions manually. Further research can be focused on creation of the required expressions from crowdsourced resources or by means of machine learning.

REFERENCES

1. **Dallin D. Oaks. 2010.** Structural Ambiguity in English: An Applied Grammatical Inventory, 2 vols., Bloomsbury Academic. Reprint edition (February 23, 2012) London, p. 576.
2. **Anisimov A., Marchenko O., Taranukha V., Vozniuk T. 2014.** Development of a semantic and syntactic model of natural language by means of non-negative matrix and tensor factorization. – in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pp. 324–335.
3. **Eddy S .R. 1994.** RNA sequence analysis using covariance models / S. R. Eddy, R. Durbin // Nucleic Acids Research, Vol. 22, No. 11, pp. 2079–2088.
4. **Koster C. H. A. 1991.** Affix Grammars for natural languages / C. H. A. Koster // In: Attribute Grammars, Applications and Systems, International Summer School SAGA, Lecture Notes in Computer Science, Prague, Czechoslovakia. Vol. 545, pp. 469–484.
5. **Oostdijk N. 1984.** An Extended Affix Grammar for the English Noun Phrase / N. Oostdijk // In: Jan Aarts and Wim Meijs (eds), Corpus Linguistics. Recent Developments in the Use of Computer Corpora in English Language Research, Amsterdam: Rodopi.
6. **Collins M. 2003**. Head-driven statistical models for natural language parsing. In Computational Linguistics, pp. 589–638.
7. **Blackburn P., Bos J. 2005**. Representation and Inference for Natural Language: A First Course in Computational Semantics. CSLI Publications, Stanford.
8. **Plank B. 2007.** Sub-domain driven parsing. M. Sc. thesis. European Masters Program in Language and Communication Technologies (LCT). University of Amsterdam.
9. **Wittgenstein L. 1958.** Philosophical Investigations, trans. G.E.M. Anscombe, New York: Macmillan.
10. **Landauer T. K. Foltz P. W., Laham D. 1998.** An introduction to latent semantic analysis, Discourse processes, 25, pp. 259–284.
11. **Jiang J. J., Conrath D. W. 1997**. Semantic similarity based on corpus statistics and lexical taxonomy. Proc. of the Int'l. Conf. on Research in Computational Linguistics, pp. 19–33.
12. **Rhee S. K., Lee J., Park M.-W. 2007.** Ontology-based Semantic Relevance Measure. – Proceedings of the The First International Workshop on Semantic Web and Web 2.0 in Architectural, Product and Engineering Design, Busan, Korea. Access mode: http://ceur-ws.org/Vol-294/paper07.pdf.
13. **Anisimov A., Marchenko O., Vozniuk T. 2014.** Determining Semantic Valences of Ontology Concepts by Means of Nonnegative Factorization of Tensors of Large Text Corpora. Cybernetics and Systems Analysis, 50 (3), pp. 327–337.
14. **Nagarajan M., Sheth A. P., Aguilera M., Keeton K., Merchant A., Uysal M. 2007.** Altering Document Term Vectors for Classification – Ontologies as Expectations of Co-occurrence, pp. 1225–1226.
15. **McDonald R., Pereira F., Ribarov K., Hajič J. 2005.** Non-projective dependency parsing using spanning tree algorithms, Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, pp. 523–530.
16. **Davydov M. V. 2014.** A probabilistic search algorithm for finding suboptimal branchings in mutually exclusive hypothesis graph. Int. J. Know.-Based Intell. Eng. Syst. 18, 4, pp. 247–253.
17. **Manning C. D., Schütze H. 1999**. Foundations of Statistical Natural Language Processing, MIT Press, p. 680.
18. **Unger C., Hieber F., Cimiano P. 2010.** Generating LTAG grammars from a lexicon-ontology interface. In: Proceedings of the 10th International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+10), Yale University, pp. 61–68.
19. **Kharbat F. 2011.** A new architecure for translation engine using ontology:one step ahead. In Proc. of The International Arab Conference on Information Technology (ACIT'2011), pp. 169–173.
20. **Miller G. A.1995**. WordNet: A Lexical Database for English. Communications of the ACM, Vol. 38, No. 11, pp. 39–41.
21. Temizsoy M., Cicekli I. 1998. An Ontology Based Approach to Parsing Turkish Sentences. In: Proceedings of AMTA'98-Conference, Springer, October, Langhorne, PA, USA, pp. 124–135.
22. **Davydov M., Lozynska O. 2015.** Spoken and sign language processing using grammatically augmented ontology. Applied Computer Science. ACS journal, Poland, Vol. 11, No. 2, pp. 29–42.
23. **Project UkrParser. 2015.** Access mode: https://github.com/mdavydov/UkrParser.
24. **Princeton University. 2010.** About WordNet. Access mode: http://wordnet.princeton.edu.