Linguistic **I**ssues in **L**anguage **T**echnology – **LiLT**
Submitted, January 2012

Bulgarian-English Treebank

Design and Implementation

Kiril Simov and Petya Osenova

Published by CSLI Publications

LiLT volume 7, issue 14 January 2012

Bulgarian-English Treebank

Design and Implementation

Kiril Simov and Petya Osenova, *Linguistic Modelling Department, IICT, Bulgarian Academy of Sciences*

Abstract

The paper describes the construction of a Bulgarian-English treebank aligned on the word and semantic level. We consider the manual word level alignment easier and more reliable than the manual alignment on syntactic and semantic level. Thus, after manual word level alignment we apply an automatic procedure for the construction of semantic level alignments. Our work presents the main steps of this automatic procedure which exploits the syntactic analysis of both sentences, morphosyntactic annotation, manual word level alignment in producing the semantic annotation of the sentences and semantic alignment. Last, but not least, a method for identification of potential errors is discussed using the automatically constructed semantic analyses of Bulgarian sentences and their comparison to the semantic representation of the English sentences.

LiLT Volume 7, Issue 14, January 2012.
Bulgarian-English Treebank.

1 Introduction

In this paper we report on the design of the annotation schema of the Bulgarian-English Parallel Treebank (BulEngTreebank) and semi-automatic error correction of the automatic dependency analysis of the Bulgarian sentences. The main goal for the construction of the treebank is for it to be used as a source for learning of statistical transfer rules for Bulgarian-English machine translation along the lines of Bond et al. (2011). The transfer rules in this framework are rewriting rules over MRS (Minimal Recursion Semantics) structures. The basic format of the transfer rules is:

$$[C :]I[!F] \rightarrow O$$

where I is the *input* of the rule, O is the *output*, C determines the *context* and F is the *filter* of the rule. C selects positive context and F selects negative context for the application of a rule — for more details see Oepen (2008). Thus, the treebank has to contain parallel sentences, their semantic analyses and correspondences on the level of MRS structures.

In the development of such a parallel treebank we rely on the Bulgarian HPSG resource grammar BURGER, and on a dependency parser (Malt Parser — Nivre et al. (2006)), trained on the dependency version of BulTreeBank — Simov et al. (2004). For English we rely on the English Resource Grammar (ERG) — Flickinger (2000). All three parsers produce semantic representations in terms of MRS. The treebank is a parallel resource aligned first on a sentence level. Then the alignment is done on the level of MRS structures. The motivation for this choice is the fact that MRS analyses abstract over the syntactic analyses. Also MRS level alignment is appropriate for learning of correspondence rules of the type presented above. The annotation procedure is as follows: first, the Bulgarian sentences are parsed with BURGER. If it succeeds, then the resulting MRS structures are used for the alignment. If BURGER fails, the sentences are parsed with Malt Parser, and then MRS structures are constructed on the basis of the dependency analysis. The main problem of this approach to semantic alignment is the level of errors produced by the parser. These errors are of two kinds — (1) genuine errors — the parser fails to produce an analysis or it selects a wrong analysis; (2) possible, but inappropriate readings — the parser selects one good analysis, but it is not the one that corresponds to the English translation.

With respect to the MRS level alignments, a very pragmatic approach has been adopted — the MRS level alignments originated from the word level alignment. This approach is based on the following ob-

servations and requirements:

- Both approaches for generation of MRS are lexicalized;
- Non-experts in MRS can do the alignments successfully on the word level;
- Different rules for generation/testing are possible.

Both parsers (for Bulgarian and English) are lexicalized in their nature. They first assign elementary predicates to the lexical elements, and then, on the basis of the syntactic analysis, these elementary predicates are composed into MRSes for the phrases, and the whole sentence. Our belief is that by having alignments on the word level, syntactic analyses and the rules for composition of MRS structures, we will be able to determine correspondences between bigger MRS structures than only at the lexical level, using the ideas of Tinsley et al. (2009). They establish the mapping on word level (automatically), then for candidate phrases they calculate their rank of correspondences on the basis of the word alignment.

The alignment on word level allows us to do more reliable alignments using annotators who are non-experts in MRS analyses. The inter-annotator agreement is 92%. This kind of alignment does not require any initial knowledge of MRS from the annotators. Another advantage is that the result might be used for training tools for automatic word alignment. Additionally, word level alignment might be done before the actual analysis of the sentences. This is useful the in case of Bulgarian, where the BURGER grammar does not have enough coverage yet.

The treebank includes the following subsets: (1) English Resource Grammar datasets (CSLI dataset, MRS dataset and Norwegian-English parallel touristic texts), translated into Bulgarian, and (2) Parallel texts from SETIMES news corpus[1]. The manually annotated and checked data are about 105 000 tokens.

In this paper we present the levels of the annotation of the treebank, the type of rules for construction of MRS structures over dependency parses and the procedure for error detection for manual editing of the dependency analyses of Bulgarian sentences.

2 Background and Related Work

Our approach is inspired by the work on MRS and RMRS (see Copestake (2003) and Copestake (2007)) and the previous work on transfer of dependency analyses into RMRS structures described in Spreyer and

[1] http://opus.lingfil.uu.se/SETIMES.php

Frank (2005) for TIGER treebank of German, and Jakob et al. (2010) Prague Dependency Treebank of Czech (PDT). The work on PDT presented in Jakob et al. (2010) first assigns elementary predications to each node in the tectogrammatical tree. Then the elementary predications for the nodes are combined on the basis of the dependency annotation. A similar approach is taken by us, except that the analyses from which we start are not trees on the tectogrammatical level.

MRS is introduced as an underspecified semantic formalism Copestake et al. (2005). It is used to support the semantic analyses in the HPSG English grammar — ERG, Copestake and Flickinger (2000), but also in other grammar formalisms, such as LFG. The main idea is for the formalism to rule out spurious analyses resulting from the representation of logical operators and the scope of quantifiers. Here we will present only basic definitions from Copestake et al. (2005). For more details the cited publication should be consulted. An MRS structure is a tuple <GT, R, C>, where GT is the top handle, R is a bag of EPs (elementary predicates) and C is a bag of handle constraints, such that there is no handle h that outscopes GT. Each elementary predication contains exactly four components: (1) a handle which is the label of the EP; (2) a relation; (3) a list of zero or more ordinary variable arguments of the relation; and (4) a list of zero or more handles corresponding to scopal arguments of the relation (i.e., holes). Here is an example of an MRS structure for the sentence *"Every dog chases some white cat."*

<h0, {h1:every(x,h2,h3), h2:dog(x), h4:chase(x, y),
h5:some(y,h6,h7), h6:white(y), h6:cat(y)}, {}>

The top handle is h0. The quantifiers are represented as relations $every(x, y, z)$ and $some(x, y, z)$ where x is the bound variable, y and z are handles determining the restriction and the body of the quantifier. The conjunction of two or more relations is represented by sharing the same handle (h6 above). The outscope relation is defined as a transitive closure of the immediate outscope relation between two elementary predications — EP immediately outscopes EP' iff one of the scopal arguments of EP is the label of EP'. In the example the set of handle constraints is empty, which means that the representation is underspecified with respect to the scope of both quantifiers.

RMRS is introduced as a modification of MRS which captures the semantics resulting from the shallow analysis. The main assumption is that the shallow processor does not have access to a lexicon. Thus it does not have access to arity of the relations in EPs. Therefore, the representation has to be underspecified with respect to the number of arguments of the relations. Additionally, the forming of the relation names follows conventions that provide possibilities to construct a cor-

rect semantic representation on the basis of information provided by a POS tagger, for example. The arguments are introduced separately by argument relations between the label of a relation and the argument. The names of the argument relations follow some standardized convention like RSTR, BODY, ARG1, ARG2, etc. These argument relations are grouped in a separate set in a given RMRS structure. Both representations MRS and RMRS could be transferred to each other under certain conditions. In the paper we follow the representation of RMRS used in Jakob et al. (2010), which defines an RMRS structure as a quadruple < *hook, EP-bag, argument set, handle constraints* >, where a hook consists of three elements $l{:}\,a{:}\,i$, l is a label, a is an anchor and i is an index. Each elementary predication is additionally marked with an anchor — $l{:}\,a{:}\,r(i)$, where l is a label, a is an anchor and $r(i)$ is a relation with one argument of the appropriate kind — referential index or event index. The argument set contains argument statements of the following kind $a{:}\,ARG(x)$, where a is an anchor which determines for which relation the argument is defined, ARG is the name of the argument, and x is an index or a hole variable or handle (h) for scopal predicates. The handle constraints are of the form $h =_q l$, where h is a handle, l is a label and $=_q$ is the relation expressing the constraint similarly to MRS. $=_q$ sometimes is written as qeq.

3 Word Level Alignment

The annotation guidelines for Bulgarian-English word alignment benefitted from the tradition established by the guidelines used in similar projects, such as the Blinker project for English-French alignment (Melamed, 1998), the alignment task for the Prague Czech-English Dependency Treebank (Kruijff-Korbayová et al., 2006), the Dutch parallel Corpus project (Macken, 2010). Different linguistic theoretical backgrounds appear to be another source of divergence that affects the rules of phrase alignments as well as the specific grammatical techniques. This holds especially in correspondences between synsemantic words (like prepositions, determiners, particles, auxiliary verbs) and synsemantic and/or autosemantic words (Macken, 2010). In our work we distinguish between *strong* and *weak* alignment (Kruijff-Korbayová et al., 2006), so P-link either represents *weak* alignment, or that the annotator is *uncertain* about the pairing, or both. S-link either shows *strong* alignment, or shows that the annotator is *certain* about the pairing, or both.

Here we present only the general rules of our word level alignment guidelines. We adopt the general rules that have proven to be shared by

the different annotation tasks and alignment strategies. The number of corresponding tokens to be aligned can be estimated by following two rules (Macken (2010) and references there in):

1. Mark as many tokens as necessary in the source and in the target sentence to ensure a two-way equivalence.
2. Mark as few tokens as possible in the source and in the target sentence, but preserve the two-way equivalence.

If a token or a phrase has no corresponding counterpart in the other language, it should be left unlinked (NULL link) (Melamed, 1998).

Idioms and free translations present a special case. If two autosemantic words or phrases refer to the same object, but do not share the same meaning, they are aligned with a P-link, e.g.:

(1) this *animal*
tova *kuche*
this dog

The rule holds when there is a synsemantic–autosemantic correspondence:

(2) *Ivan 's* mother called.
Negovata majka se obadi.
His mother called.

P-link: Ivan 's = Negovata

P-link is used when a lexical item is paraphrased in the other language:

(3) these *non-Serbs*
tezi *lica ot nesrybski proizhod*
these persons from a non-Serbian origin

P-link: non-Serbs = lica ot nesrybski proizhod

Idioms are linked with an S-link; each token from the idiom in the source sentence is aligned with each token from the idiom in the target sentence.

(4) She'll marry him *when pigs begin to fly.*
Tya ste se omyzhi za nego *na kukovo lyato.*

S-link: when pigs begin to fly = na kukovo lyato

Besides these general rules we have a list of more than 50 mapping rules specific to Bulgarian-English which will not be presented here. They ensure that the RMRS assigned by the grammars to the lexical items in the sentences are equivalent if the corresponding items are aligned on the word level.

4 Bulgarian Dependency Parsing and RMRS Analysis

Many parsers have been trained on data from BulTreeBank — Simov et al. (2004). Especially successful was the MaltParser of Joakim Nivre — Nivre et al. (2006). It works with 87.6 % accuracy. In Table 1 the dependency tagset related to the Dependency part of the BulTreeBank is presented.

adjunct	12009	Adjunct (optional verbal argument)
clitic	2263	Short forms of the possessive pronouns
comp	18043	Complement (argument of non-finite verbs, copula, auxiliaries)
conj	6342	Conjunction in coordination
conjarg	7005	Argument (second, third, ...) of coordination
indobj	4232	Indirect Object
marked	2650	Marked (clause, introduced by a subordinator)
mod	42706	Modifier (dependants which modify nouns, adjectives, adverbs, ...)
obj	7248	Object (direct argument of a non-auxiliary verbal head)
pragadjunct	1612	Pragmatic adjunct
punct	28134	Punctuation
subj	14064	Subject
xadjunct	1826	Clausal adjunct
xcomp	4651	Clausal complement
xmod	2219	Clausal modifier
xprepcomp	168	Clausal complement of preposition
xsubj	504	Clausal subject

TABLE 1 Dependency relations in BulTreeBank.

In addition to the dependency tags we also have morphosyntactic tags attached to each word. For each lexical node the lemma is assigned. The number under the name of each relation indicates how many times the relation appears in the dependency version of BulTreeBank.

In the rest of the section we present some of the rules for transfer of dependency parses into RMRS representations. The input for the RMRS structures includes the following linguistic annotations — the lemma (*Lemma*) for the given wordform; the morphosyntactic tag (*MSTag*) of the wordform, and the dependent relations. Thus, the algo-

rithm for producing an RMRS from a dependency parse is implemented via two types of rules:

< Lemma, MSTag > → EP-RMRS

The rules of this type produce an RMRS including an elementary predicate.

< DRMRS, Rel, HRMRS > → HRMRS'

The rules of this type unite the RMRS constructed for a dependent node (*DRMRS*) into the current RMRS for a head node (*HRMRS*). The union (*HRMRS'*) is determined by the relation (*Rel*) between the two nodes. In the rest of the section we present examples of these rules.

First, we start with assigning EPs for each lemma in the dependency tree. These EPs are similar to node EPs of Jakob et al. (2010). Each EP for a given lemma consists of a predicate generated on the basis of the lemma string. On the basis of the part-of-speech tag the type of ARG0 is determined — referential index or event index. After this initial step the basic RMRS structure for each lemma in the sentence is compiled. Below we discuss the exploitation of the rest of the information in the dependency tree — the types of links to the other lemmas as well as the further contributions of the morphosyntactic features. Here is an example for the verb 'cheta' (to read):

Rule:
<lemma, Vp> →
< l1:a1:e1, {l1:a1:lemma_v_rel(e1)}, {a1:ARG1(x1)}, {} >
Application:
<cheta, Vp> →
< l1:a1:e1, {l1:a1:cheta_v_rel(e1)}, {a1:ARG1(x1)}, {} >

In this example we also include information for the unexpressed subject (ARG1) which is always incorporated in the verb form. In case the subject is expressed, it will be connected to the same referential index. For some types of nodes the EP RMRS will include information only for arguments of the predicate of the head node.

The short forms of pronouns (clitics) contribute semantically in the following way: the semantics of the argument-bearing clitics (accusative and dative) is incorporated into the verb semantics, while the modifying ones (possessive) introduce explicitly a semantic relation. Thus, in the former case the explicit semantic relation is established only by the full counterparts. It is a rather straightforward transfer, since the short forms are annotated as clitics, while the full forms are assigned grammatical roles — object or indirect object. Our analysis is close to the

one, taken in the Modern Greek Grammar[2]. The full form complements are automatically represented as ARG2 and ARG3 of the corresponding verb. In this transfer we always connect the object to the argument ARG2 slot and the indirect object to the ARG3 slot. For example, the sentence *cheta mu ya* (Read-I him-dative her-accusative, 'I read it to him') will have the following representation:

Rule (*accusative clitic*):
< <l2:a2:x2, {}, {a2:ARG2(x2)}, HC1>, comp,
< l1:a1:e1, {l1:a1:lemma_v_rel(e1)|R}, ARGS, HC2 > >
→
< l1:a1:e1, {l1:a1:lemma_v_rel(e1)|R},
{a1:ARG2(x2)} ∪ ARGS, HC1 ∪ HC2 >
Application:
< <l2:a2:x2, {}, {a2:ARG2(x2)}, {}>, comp,
< l1:a1:e1, {l1:a1:cheta_v_rel(e1)}, {a1:ARG1(x1)}, {} > >
→
< l1:a1:e1, {l1:a1:cheta_v_rel(e1)},
{a1:ARG1(x1), a1:ARG2(x2)}, {} >

Rule (*dative clitic*):
< <l3:a3:x3, {}, {a3:ARG3(x3)}, HC1>, comp,
< l1:a1:e1, {l1:a1:lemma_v_rel(e1)|R}, ARGS, HC2 > >
→
< l1:a1:e1, {l1:a1:lemma_v_rel(e1)|R},
{a1:ARG3(x3)} ∪ ARGS, HC1 ∪ HC2 >
Application:
< <l3:a3:x3, {}, {a3:ARG3(x3)}, {}>, comp,
< l1:a1:e1, {l1:a1:cheta_v_rel(e1)},
{a1:ARG1(x1), a1:ARG2(x2)}, {} > >
→
< l1:a1:e1, {l1:a1:cheta_v_rel(e1)},
{a1:ARG1(x1), a1:ARG2(x2), a1:ARG3(x3)}, {} >

The EP RMRS for the accusative clitic introduces only ARG2 and appropriate grammatical features for the variable x2 (third person, singular, feminine). The EP RMRS for the dative clitic provides ARG3 and its grammatical features (third person, singular, masculine). After being incorporated into the head RMRS, the anchors for the ARG2 and ARG3 are changed with respect to the anchor of the head.

The subject is mapped to ARG1. It is worth noting that the Subject

[2]http://www.delph-in.net/mgrg/documentation.pdf

argument is partially determined during the previous step in building EPs, because Bulgarian is a pro-drop language, and the main subject properties are considered part of the verb form. Here is an example for the sentence *momche mu ya chete* (Boy him-dative her-accusative read, 'A boy reads it to him)[3]:

< l2:a4:e1,
{l1:a1:momche_n_rel(x1), l2:a4:cheta_v_rel(e1)},
{a4:ARG1(x1), a4:ARG2(x2), a4:ARG3(x3)}, {} >

Another example shows an explicit direct object for the sentence *momche mu chete kniga* (Boy him-dative reads book, 'A boy reads a book to him'):

< l2:a3:e1,
{l1:a1:momche_n_rel(x1), l2:a3:cheta_v_rel(e1),
l3:a4:kniga_n_rel(x2)},
{a3:ARG1(x1), a3:ARG2(x2), a3:ARG3(x3)}, {} >

These rules are applied by traversing the dependency tree top-down for constructing lexical EPs and then bottom-up to incorporate the dependent RMRS in the head RMRS. The main algorithm **rmrs** selects the root of the tree and calls the recursive function to calculate the RMRS for the sentence:

algorithm **rmrs**
Input: *DTree* (dependency tree in CoNLL format)
Output: < *hook, EP-bag, argument set, handle constraints* >
(RSMS structure for the sentence)
RootNode ← root(*DTree*);
setEnumerators();
RMRS ← nodeRMRS(*DTree, RootNode*);
return addQuantifiers(*RMRS*);
end_algorithm

The function root(*DTree*) selects the root of the tree. The function nodeRMRS(*DTree, Node*) recursively constructs the RMRS structure for the subtree starting at node *Node*. The subtree is part of the whole tree for the sentence — *DTree*. The function setEnumerators() sets the initial numbers for labels, referential and event variables. For anchors we use the token numbers that are already in the CoNLL format of the dependency tree. The function addQuantifiers(*RMRS*) introduces the missing quantifiers in the final RMRS. Here is the pseudo code for the function:

[3]We represent only the results of the application of the rules.

```
function nodeRMRS(DTree, CurrentNode)
    NodeEP ← nodeEP(DTree, CurrentNode);
    for DNode ∈ depNodes(DTree, CurrentNode)
        DNodeRMRS ← nodeRMRS(DTree, DNode);
        DRel ← nodeRel(DTree, DNode);
        NodeEP ← union(NodeEP, DNodeRMRS, DRel);
    end_for
    return NodeEP;
end_function
```

This function first calls the function for constructing the RMRS for the elementary predication for the current node in the dependency tree — nodeEP(*DTree, CurrentNode*). This function implements the first kind of rule mentioned above. It has access to the lemma and the grammatical information for the current node. The predicate name is constructed on the basis of the lemma and the part of speech (for example, cheta_v_rel — 'read'); the argument type is determined on the basis the grammatical information — event or referential index. Additional information can be added for other arguments of the verbs as described above. In case of access to a lexicon, the function will be tuned to the information within the lexicon. This will be relevant for the case of the valency lexicon.

The function depNodes(*DTree, CurrentNode*) returns a set of nodes in the tree which are dependent on the current node. For each of them the function nodeRMRS(*DTree, Node*) is called. The result of this recursive call is incorporated within the current RMRS on the basis of the dependency relations. This is done by the function union(*NodeEP, DNodeRMRS, DRel*). This function is defined by the second kind of rule described above. Note that all the relevant information is available in the already constructed RMRS structures for the head node and the dependent nodes and the relation. The implementation is done within the CLaRK system[4].

5 Semantic Level Alignment

As mentioned above, we use word level alignment in order to establish alignment on the level of the RMRS. For both languages the phrases are assigned an RMRS structure which represents the semantic value of the phrase (in the case of the dependency parse this MRS incorporates the semantic values of all dependent elements). The intuition behind our approach is that the lexical data of each structure in the syntactic analysis for a pair of sentences are aligned on the word level. Then we

[4] http://www.bultreebank.org/clark/index.html

assume that their MRS structures are equivalent modulo the meaning of the language-specific elementary predicates. We exploit this intuition in constructing the semantic alignment in our treebank.

First we establish correspondences on the lexical level. Each pair of lexical items in the corresponding analyses are made equivalent on the basis of word alignment. The next step is to traverse the trees bottom-up. For each phrase or head for which the components are aligned, a correspondence on the MRS level is established. It should be explicitly noted that a correspondence on the sentence level is also established. Here we present an example. Let us consider the following pair of sentences from the English Resource Grammar datasets:

(5) Kucheto na Braun lae.
Dog-the(neut) of Browne barks.
Browne's dog barks.

The word level alignment is:
(**Kucheto** = **dog**)
(**na** = **'s**)
(**na Braun** = **Browne 's**)
(**lae** = **barks**)
(**Braun** = **Browne**)

Here are the RMRS structures assigned to both sentences represented only as bag of relations. The arguments are also represented in the usual way:

English
{ h3:proper_q_rel(x3,h4,h6), h7:named_rel(x5,"Browne"),
h8:def_explicit_q_rel(x10, h9, h11), h12:poss_rel(e13,x10,x5),
h12:dog_n_1_rel(x10), h14:bark_v_1_rel(e2,x10) }

Bulgarian
{ h3:kuche_n_1_rel(x4), h3:na_p_1_rel(e5,x4,x6),
h7:named_rel(x6, "Braun"), h8:exist_q_rel(x6, h9, h10),
h11:exist_q_rel(x4, h12, h13), h1:laya_v_rel(e2,x4)}

The result of correspondences between the RMRS on the basis of word level establishes the following mappings of elementary predicate lists:

(*m1*)
(**Braun** = **Browne**)
{h7:named_rel(x6,"Braun"), h8:exist_q_rel(x6,h9,h10)}
to
{h3:proper_q_rel(x5,h4,h6), h7:named_rel(x5,"Browne")}
(*m2*)
(**na** = **'s**)

{h3:na_p_1_rel(e5,x4,x6)}
to
{h12:poss_rel(e13,x10,x5)}

(*m3*)
(**na Braun** = **Browne 's**)
{h3:na_p_1_rel(e5,x4,x6), h7:named_rel(x6,"Braun"),
h8:exist_q_rel(x6,h9,h10)}
to
{h3:proper_q_rel(x5,h4,h6), h7:named_rel(x5,"Browne"),
h8:def_explicit_q_rel(x10,h9,h11),
h12:poss_rel(e13,x10,x5)}

(*m4*)
(**Kucheto** = **dog**)
{h3:kuche_n_1_rel(x4), h11:exist_q_rel(x4,h12,h13)}
to
{h12:dog_n_1_rel(x10)}

(*m5*)
(**lae** = **barks**)
{h1:laya_v_rel(e2,x4)}
to
{h14:bark_v_1_rel(e2,x10)}

Our goal is to have RMRS alignment not just on the word level, but also on the phrase level. Thus, using the correspondences described above and the syntactic analyses of both sentences we can infer the following mapping:

(*m6*)
(**Kucheto na Braun** = **Browne 's dog**)
{h3:na_p_1_rel(e5,x4,x6), h7:named_rel(x6,"Braun"),
h8:exist_q_rel(x6,h9,h10), h3:kuche_n_1_rel(x4),
h11:exist_q_rel(x4,h12,h13)}
to
{h3:proper_q_rel(x5,h4,h6), h7:named_rel(x5,"Browne"),
h8:def_explicit_q_rel(x10,h9,h11),
h12:poss_rel(e13,x10,x5), h12:dog_n_1_rel(x10)}

This automatic procedure for inferring semantic correspondences on the phrasal level provide flexability for different strategies for such alignments. For example, such correspondences might be equipped with similarity scores on the basis of word alignment types involved in the corresponding phrase, as well as the type of the phrase itself. For instance, if the word alignment of two corresponding phrases involves only *sure* links, then the MRS level alignment for these phrases also is assumed to be *sure*. Respectively, if on the word level there are *unsure*

links, then the MRS level alignment could be assumed to be *unsure*. This idea could be developed further depending on the application. Also, in some cases the MRS level alignment could be assumed to be *sure*, although it includes some *unsure* links on word level. For example, in case of analytical verb forms many elements will be aligned only by possible links, but the whole forms are linked as a *sure* correspondence. We believe that such pairs of sentences with appropriate syntactic and semantic analyses and word alignment are a valuable source for construction of alignments on the semantic level.

6 Error Detection

As mentioned above, the automatic step of dependency parsing introduces errors in the RMRS alignment. Currently we are not able to check the whole treebank manually because of time limits. Thus, we had to locate the most striking errors. The errors are grouped on several levels. The first group is the case where the RMRS rules, presented above, fail to produce an RMRS structure for a given dependency tree. Such cases are currently about 14% of the cases in two of the datasets distributed together with ERG — the MRS dataset and the CSLI dataset. These errors are easy to spot and the reasons for the failure are: (1) total erroneous dependency analyses; or (2) RMRS rule failure. For the first case of errors we are currently working on improving the dependency parsing with additional information from the partial parsing module. For the second type of errors we extend the set of RMRS rules.

More interesting errors are the cases where the RMRS rule system produces an RMRS structure for a given dependency parse, but the RMRS structures for the Bulgarian and the English sentences are not compatible. Here under compatibility of RMRS structures we understand the following situations: 1. there is no contradiction between them, i.e. the relations in both structures are translated in one common representation (mapping to the same ontology, for example) and 2. there is a mapping between the variables which ensures the sharing of the same individuals by the models. Obviously, we are not ready to provide a procedure which ensures full compatibility. Thus, we have implemented a weaker version. Our idea is based on observations over a mapping of MRS structures produced by the ERG and BURGER over the MRS dataset. The assumption is that the corresponding structures agree on the correspondences of the elementary predicates, their arguments and modifiers. Therefore, we have designed a procedure which implements this intuition.

First, we ignore all the quantifiers in both RMRS structures. Then,

for each word level alignment we construct correspondences on the level of predicate(s) introduced by the RMRS rules and the corresponding variables and arguments. The word level alignment, as it was described above, provides a mapping on the minimal level of correspondences. This ensures that the mapping of the elementary predicates, the variables and the arguments is possible. For each phrasal RMRS we join the correspondences for the given substructures. For each two word level elements in one of the sentences that are aligned, two word level elements in the other sentence have to be connected in the same way. This means that arguments of a given predicate in the Bulgarian RMRS are aligned with the corresponding arguments of the appropriate predicate in the English RMRS. Similarly, the modifiers of a predicate in one RMRS correspond to the modifiers in the other. This procedure identifies the following discrepancies between the RMRS structures: (1) the arguments of compatible relations mix their arguments; (2) the argument of a given relation is mapped onto a modifier in the other structure; (3) a modifier of a given predicate in one structure is mapped to a modifier of a different predicate in the other structure. In our case such a discrepancy identifies wrong attachments in the dependency tree.

These discrepancies correspond to three cases. First, there is the case of genuine errors. For example, an argument in the English sentence is analyzed as a modifier in the Bulgarian sentence. For example, in the sentence *A boy read a book* the phrase *a book* is analyzed as an adjunct of *read*. Second, there is the case of possible, but inappropriate readings. In our case this corresponds to mixing of modifiers of different relations on the basis of a wrong PP attachment. For example, in *Ejbrams pokaza ofisa na Braun* (Abrams showed the office to Browne), the Bulgarian PP *na Braun* ('to Brown' or 'of Brown') is attached to *ofisa* (the office) instead of to the verb *pokaza* (showed). This new reading is acceptable, but does not agree with the English reading. Third, there is the case where the different attachments do not make any difference in the interpretation. Examples are sentences like *They discussed it at the meeting in Berlin*. At least in Bulgarian the attachment of the PP *in Berlin* to the verb phrase *discussed it* or to the noun phrase *the meeting* does not change the interpretation that the event of discussion was in Berlin at a meeting[5].

We have tested the procedure on the basis of the CSLI dataset of ERG containing 844 sentence pairs. The number of cases in which the procedure indicated a problem of this kind is 21 %. About 2/3 of the cases represent the wrong attachment of a modifier. This approach

[5]Similar examples are presented in the Prague Dependency Treebank.

ensures that after some manual inspection and subsequent repairing of the dependency parses, the main sources of errors are detected and the aligned parallel treebank shows improvement in the agreement of the arguments and modifiers of predicates.

7 Conclusion

In this paper we presented the annotation approach for a parallel Bulgarian-English treebank. The semantic level annotation is automatically constructed via an HPSG English grammar and a hybrid architecture for Bulgarian — an HPSG Bulgarian grammar or a dependency parser. The hybrid approach is necessary because the Bulgarian HPSG grammar does not have enough coverage yet. However, such an automatic semantic analysis is inevitably open to errors. We have presented an approach to identify some of these errors via comparing the semantic representation for the sentences in each pair. In our current work we assume that the English analysis is the correct one and the Bulgarian one has to agree with it.

An alternative strategy for semantic alignment, suggested by one of the reviewers, might be the transfer of English MRS structures onto the Bulgarian aligned data. We are thankful for this suggestion, since this approach could be very useful as a method for the creation of correspondences between Bulgarian words and phrases and English MRS structures. However, for the task of machine translation from Bulgarian to English using a Bulgarian grammar, such correspondences could lead to problems. First, the Bulgarian grammar would not be able to produce English MRS structures since the relation names in MRS are language-dependant. Second, the conceptualization of the meaning for the grammars of both languages could be on different levels of granularity. Thus, we chose to construct Bulgarian to English MRS level alignments.

In our view the suggested strategy could be used in cases where there are enough word level aligned texts, and for one of the languages there is a grammar producing MRS analyses, and there is no such grammar for the other language. In such a case one would be able to construct automatically a parallel treebank aligned on the basis of the word/phrase to MRS structure correspondences. This treebank could be used to train a statistical translation system from MRS structures to text. We consider this possibility as an interesting direction for future work.

Acknowledgments

This work has been supported by the European project EuroMatrixPlus (IST-231720). The authors would also like to thank the three anonymous reviewers for their very useful critical reviews as well as Dan Flickinger for kindly agreeing to proof-read the final version.

References

Bond, Francis, Stephan Oepen, Eric Nichols, Dan Flickinger, Erik Velldal, and Petter Haugereid. 2011. Deep open-source machine translation. *Machine Translation* 25(2):87–105.

Copestake, Ann. 2003. Robust minimal recursion semantics (working paper).

Copestake, Ann. 2007. Applying robust semantics. In *Proceedings of the 10th Conference of the Pacific Assocation for Computational Linguistics (PACLING)*, pages 1–12.

Copestake, Ann and Dan Flickinger. 2000. An open source grammar development environment and broad-coverage english grammar using hpsg. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*. Athens, Greece.

Copestake, Ann, Dan Flickinger, Carl Pollard, and Ivan Sag. 2005. Minimal recursion semantics: An introduction. *Research on Language & Computation* 3(4):281–332.

Flickinger, Dan. 2000. On building a more efficient grammar by exploiting types. *Nat. Lang. Eng.* 6:15–28.

Jakob, Max, Markéta Lopatková, and Valia Kordoni. 2010. Mapping between dependency structures and compositional semantic representations. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, pages 2491–2497.

Kruijff-Korbayová, Ivana, Klára Chvátalová, and Oana Postolache. 2006. Annotation guidelines for czech-english word alignment. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 1256–1261.

Macken, Lieve. 2010. Annotation guidelines for dutchenglish word alignment. *Technical Report* .

Melamed, Dan. 1998. Annotation style guide for the blinker project. *Technical Report* .

Nivre, Joakim, Johan Hall, and Jens Nilsson. 2006. Maltparser: a data-driven parser-generator for dependency parsing. In *Proceedings of LREC-2006*.

Oepen, Stephan. 2008. The transfer formalism. general purpose mrs rewriting. *Technical Report* .

Simov, Kiril, Petya Osenova, Alexander Simov, and Milen Kouylekov. 2004. Design and implementation of the bulgarian hpsg-based treebank. *Research on Language & Computation* 2(4):495–522.

Spreyer, Kathrin and Anette Frank. 2005. Projecting RMRS from TIGER Dependencies. In *Proceedings of the HPSG 2005 Conference*, pages 354–363. Lisbon, Portugal.

Tinsley, John, Mary Hearne, and Andy Way. 2009. Exploiting parallel treebanks to improve phrase-based statistical machine translation. In *Proceedings of the 10th International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '09, pages 318–331. Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-642-00381-3.