Coreference of Deletions – The Case of Control

Giang Linh Nguy, Zdeněk Žabokrtský

Charles University in Prague
Institute of Formal and Applied Linguistics
Malostranské náměstí 25, CZ-11800
{linh, zabokrtsky}@ufal.mff.cuni.cz

Abstract

In the present paper we focus on control as a subtype of anaphora. We work with the theory of control present within the dependency-based framework of Functional Generative Description (FGD[1]), in which control is defined as a relation of a referential dependency between a controller (antecedent – semantic argument of the main clause) and a controllee (anaphor – empty subject of the nonfinite complement (controlled clause)). First this paper presents the rule-based reconstruction of controllees, then, it discusses the perceptron-based determination of the controllees' antecedent. We evaluated our approach on data from the Prague Dependency Treebank 2.0, however, the rules and features of our system are supposed to be language independent and can be tested on other languages in the future.

Keywords

Control, control verb, controller, controllee, antecedent, anaphora, coreference.

1 Introduction

Anaphora resolution is widely studied for its important role in machine translation (MT). We believe that control as a subtype of anaphora can be helpful in MT as well. Consider following English sentences and their translations into Czech:

(1) John$_i$ told Mary$_j$ [Ø$_j$ to come].
Jan$_i$ řekl Marii$_j$, aby (ona$_j$) přišla.
Lit. John told Mary, so that (she) came.

[1] FGD bears numerous resemblances with Meaning-Text Theory, as discussed in (Žabokrtsky,2005)

(2) Mary$_i$ did not agree [Ø$_i$ to come].
Marie$_i$ nesouhlasila, že (ona$_i$) přijde.
Lit. Mary didn't agree, that (she) comes.

(3) Mary$_i$ hates John$_j$ [Ø$_j$ smoking].
Marie$_i$ nesnáší, když Jan$_j$ kouří.
Lit. Mary hates, when John smokes.

The mentioned examples show that the controlled clause can be expressed in one language by an infinitive verb or a gerund verb, whereas in another language, it can be expressed only by a finite verb.

Our work is divided into two steps: first we try to reconstruct controllees automatically by rules. After that we apply a perceptron-based ranker to identify the controllees' antecedent.

Before presenting our approach in Section 5, we discuss the theoretical background of control in Section 2, describe its annotation in the Prague Dependency Treebank in Section 3 and related works in Section 4. Section 6 summarizes experimental results. Conclusions and final remarks follow in Section 7.

To our knowledge, the present paper is one of the few papers, which deals with automatic coreference resolution of deletions in the case of control.

2 Theoretical Background

The terms used in our paper: *verb of control (control verb, governing verb)*, *controller (C-er)*, *controllee (C-ee)*[2], are known from Chomsky's framework of Government and Binding (Chomsky, 1981). In this paper, we work with Panevová's conception of Czech control (Panevová, 2000), in which control is understood in a broader way.

Panevová divides control into two groups: infinitive and nominalized constructions. The infinitive group is further divided into subgroups according to the position of the infinitive and the argument type of the controller. The nominalized group consists of only subgroups according to the argument type of the controller with the nominalized verb in the position of Patient and lacks the division according to the subject position.

Panevová et al. (2002) also presents another classification of control constructions: a combination of control verb and dependent verb both of which can be nominalized. An example of a control construction that can be expressed in all mentioned categories is: 1. *slíbit napsat dopis* (*to promise to write a letter),* 2. *slib napsat dopis* (*a promise to write a letter),* 3. *slíbit napsání dopisu* (*to promise writing of a letter),* 4. *slib napsání dopisu* (*a promise of writing of a letter).*

[2] In Ex. 1, the control verb is *told*, the controller is *Mary$_j$*, and the controllee is the covert argument $Ø_j$.

3 Control in the PDT 2.0

The Prague Dependency Treebank 2.0 (PDT 2.0) is a collection of Czech newspaper texts from 1990s to which a morphological annotation and annotation at two syntactic layers was assigned, at the so called analytical layer (a-layer, at which the surface shape of a sentence is reflected) and at the tectogrammatical layer (t-layer, which captures the linguistic meaning of the sentence); see (Hajič, 2006). Annotation of all three types is available for more than 49,000 sentences, consisting of more than 830,000 tokens.

At the t-layer, the meaning of the sentence is represented as a dependency tree structure (t-tree). Nodes of the tectogrammatical tree (t-nodes) are labeled with t-lemmas and dependency relations (functors[3], semantic roles) and enriched with valency annotation, annotation of semantically relevant grammatical meanings (grammatemes), annotation of topic-focus articulation, and annotation of coreferential relations including control constructions. The rules of t-layer annotation are described in (Mikulová et al., 2007).

In the PDT 2.0, two types of coreferential relations are distinguished: grammatical coreference, which is governed by rules of grammar of the given language, and textual coreference (Panevová, 1991). One kind of grammatical coreference is the relation between the controllee and the controller.

The annotation of control constructions, as of a special subtype of coreferential relations, in PDT 2.0 is based on Panevová's conception of control (Panevová, 2000). The up-to-date technical implementation of coreference in the PDT can be found in (Mikulová et al., 2007)[4]. Each t-node has the following coreferential attributes:

- `coref_gram.rf` – the ID of the grammatical antecedent (antecedents – in the case of conjunction)
- `coref_text.rf` – the ID of the textual antecedent(s)
- `coref_special = segm` (the antecedent is a segment of text) | `exoph` (the case of exophoric reference)

In (Kučová et al., 2003) and (Mikulová et al., 2007), the control classification given in Table 2 was extended by a new type of control – **quasi-control**. Quasi-control can be found within a complex (multi-word) predicate (Cinková, S. & V. Kolářová, 2004), where its verbal part and nominal part share some of their valency modifications. This sharing is called quasi-control.

(4) Jan$_{\text{i-ACT}}$ poskytl Marii$_{\text{j-ADDR}}$ [Ø$_{\text{i-ACT}}$ ochranu Ø$_{\text{j-PAT}}$].
Lit. John provided Mary protection.
John$_{\text{i-ACT}}$ provided [Ø$_{\text{i-ACT}}$ protection Ø$_{\text{j-PAT}}$] for Mary$_{\text{j-ADDR}}$.

[3] Functors represent the semantic values of syntactic dependency relations.

[4] (Panevová et al., 2002) and (Kučová et al., 2003) do not describe the current coreference implementation.

In Ex. 4, *'to provide protection'* is a complex predicate formed by a semantically empty verb *'to provide'* and a noun carrying the main lexical meaning of the entire phrase *'protection'*[5]. The omitted argument Actor of the noun *'protection'* refers to the verb's Actor *'John'* and the noun's non-expressed Patient refers to the verb's Addressee *'Mary'*.

At the t-layer, nodes correspond to autosemantic words only (including pronouns and numerals), prepositions and other functional words have no node in the tree. Besides t-nodes corresponding to surface tokens, in the tree there are newly established (reconstructed) t-nodes that have no counterpart in the outer shape of the sentence. These t-nodes have artificial t-lemmas prefixed by '#' (see their example in Fig. 1).

4 Related Work

There are many types of anaphora which have been actively studied in recent years, e.g. nominal and pronominal anaphora (Yang, 2008; Charniak & Elsner, 2009; Denis & Baldridge, 2009), bridging (indirect) anaphora (Poesio et al., 2004; Vieira et al., 2009), and zero anaphora (Kong & Zhou, 2010; Iisa & Poesio, 2011). Control as a subtype of zero anaphora was resolved in (Kučová et al., 2003) and (Nguy, 2006).

Kučová et al. (2003) provided a rule set for some of control types: if the parent of an infinitive is a verb, then it is a control verb and the controllee refers to one of the control verb's arguments according to the list of control verbs. The list of control verbs was taken from the valency lexicon of Czech verbs VALLEX 1.0 and it includes only three types of control verbs: control verbs with Actor / Addressee / Patient controller[6]. The reported success rate of the rules was the following: ControlRuleACT 69.93%; ControlRuleADDR 88.64% and ControlRulePAT 33.33%.

Nguy (2006) implemented a machine learning approach for the control coreference resolution, but the features given for training a decision tree were gained mainly from the list of control verbs extended by nominalized verbs. First a list of antecedent candidates was created. The list includes effective children of the controllee's effective grandparent[7] (except the controllee's effective parent); in cases of constructions 'to be resolved / able to do' effective children of controllees's great-grandparent; in cases of constructions 'It's possible / necessary to do' effective children of nodes with t-lemma 'možný / nutný / třeba'. Then, features of candidates were extracted. The features set was small, containing the candidate's t-lemma, functor, only one possible candidate option and the agreement of the candidate's and controllee's anaphor with the grandparent's category. Grandparent's categories are lists of control verbs and deverbal nouns. In addition to them are also ambiguous control verb lists – verbs with controllers of different functors or controller's and controllee's functors differ. The agreement of the candidate's and controllee's anaphor with the grandparent's category is then

[5] Its synonymous one-word predicate is 'to protect'.

[6] E.g. 'doporučit'-ADDR – to urge someone$_{i\text{-}ADDR}$ [Ø$_i$ to do something]; 'snažit se'-ACT – someone$_{i\text{-}ACT}$ to try [Ø$_i$ to do something]; 'poslat'-PAT – to send someone$_{i\text{-}PAT}$ [Ø$_i$ to do something]

[7] The "true governor" in terms of dependency relations.

detected by 18 rules. Using the described features, Ngụy trained a decision tree to decide whether a pair of controllee and antecedent candidate are coreferential. The success rate of her approach is 91.53%.

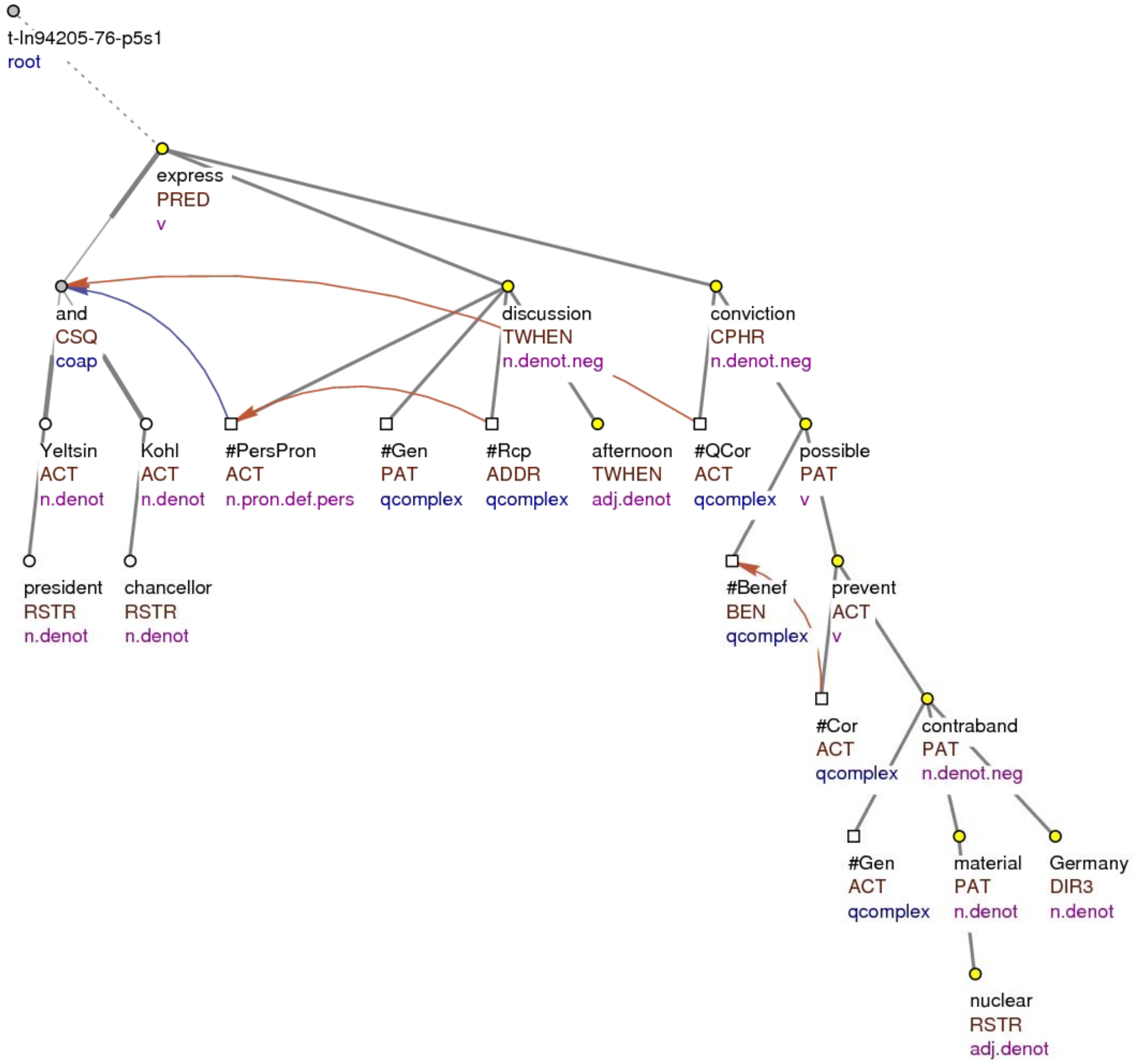

Figure 1: Simplified translated t-tree representing the sentence[8] *Prezident Jelcin a kancléř Kohl vyjádřili po odpoledních jednáních přesvědčení, že lze zabránit pašování jaderného materiálu do Německa*. (Lit.: *President Yeltsin and chancellor Kohl expressed after afternoon discussions the conviction, that it is possible to prevent from contraband of nuclear material to Germany*.)

[8] `#(Q)Cor` represents the (quasi-)controllee in control constructions; `#Benef` represents the beneficiary in control constructions; `#PersPron` represents overt and unstated personal or possessive pronouns (incl. the reflexives); `#Gen` represents a general participant absent at the surface level; and `#Rcp` stands for the omitted argument participating in a reciprocal relation.

5 Control Resolution

Our control coreference resolution task consists of two subtasks: first we have to identify anaphors, in our case the controllees; after that the antecedents, in our case the controllers have to be detected. The resolution for the first subtask is based on a list of t-lemmas of the controllees' effective parent. The second subtask is resolved by using a perceptron-based ranker inspired by Collins (2002).

5.1 Controllee Identification

The controllee identification process relies on the creation of a list of dependent verbs (deverbal nouns) for controllees and quasi-controllees from the training data. The list contains pairs of a dependent verb (noun) lemma and a controllee's functor. There are two independent procedures for identifying controllees and quasi-controllees. The procedure for controllees works as follows: for each infinitive, reconstruct a controllee with the functor, which either was found from the extracted list by the infinitive's lemma or was filled with 'ACT'.

In the case of quasi-controllees, the following simple rule was used: for each node with the functor 'CPHR'[9] and a lemma from the extracted list, reconstruct one or more quasi-controllees with different functors according to the list. [10]

5.2 Controller Detection

For the controller detection we use a simple scoring function: the optimal weight vector of which is estimated by averaged perceptron learning modified for ranking (Nguy et al., 2009). The ranker is trained on the basis of feature vectors for a controllee and its possible antecedents. For every controllee a set of feature vectors containing only one positive instance and negative instances is formed. The positive instance includes features obtained from the controllee and its controller, whereas the negative ones are from the controllee and the non-coreferent phrase.

We consider three possible positions of the controller with respect to the controllee (Fig. 2):

1. the controller is the controllee's 'uncle' (the most frequent case)
2. the controller is the controllee's 'cousin' (in cases of control constructions 'It's possible / necessary to do')
3. the controller is a sibling of the controllees' effective grandparent (in cases of complex control construction[11])

[9] 'CPHR' is filled for the nominal part of a complex predicate.

[10] See the Example (4), in which two quasi-controllees occur: one with 'ACT' and another with 'PAT'.

[11] A complex control construction is meant as a construction of a complex control verb (predicate) + a dependent verb

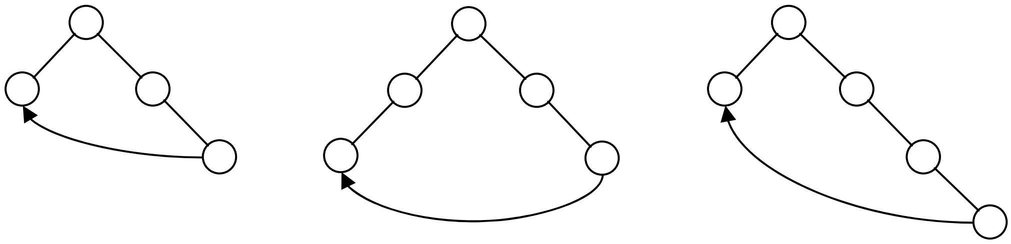

Figure 2: The tree representation of controllers' positions

The training features can be unary and related either to the controllee or to the candidate for the controller or to the controllee's effective parents (control verb and dependent verb), or they can be concatenated to represent the more complex relations between the controllee, the controller, the (complex) control verb (noun) and the dependent verb (noun). Altogether 30 features are used:

- Candidate (i): t-lemma, functor, tree position according to the controllee, semantic POS[12] (sempos), candidate's effective parent (ipar)'s t-lemma
- Controllee (j): t-lemma, functor
- Controllee's effective parent (jpar): t-lemma (lemma), functor (fun), sempos
- Controllee's effective grandparent (jpar2): t-lemma, functor, sempos
- Controllee's effective great-grandparent (jpar3): t-lemma, functor, sempos
- Concatenate(ipar_lemma, i_lemma): concatenation of the t-lemma of the candidate's effective parent and the candidate's t-lemma
- Concatenate(ipar_lemma, i_fun), Concatenate(jpar_lemma, i_fun, j_fun)
- Concatenate(jpar2_lemma, ipar_lemma), Concatenate(jpar2_lemma, i_fun), Concatenate(jpar2_sempos, i_fun), Concatenate(jpar2_lemma, i_fun, j_fun)
- Concatenate(jpar2_lemma, jpar_lemma, i_fun, j_fun), Concatenate(jpar2_lemma, jpar_sempos, i_fun), Concatenate(jpar2_lemma, jpar_sempos, i_fun, j_fun),
- Concatenate(jpar3_lemma, jpar2_lemma, i_fun, j_fun),
- Concatenate(jpar3_lemma, jpar2_lemma, jpar_lemma, i_fun, j_fun)
- Concatenate(jpar2_lemma, ipar_lemma, jpar_lemma, i_fun, j_fun)
- Concatenate(jpar2_lemma, ipar_lemma, i_lemma, jpar_lemma, j_lemma, i_fun, j_fun)

[12] Semantic parts of speech correspond to the basic onomasiological categories.

6 Evaluation

Manually annotated tectogrammatical trees from the PDT 2.0 are used both for training and evaluation purposes. We employ the standard division of the PDT 2.0 into three parts: 80% of data is used for training, 10% for development testing and 10% for evaluation testing. The training data contains 6,598 controllees, 874 development test data and 907 evaluation test data. In the evaluation we used standard metrics with precision, recall and f-measure (Table 1) for controllee identification and controller detection.

Precision = N_c / N_e	Recall = N_c / N_t	F-measure = 2 x P x R / (P + R)
N_c	Number of correctly identified controllees/controllers	
N_e	Number of identified controllees/controllers	
N_t	Number of all controllees/controllers	

Table 1: Evaluation metrics for the control resolution

We applied the following baseline rule for controller detection: for each controllee, select its ‘uncle’ with functor ‘ACT’ as its controller. The scores of rules for the controllee and quasi-controllee identification and the baseline rule and ranker for controller detection are given in Table 2.

	P	R	F
Cor.Ident.Rule	83.381%	86.222%	84.778%
QCor.Ident.Rule	86.219%	85.915%	86.067%
Coref.Baseline	56.065%	57.351%	56.701%
Coref.Ranker	82.161%	84.046%	83.093%

Table 2: Results for the control resolution

The errors of controllee (Cor) identification arise in cases: dependent verb is nominalized (14.525%); Cor was not annotated; Cor was annotated with `#PersPron` or `#Gen` instead. The problem with quasi-controllee (QCor) identification was the recognition of its functor. If the correct recognition of QCor’s functor is not in the task, then the f-measure is 96.075%.

The success rate of the automatic control coreference resolution depends on the previous subtask, the controllee identification. If the control coreference ranker is tested on golden trees (with manually annotated controllees), then it achieves the f-measure of 96.246% and outperforms the system of (Nguy, 2006). The errors of the ranker occur when the controller is a verb or an adjective; or the controller is in another position than those given in Fig. 2.

7 Conclusions

In this paper we report two systems for the automatic resolution of control. One addresses controllee identification and uses hand-written rules based on lists of control verbs. The second system works with a ranker for controller detection and achieves the success rate of 83.093%. The result can approach nearly 96.246% if the controllee identification resolution is significantly improved.

Acknowledgements

The work on this project was supported by the grants MSM0021620838, GA405/09/0729, MŠMT ČR LC536 and GAUK 4383/2009.

Bibliography

Charniak, E. & M. Elsner. 2009. EM works for pronoun anaphora resolution. In: *Proceedings of EACL '09*, 148–156.

Chomsky, N. 1981. *Lectures on Government and Binding: The Pisa Lectures*. Mouton de Gruyter.

Cinková, S. & V. Kolářová. 2004. Nouns as Components of Support Verb Constructions in the Prague Dependency Treebank. In: *Korpusy a korpusová lingvistika v zahraničí a na Slovensku*.

Collins, M. 2002. Discriminative Training Methods or Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In: *Proceedings of EMNLP '02*, Vol. 10, 1–8.

Denis, P. & J. Baldridge. 2008. Specialized models and ranking for coreference resolution. In: *Proceedings of EMNLP '08*, 660–669.

Hajič J., J. Panevová, E. Hajičová, P. Sgall, P. Pajas, J. Štěpánek, J. Havelka, M. Mikulová, Z. Žabokrtský & M. Ševčíková-Razímová. 2006. *Prague Dependency Treebank 2.0*. Software prototype, Linguistic Data Consortium, Philadelphia, PA, USA, ISBN 1-58563-370-4, www.ldc.upenn.edu.

Iida, R. & M. Poesio. 2011. A Cross-Lingual ILP Solution to Zero Anaphora Resolution. In: *Proceedings of ACL '11*, 60–69.

Kong, F. & G. Zhou. 2010. A tree kernel-based unified framework for Chinese zero anaphora resolution. In: *Proceedings of EMNLP '10*, 882–891.

Kučová, L., V. Kolářová, Z. Žabokrtský, P. Pajas & O. Čulo. 2003. *Anotování koreference v Pražském závislostním korpusu*, Prague: MFF UK, TR-2003-19.

Mikulová, M., A. Bémová, J. Hajič, E. Hajičová, J. Havelka, V. Kolářová, L. Kučová, M. Lopatková, P. Pajas, J. Panevová, M. Ševčíková, P. Sgall, J. Štěpánek, Z. Urešová, K. Veselá

& Z. Žabokrtský. 2007. *Annotation on the tectogrammatical level in the Prague Dependency Treebank*. Technical report no. 2007/3.1, ÚFAL, Charles University.

Nguy, G. L. 2006. *Proposal of a Set of Rules for Anaphora Resolution in Czech*. Master's thesis, Faculty of Mathematics and Physics, Charles University.

Nguy, G. L., V. Novák & Z. Žabokrtský. 2009. Comparison of Classification and Ranking Approaches to Pronominal Anaphora Resolution in Czech. In: *Proceedings of the SIGDIAL 2009 Conference*, 276–285.

Panevová, J. 1991. Koreference gramatická nebo textová? In: *Etudes de linguistique romane et slave*. Krakow.

Panevová, J. 2000. More Remarks on Control. In: *Prague Linguistic Circle Papers*, Vol. 2, 101–120. Amsterdam/Philadelphia: Benjamins Publishing House.

Panevová, J., V. Řezníčková & Z. Urešová. 2002. The theory of control Applied to the Prague Dependency Treebank (PDT). In: *Proceedings of the 6th International Workshop on Tree Adjoining Grammars and Related Frameworks*, 175–180. Università di Venezia.

Poesio, M., R. Mehta, A. Maroudas & J. Hitzeman. 2004. Learning to resolve bridging references. In: *Proceedings of ACL '04*.

Sgall, P. 1967. *Generativní popis jazyka a česká deklinace*. Praha, Academia.

Sgall, P., E. Hajičová & E. Panevová. 1986. *The Meanings of the Sentence in Its Semantic and Pragmatic Aspects*. Dordrecht, D. Reidel Publishing Company - Praha, Academia.

Sgall, P. et al. 1986. *Úvod do syntaxe a sémantiky*. Praha, Academia.

Vieira, R., E. Bick, J. Coelho, V. Muller, S. Collovini, J. Souza & L. Rino. 2009. Semantic tagging for resolution of indirect anaphora. In: *Proceedings of SigDIAL '06*, 76–79.

Yang, X., J. Su & C. L. Tan. 2008. A twin-candidate model for learning-based anaphora resolution. In: *Computational Linguistics*, Vol. 34, 3, 327–356.

Žabokrtský, Z. 2005. Resemblances between Meaning-Text Theory and Functional Generative Description. In: *Proceedings of the 2nd International Conference of Meaning-Text Theory*, 549–557. Slavic Culture Languages Publishers House, Moskva, Russia, ISBN 5-9551-0094-6.