Semi-automatic Acquisition of Lexical Resources and Grammars for Event Extraction in Bulgarian and Czech

Hristo Tanev
Joint Research Centre
European Commission
via Fermi 2749, Ispra
Italy
hristo.tanev@jrc.ec.europa.eu

Josef Steinberger
University of West Bohemia
Faculty of Applied Sciences
Department of Computer Science and Engineering
NTIS Centre Univerzini 8, 30614 Plzen
Czech Republic
jstein@kiv.zcu.cz

Abstract

In this paper we present a semi-automatic approach for acqusition of lexico-syntactic knowledge for event extraction in two Slavic languages, namely Bulgarian and Czech. The method uses several weakly-supervised and unsupervised algorithms, based on distributional semantics. Moreover, an intervention from a language expert is envisaged on different steps in the learning procedure, which increases its accuracy, with respect to unsupervised methods for lexical and grammar learning.

1 Introduction

Automatic detection and extraction of events from online news provide means for tracking the developments in the World politics, economy and other important areas of life.

Event extraction is a branch of information extraction, whose goal is the automatic retrieval of structured information about events described in natural language texts. Events include interactions among different entities, to each of which an event-specific semantic role can be assigned. This role reflects the way in which the entity participates in the event and interacts with the other entities. For example, in the fragment "Three people were injured in a building collapse", the phrase "three people" may be assigned a semantic role $injured - victim$. The list of semantic roles depends on the adopted event model.

The event extraction technology may decrease the information overload, it allows automatic conversion of unstructured text data into structured one, it can be used to pinpoint interesting news articles, also extracted entities and their corresponding semantic roles can provide brief summaries of the articles.

Using lexico-syntactic knowledge is one of the promising directions in modeling the event-specific semantic roles (Hogenboom et al., 2011). While for English linear patterns seem to work quite well (Tanev et al., 2008), for other languages,where word ordering is more free, cascaded grammars proved to improve the results (Zavarella et al., 2008). In particular, Slavic languages are more free-order than English; consequently, using cascaded grammars may be considered a relevant approach.

In this paper we present an ongoing effort to build event extraction cascaded grammars for Bulgarian and Czech in the domain of violent news. To achieve this goal we put forward a semi-automatic approach for building of event extraction grammars, which uses several weakly-supervised algorithms for acquisition of lexical knowledge, based on distributional semantics and clustering. Moreover, the lexical knowledge is learned in the form of semantic classes, which then can be used as a basis for building of a domain-specific ontology.

To the best of our knowledge, there are no previous attempts to perform event extraction for Slavic languages, apart from the work presented in (Turchi et al., 2011).

The importance of Czech and Bulgarian languages comes from the geopolitical positions of the countries where they are spoken: Czech Republic is in a central geographical position between Eastern and Western Europe; Bulgaria is on the borders of the European Union, on a crossroad between Europe and Asia, surrounded by different cultures, languages and religions. These geopolitical factors contribute to the importance of the news from Czech Republic and Bulgaria and consequently make automatic event extraction from these news an useful technology for political analysts.

The paper has the following structure: In section 2 we make a short overview of the related ap-

Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing, pages 110–118,
Sofia, Bulgaria, 8-9 August 2013. ©2010 Association for Computational Linguistics

proaches; in section 3 we describe our method for lexical and grammar learning; section 4 presents our experiments and evaluation for Bulgarian and Czech languages and section 5 discusses the outcome of the experiments and some future directions.

2 Related Work

There are different approaches for event extraction. Most of the work up to now has aimed at English (see among the others (Naughton et al., 2006) and (Yangarber et al., 2000)), however (Turchi et al., 2011) presented automatic learning of event extraction patterns for Russian, English and Italian.

Our work is based on weakly supervised algorithms for learning of semantic classes and patterns, presented in (Tanev et al., 2009) and (Tanev and Zavarella, 2013); these approaches are based on distributional semantics. There are different other methods which use this paradigm: A concept and pattern learning Web agent, called NELL (Never Ending Language Learning) is presented in (Carlson et al., 2010). Parallel learning of semantic classes and patterns was presented in (Riloff and Jones, 1999). However these approaches do not try to derive grammars from the acquired resources, but stop at purely lexical level.

Relevant to our approach are the grammar learning approaches. A survey of supervised and unsupervised approaches is presented in (D'Ulizia et al., 2011). The supervised ones require annotation of big amounts of data which makes the development process long and laborious. On the other hand, unsupervised methods try to generalize all the training data by using different heuristics like the minimal description length. Since for event extraction only specific parts of the text are analyzed, in order to use unsupervised grammar acquisition methods for learning of event extraction grammars, one should collect the exact phrases which describe the events. In practice, this would transform the unsupervised methods into supervised ones. With respect to the state-of-the art grammar inference approaches, our method allows for more interaction between the grammar expert and the learning system. Moreover, our learning starts from lexical items and not from annotated texts, which decreases the development efforts.

3 Semi-automatic Learning of Lexica and Grammars

The event extraction grammar, exploited in our approach is a cascaded grammar which on the first levels detects references to entities, like people, groups of people, vehicles, etc. On the upper levels our cascaded grammar detects certain events in which these entities participate: In the domain of violent news, people may get killed, wounded, kidnapped, arrested, etc. If we consider as an example the following Bulgarian text: "Група протестиращи бяха арестувани вчера по време на демонстрации в центъра на столицата" ("A group of protesters were arrested yesterday during demonstrations in the centre of the capital"), our grammar will detect first that "Група протестиращи" ("A group of protesters") refers to a group of people and then, it will find that "Група протестиращи бяха арестувани" ("A group of protesters were arrested") refers to an arrest event where the aforementioned group of people is assigned the semantic role $arrested$.

In order to build such a grammar, we acquire semi-automatically the following resources:

1. a dictionary of words which refer to people and other entities in the required domain-specific context, e.g. "войник", "voják" ("soldier" in Bulgarian and Czech), "жена" , "žena" ("woman" in Bulgarian and Czech), etc.

2. a list of modifiers and other words which appear in phrases referring to those entities, e.g. "цивилен", "civilní" ("civil" in Bulgarian and Czech), "НАТО" ("NATO"), etc.

3. grammar rules for parsing entity-referring phrases. For example, a simple rule can be: $PERSON\_PHRASE \rightarrow PER$ *connector* ORG
where PER and ORG are words and multiwords, referring to people and organizations, *connector* → "от" for Bulgarian or *connector* → "" (empty string) for Czech. This rule can parse phrases like "войник от НАТО" or "voják NATO" ("NATO soldier")

4. a list of words which participate in event patterns like "арестуван", "zadržen" ("arrested" in Bulgarian and Czech) or "убит", "zabit" ("killed" in Bulgarian and Czech).

5. a set of grammar rules which parse event-description phrases. For example, a simple rule can be:
   $KILLING \rightarrow PER\ connector\ KILLED\_PARTICIPLE$
   where $connector$ → "беше" for Bulgarian or $connector$ → "byl" for Czech.
   This rule will recognize phrases like "Войник от НАТО беше убит" or "Voják NATO byl zabit" ("A NATO soldier was killed" in Bulgarian and Czech")

In order to acquire this type of domain lexica and a grammar, we make use of a semi-automatic method which acquires in parallel grammar rules and dictionaries. Our method exploits several state-of-the-art algorithms for expanding of semantic classes, distributional clustering, learning of patterns and learning of modifiers, described in (Tanev and Zavarella, 2013). The semantic class expansion algorithm was presented also in (Tanev et al., 2009). These algorithms are multilingial and all of them are based on distributional semantics. They use a non-annotated text corpus for training.

We integrated these algorithms in a semi-automatic schema for grammar learning, which is still in phase of development. Here is the basic schema of the approach:

1. The user provides a small list of seed words, which designate people or other domain-specific entities, e.g." soldiers","civilians", "fighters" (We will use only English-language examples for short, however the method is multilingual and consequently applicable for Czech and Bulgarian).

2. Using the multilingual semantic class expansion algorithm (Tanev et al., 2009) other words are learned (e.g. "policemen", "women", etc.), which are likely to belong to the same semantic class. First, the algorithm finds typical contextual patterns for the seed words from not annotated text. For example, all the words, referring to people tend to appear in linear patterns like *[PEOPLE] were killed*, *thousands of [PEOPLE]* , *[PEOPLE] are responsible*, etc. Then, other words which tend to participatre in the same contextual patterns are extracted from the unannotated text corpus. In such a way the algorithm learns additional words like "policemen", "killers", "terrorists", "women", "children", etc.

3. Since automatic approaches for learning of semantic classes always return some noise in the output, a manual cleaning by a domain expert takes place as a next step of our method.

4. Learning modifiers: At this step, for each semantic class learned at the previous step (e.g. *PEOPLE*, we run the modifier learning algorithm, put forward by (Tanev and Zavarella, 2013) , which learns domain-specific syntactic modifiers. Regarding the class *PEOPLE*), the modifiers will be words like " Russian", "American", "armed", "unarmed", "masked", etc. The modifier learning algorithm exploits the principle that the context distribution of words from a semantic class is most likely similar to the context distribution of these words with syntactic modifiers attached. The algorithm uses this heuristic and does not use any morphological information to ensure applications in multilingual settings.

5. Manual cleaning of the modifier list

6. Adding the following grammar rule at the first level of the cascaded grammar, which uses the semantic classes and modifiers, learned at the previous steps:
   $Entity(class : C) \rightarrow (LModif(class : C))*\ Word(class : C)\ (RModif(class : C))*$
   This rule parses phrases, like "masked gunmen from IRA", referring to an entity from a semantic class C, e.g. *PERSON*. It should consist of a sequence of 0 or more left modifiers for this class, e.g. "masked", a word from this class ("gunmen" in this example) and a sequence of 0 or more right modifiers ("from IRA" in the example").

7. Modifiers learned by the modifier learning algorithm do not cover all the variations in the structure of the entity-referring phrases, since sometimes the structure is more complex and cannot be encoded through a list of lexical patterns. Consider, for example, the following phrase "soldiers from the special forces of the Russian Navy". There is a little

chance that our modifier learning algorithm acquires the string “from the special forces of the Russian Navy”, on the other hand the following two grammar rules can do the parsing:
$RIGHT\_PEOPLE\_MODIFIER \rightarrow$
$``from'' MILITARY\_FORMATION$
$MILITARY\_FORMATION \rightarrow$
$LeftModMF * \ MFW \ RightModMF*$
where $MILITARY\_FORMATION$ is a phrase which refers to some organization (in the example, shown above, the phrase is “the special forces of the Russian Navy”), MFW is a term which refers to a military formation (“the special forces”) and $LeftModMF$ and $RightModMF$ are left and right modifiers of the military formation entity (for example, a right modifier is“of the Russian Navy”).

In order to learn such more complex structure, we propose the following procedure:

(a) The linguistic expert chooses semantic classes, for which more elaborated grammar rules should be developed. Let’s take for example the class *PEOPLE*.

(b) Using the context learning sub-algorithm of the semantic class expansion, used in step 2, we find contextual patterns which tend to co-occur with this class. Apart from the patterns shown in step 2, we also learn patterns like *[PEOPLE] from the special forces*, *[PEOPLE] from the Marines*, *[PEOPLE] from the Russian Federation*, *[PEOPLE] from the Czech Republic*, *[PEOPLE] with guns*, *[PEOPLE] with knives*, *[PEOPLE] with masks*, etc.

(c) We generalize contextual patterns, in order to create grammar rules. In the first step we create automatically syntactic clusters separately for left and right contextual patterns. Syntactic clustering puts in one cluster patterns where the slot and the content-bearing words are connected by the same sequence of stop words. In the example, shown above, we will have two syntactic clusters of patterns: The first consists of patterns which begin with *[PEOPLE] from the* and the second contains the patterns, which start with *[PEOPLE] with*. These clusters can be represented via grammar rules in the following way:
RIGHT_PEOPLE_MODIFIER →“from the” X
X → (special forces | Marines | Russian Federation | Czech Republic)
RIGHT_PEOPLE_MODIFIER → “with” Y
Y → (knives | guns | masks)

(d) Now, several operations can be done with the clusters of words inside the grammar rules:

- Words inside a cluster can be clustered further on the basis of their semantics. In our system we use bottom up agglomerative clustering, where each word is represented as a vector of its context features. Manual cleaning and merging of the clusters may be necessary after this automatic process. If words are not many, only manual clustering can also be an option. In the example above “special forces” and “Marines” may form one cluster, since both words designate the class *MILITARY_FORMATION* and the other two words designate countries and also form a separate semantic class.
- In the grammar introduce new non-terminal symbols, corresponding to the newly learnt semantic classes. Then, in the grammar rules substitute lists of words with references to these symbols. (Still we do modification of the grammar rules manually, however we envisage to automate this process in the future). For example, the rule

  X → (special forces | Marines | Russian Federation | Czech Republic)
  will be transformed into
  X → (MILITARY_FORMATION | COUNTRY)
  MILITARY_FORMATION → (special forces | Marines)
  COUNTRY → (Russian Federation

PEOPLE → (NUMBER от *(from))?* $PEOPLE_a$ Example: “двама от българските войници” *(“two of the Bulgarian soldiers”)*
$PEOPLE_a$ → *PEOPLE*$_b$ *((*от *(from)* | на *(of)* | в *(in)) (ORG | PLACE))** Example: “служители на МВР” *(“staff from the MVR (Ministry of the Internal Affairs)”)*
$PEOPLE_b$ → *LeftPM* PEOPLE_W RightPM** Example: “неизвестни нападатели с качулки” *(“unknown attackers with hoods”)*

Table 1: Rules for entity recognition for the Bulgarian language

| *Czech Republic)*

- Clusters can be expanded by using the semantic class expansion algorithm, introduced before, followed by manual cleaning. In our example, this will add other words for *MILITARY_FORMATION* and *COUNTRY*. Consequently, the range of the phrases, parsable by the grammar rules will be augmented.

(e) The linguistic expert may choose a subset of the semantic classes, obtained on the previous step, (e.g. the the semantic class *MILITARY_FORMATION*) to be modeled further via extending the grammar with rules about their left and right modifiers. Then, the semantic class is recursively passed to the input of this grammar learning procedure.

8. Learning event patterns: In this step we learn patterns like *[PEOPLE]* “бяха арестувани” or *[PEOPLE] “byl zadržen”* (*[PEOPLE] were/was arrested* in Bulgarian and Czech). The pattern learning algorithm collects context patterns for one of the considered entity categories (e.g. *[PEOPLE]*. This is done through the context learning sub-algorithm described in step 2. Then, it searches for such context patterns, which contain words, having distributional similarity to words, describing the target event (e.g. “арестувани”, “zadržen” (“arrested”)).

   For example, if we want to learn patterns for *arrest* events in Bulgarian, the algorithm first learns contexts of *[PEOPLE]*. These contexts are *[PEOPLE]* бяха убити (*[PEOPLE] were killed*), хиляди [PEOPLE] (*thousands of [PEOPLE]*), *[PEOPLE]* бяха заловени (*[PEOPLE] were captured*), etc. Then, we pass to the semantic expansion algorithm (see step 2) seed words which express the event arrest, namely “задържани”, “арестувани” (“apprehended”, “arrested”), etc. Then, it will discover other similar words like “заловени” (“captured”). Finally, the algorithm searches such contextual patterns, which contain any of the seed and learnt words. For example, the pattern [PEOPLE] бяха заловени (*[PEOPLE] were captured*) is one of the newly learnt patterns for *arrest* events.

9. Generalizing the patterns: In this step we apply a generalization algorithm, described in step 7 to learn grammar rules which parse events. For example, two of the learned rules for parsing of arrest events in Bulgarian are:

   ARREST → PEOPLE “бяха” *(“were”)*
   ARREST_PARTICIPLE
   ARREST_PARTICIPLE → (“арестувани” (“arrested”) | “заловени”(“captured”) | “закопчани” (“handcuffed”) *)*

The outcome of this learning schema is a grammar and dictionaries which recognize descriptions of different types of domain-specific entities and events, which happened with these entities. Moreover, the dictionaries describe semantic classes from the target domain and can be used further for creation of a domain ontology.

4 Experiments and Evaluation

In our experiments, we applied the procedure shown above to learn grammars and dictionaries for parsing of phrases, referring to people, groups of people and violent events in Bulgarian and Czech news. We used for training 1 million news titles for Bulgarian and Czech, downloaded from

KILLING → KILL_VERB (a (and) | i (and) | jeden (one) | jeden z (one of))? [PEOPLE]
KILL_VERB → (zabit (killed) | zabila | zahynul (died) | zabiti | ubodal (stabbed) | ubodala | ...)
KILLING → KILL_ADJ [PEOPLE]
KILL_ADJ → (mrtvou (dead) | mrtvého (dead) | ...)
KILLING → [PEOPLE] KILL_VERB$_a$
KILL_VERB$_a$ *→ (zahynul (died) | zamřel (died) | ...)*
KILLING → [PEOPLE] byl (was) KILL_VERB$_b$
KILL_VERB$_b$ *→ (zabit (killed) | ...)*

Table 2: Rules for parsing of killing events and their victims in Czech

the Web and a small number of seed terms, referring to people and actions. We had more available time to work for the Bulgarian language, that is why we learned more complex grammar for Bulgarian. Both for Czech and Bulgarian, we learned grammar rules parsing event description phrases with one participating entity, which is a person or a group of people. This is simplification, since often an event contains more than one participant, in such cases our grammar can detect the separate phrases with their corresponding participants, but currently it is out of the scope of the grammar to connect these entities. The event detection rules in our grammar are divided into semantic classes, where each class of rules detects specific type of events like *arrest, killing, wounding*, etc. and also assigns an event specific semantic role to the participating entity, e.g. *victim, perpetrator, arrested, kidnapped*.

In order to implement our grammars, we used the EXPRESS grammar engine (Piskorski, 2007). It is a tool for building of cascaded grammars where specific parts of the parsed phrase are assigned semantic roles. We used this last feature of EXPRESS to assign semantic roles of the participating person entities.

For Czech we learned a grammar which detects killings and their victims. For Bulgarian, we learned a grammar, which parses phrases referring to killings, woundings and their victims, arrests and who is arrested, kidnappings and other violent events with their perpetrators and targeted people.

4.1 Learning people-recognition rules

For Czech our entity extraction grammar was relatively simple, since we learned just a dictionary of left modifiers. Therefore, we skipped step 7 in the learning schema, via which more elaborated entity recognition grammars are learned. Thus, the Czech grammar for recognizing phrases, referring to people contains the following rules:
PEOPLE → LeftMod PEOPLE_TERM*
LeftMod → (“mladou” (“young”) | “neznámému”(“unknown”) | “starší” (“old”) | ...)
PEOPLE_TERM → (“vojáci” (“soldiers”) | “civilisté”(“civilians”) | “ženu” (“woman”) | ...)

This grammar recognizes phrases like “mladou ženu” (“young woman” in Czech). Two dictionaries were acquired in the learning process: A dictionary of nouns, referring to people and left modifiers of people. The dictionary of people-referring nouns contains 268 entries, obtained as a result of the semantic class expansion algorithm. We used as a seed set 17 words like “muži” (“men”), “voiáci” (“soldiers”), etc. The algorithm learned 1009 new words and bigrams, 251 of which were correct (25%), that is refer to people. One problem here was that not all morphological forms were learned by our class expansion algorithm. In a language with rich noun morphology, as Czech is, this influenced on the coverage of our dictionaries.

After manual cleaning of the output from the modifier learning algorithm, we obtained 603 terms; the learning accuracy of the algorithm was found to be **55%** .

For Bulgarian we learned a more elaborated people recognition grammar, which is able to parse more complex phrases like “един от маскираните нападатели” (“one of the masked attackers”) and “бойци от българския контингент в Ирак” (“soldiers from the Bulgarian contingent in Iraq”). The most important rules which we learned are shown in Table 1. In these rules *PEOPLE_W* encodes a noun or a bigram which refers to people, ORG is an organization; we learned mostly organizations, related to the domain of security, such as different types of military and other armed formations like “силите на реда” (“security forces”), also governmental organizations, etc. *PLACE* stands for names of places and common nouns, referring to places such as “столицата” (“the capital”). We also learned modifiers for these categories and added them to the grammar. (For simplicity, we do not show the grammar rules for parsing ORG abd $PLACE$; we will just mention that both types of phrases are allowed to have a sequence of left modifiers, one or more nouns from the corresponding class and a sequence of 0 or more right modifiers.) Both categories $PLACE$ and ORG were obtained in step 7 of the learning schema, when exploring the clusters of words which appear as modifiers after the nouns, referring to people, like in the following example “бойци от българския контингент” (“soldiers from the Bulgarian contingent”); then, we applied manual unification of the clusters and their subsequent expansion, using the semantic class expansion algorithm.

Regarding the semantic class expansion, with 20 seed terms we acquired around 2100 terms, from which we manually filtered the wrong ones and we left 1200 correct terms, referring to people; the accuracy of the algorithm was found to be **57%** in this case.

We learned 1723 nouns for organizations and 523 place names and common nouns. We did not track the accuracy of the learning for these two classes. We also learned 319 relevant modifiers for people-referring phrases; the accuracy of the modifier learning algorithm was found to be **67%** for this task.

4.2 Learning of event detection rules

This learning takes place in step 8 and 9 of our learning schema. As it was explained, first linear patterns like *[PEOPLE] “byl zadržen”* (*[PEOPLE] was arrested*) are learned, then through a semi-automatic generalization process these patterns are transformed into rules like:
ARREST → PEOPLE “byl” ARREST_VERB

In our experiments for Czech we learned grammar rules and a dictionary which recognize different syntactic constructions, expressing killing events and the victims. These rules encode 156 event patterns. The most important of these rules are shown in Table 2. Part of the event rule learning process is expansion of a seed set of verbs, and other words, referring to the considered event (in this case *killing*).For this task the semantic class expansion algorithm showed significantly lower accuracy with respect to expanding sets of nouns - only **5%**. Nevertheless, the algorithm learned 54 Czech words, expressing killing and death.

For Bulgarian we learned rules for detection of killing and its victims, but also rules for parsing of wounding events, arrests, targeting of people in violent events, kidnapping, and perpetrators of violent events. These rules encode 605 event patterns. Some of the rules are shown in Table 3.

4.3 Evaluation of event extraction

In order to evaluate the performance of our grammars, we created two types of corpora: For the precision evaluation we created bigger corpus of randomly picked excerpts of news from Bulgarian and Czech online news sources. More precisely, we used 7’550 news titles for Czech and 12’850 news titles in Bulgarian. We also carried out a preliminary recall evaluation on a very small text collection: We manually chose sentences which report about violent events of the types which our grammars are able to capture. We selected 17 sentences for Czech and 28 for Bulgarian. We parsed the corpora with our EXPRESS grammars and evaluated the correctness of the extracted events. Since each event rule has assigned an event type and a semantic role for the participating people reference, we considered a correct match only when both a correct event type and a correct semantic role are assigned to the matched text fragment. Table 4 shows the results from our evaluation. The low recall in Czech was mostly due to the insufficient lexicon for people and the too simplistic grammar.

Language	Precision	Recall
Bulgarian	93%	39%
Czech	88%	6%

Table 4: Event extraction accuracy

5 Discussion

In this paper we presented a semi-automatic approach for learning of grammar and lexical knowledge from unannotated text corpora. The method is multilingual and relies on distributional approaches for semantic clustering and class expansion.

```
KILLING → KILL_VERB (бяха (were) | са (are)) [PEOPLE]
KILL_VERB → (загинали (killed) | убити (killed) | застреляните (shot to death) | ...)
KILLING → KILL_PHRASE на (of) [PEOPLE]
KILL_PHRASE → (отне живота (took the life) | причини смъртта (caused the death) | ...)
WOUNDING → WOUND_VERB (бяха (were) | са (are)) [PEOPLE]
WOUND_VERB → (ранени (wounded) | пострадалите (injured) | ...)
ARREST → [PEOPLE] ARREST_VERB
ARREST_VERB → (арестувани (arrested) | задържани (detained) | ...)
```

Table 3: Some event parsing rules for Bulgarian

We are currently developing event extraction grammars for Czech and Bulgarian. Preliminary evaluation shows promising results for the precision, while the recall is still quite low. One of the factors which influences the law recall was the insufficient number of different morphological word variations in the learned dictionaries. The morphological richness of Slavic languages can be considered by adding morphological dictionaries to the system or creating an automatic procedure which detects the most common endings of the nouns and other words and expands the dictionaries with morphological forms.

Another problem in the processing of the Slavic languages is their relatively free order. To cope with that, often the grammar engineer should introduce additional variants of already learned grammar rules. This can be done semi-automatically, where the system may suggest additional rules to the grammar developer. This can be done through development of grammar meta-rules.

With respect to other approaches, grammars provide transparent, easy to expand model of the domain. The automatically learned grammars can be corrected and extended manually with hand-crafted rules and linguistic resources, such as morphological dictionaries. Moreover, one can try to introduce grammar rules from already existing grammars. This, of course, is not trivial because of the different formalisms exploited by each grammar. It is noteworthy that the extracted semantic classes can be used to create an ontology of the domain. In this clue, parallel learning of a domain-specific grammars and ontologies could be an interesting direction for future research.

The manual efforts in the development of the grammars and the lexical resources were mainly cleaning of already generated lists of words and manual selection and unification of word clusters. Although we did not evaluate precisely the invested manual efforts, one can estimate them by the size of the automatically acquired word lists and their accuracy, given in section *Semi-automatic Learning of Lexica and Grammars*.

We plan to expand the Czech grammar with rules for more event types. Also, we think to extend both the Bulgarian and the Czech event extraction grammars and the lexical resources, so that it will be possible to detect also disasters, humanitarian crises and their consequences. This will increase the applicability and usefulness of our event extraction grammars.

Acknowledgments

This work was partially supported by project “NTIS - New Technologies for Information Society”, European Center of Excellence, CZ.1.05/1.1.00/02.0090.

References

A. Carlson, J. Betteridge, B. Kisiel, B. Settles, R. Estevam, J. Hruschka, and T. Mitchell. 2010. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10).*

A. D'Ulizia, F. Ferri, and P. Grifoni. 2011. A survey of grammatical inference methods for natural language learning. *Artificial Intelligence Review vol. 36 issue 1.*

F. Hogenboom, F. Frasincar, U. Kaymak, and F. Jong. 2011. An overview of event extraction from text. In *Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011) at ISWC 2011.*

M. Naughton, N. Kushmerick, and J. Carthy. 2006. Event Extraction from Heterogeneous News Sources. In *Proceedings of the AAAI 2006 workshop on Event Extraction and Synthesis*, Menlo Park, California, USA.

J. Piskorski. 2007. ExPRESS – Extraction Pattern Recognition Engine and Specification Suite. In *Proceedings of FSMNLP 2007.*

E. Riloff and R. Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI 99).*

H. Tanev and V. Zavarella. 2013. Multilingual learning and population of event ontologies. a case study for social media. In P. Buitelaar and P. Cimiano, editors, *Towards Multilingual Semantic Web (in press).* Springer, Berlin & New York.

H. Tanev, J. Piskorski, and M. Atkinson. 2008. Real-Time News Event Extraction for Global Crisis Monitoring. In *Proceedings of NLDB 2008.*, pages 207–218.

H. Tanev, V. Zavarella, J. Linge, M. Kabadjov, J. Piskorski, M. Atkinson, and R. Steinberger. 2009. Exploiting Machine Learning Techniques to Build an Event Extraction System for Portuguese and Spanish. *Linguamática: Revista para o Processamento Automático das Línguas Ibéricas*, 2:550–566.

M. Turchi, V. Zavarella, and H. Tanev. 2011. Pattern learning for event extraction using monolingual statistical machine translation. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2011), Hissar, Bulgaria.*

R. Yangarber, R. Grishman, P. Tapanainen, and S. Huttunen. 2000. Unsupervised Discovery of Scenario-Level Patterns for Information Extraction. In *Proceedings of ANLP-NAACL 2000, Seattle, USA, 2000.*

V. Zavarella, H. Tanev, and J. Piskorski. 2008. Event Extraction for Italian using a Cascade of Finite-State Grammars. In *Proceedings of FSMNLP 2008.*