Named Entities in Czech: Annotating Data and Developing NE Tagger*

Magda Ševčíková, Zdeněk Žabokrtský, and Oldřich Krůza

Faculty of Mathematics and Physics, Charles University
Malostranské nám. 25, CZ-11800 Prague, Czech Republic
{sevcikova,zabokrtsky,kruza}@ufal.mff.cuni.cz

Abstract. This paper deals with the treatment of Named Entities (NEs) in Czech. We introduce a two-level NE classification. We have used this classification for manual annotation of two thousand sentences, gaining more than 11,000 NE instances. Employing the annotated data and Machine-Learning techniques (namely the top-down induction of decision trees), we have developed and evaluated a software system aimed at automatic detection and classification of NEs in Czech texts.

1 Introduction

After the series of Message Understanding Conferences (MUC; [1]), processing of NE became a well established discipline within the NLP domain (see [2] for a survey of NE related research), usually motivated by the needs of Information Extraction, Question Answering, or Machine Translation. For English, one can find literature about attempts at rule-based solutions for the NE task as well as machine-learning approaches, be they dependent on the existence of labeled data (such as CoNLL-2003 shared task data), unsupervised (using redundancy in NE expressions and their contexts, see e.g. [3]) or a combination of both (such as [4], in which labeled data are used as a source of seed for an unsupervised procedure exploiting huge unlabeled data).

For Czech, the situation is different. To our best knowledge, until the presented work there have been no data with explicitly annotated NE instances available for Czech. Although there are several other types of available resources potentially usable for recognition and classification of NE (e.g. gazetteers, or technical lemma suffixes used at the morphological layer of PDT [5]), we have not found any published attempt concerning development of NE taggers for Czech.[1]

This paper is structured as follows: in Section 2 we introduce our classification of NE, which we have used for annotating sample sentences as described in

* The research reported on in this paper was supported by the projects 1ET101120503, MSM0021620838, MSMT CR LC536, GD201/05/H014, and GA UK 643/2007.

[1] Even if some approaches developed for English are claimed to be language independent, it is obvious that they cannot be straightforwardly applied to Czech because of its rich inflection.

Section 3. Section 4 presents our NE tagger trained on the annotated data. The summary is given in Section 5.

2 Proposed Two-level NE Classification

There is no generally accepted typology of Named Entities. One can see two trends: from the viewpoint of unsupervised learning, it is advantageous to have just a few coarse-grained categories (cf. the NE classification developed for MUC conferences or the classification proposed in [3], where only persons, locations, and organizations were distinguished), whereas those interested in semantically oriented applications prefer more informative (finer-grained) categories (e.g. [6] with eight types of person labels, or Sekine's Extended NE Hierarchy, cf. [7]).

Therefore we have proposed a two-level NE classification, as depicted in Figure 1. The first level corresponds to rough categories (called *NE supertypes*) such as person names, geographical names etc., whereas the second level provides a more detailed classification: e.g. within the supertype of geographical names, the *NE types* of names of cities/towns, names of states, names of rivers/seas/lakes etc. are distinguished. If more robust processing is necessary, only the first level (NE supertypes) can be used, whereas the second level (NE types) comes into play when more subtle information is needed. Each NE type is encoded by a unique two-character tag (e.g., `gu` for names of cities/towns, `gc` for names of states; a special tag, such as `g_`, makes it possible to leave the NE type underspecified).

Besides the terms of NE type and supertype, we use also the term *NE instance*, which stands for a continuous subsequence of tokens expressing the entity in a given text. In the simple plain-text format, which we use for manual annotations, the NE instances are marked as follows: the word or the span of words belonging to the NE is delimited by symbols `<` and `>`, with the former one immediately followed by the NE type tag (e.g. `<pf` *John*`>` *loves* `<pf` *Mary*`>`).

The annotation scheme allows for the embedding of NE instances. There are two types of embedding. In the first case, the NE of a certain type can be embedded in another NE (e.g., the river name can be part of a name of a city as in `<gu` *Ústí nad* `<gh` *Labem*`>>`). In the second case, two or more NEs are parts of a (so-called) *container NE* (e.g., two NEs, a first name and a surname, form together a person name container NE such as in `<P<pf` *Paul*`>` `<ps` *Newman*`>>`). The container NEs are marked with a capital one-letter tag: `P` for (complex) person names, `T` for temporal expressions, `A` for addresses, and `C` for bibliographic items. A more detailed description of the NE classification can be found in [8].

3 Annotating Data

We have created the data with labeled NE instances by the following procedure:

1. We have randomly selected 2000 sentences from the Czech National Corpus[2] from the result of the query `([word=".*[a-z0-9]"] [word="[A-Z].*"])`

[2] `http://ucnk.ff.cuni.cz`

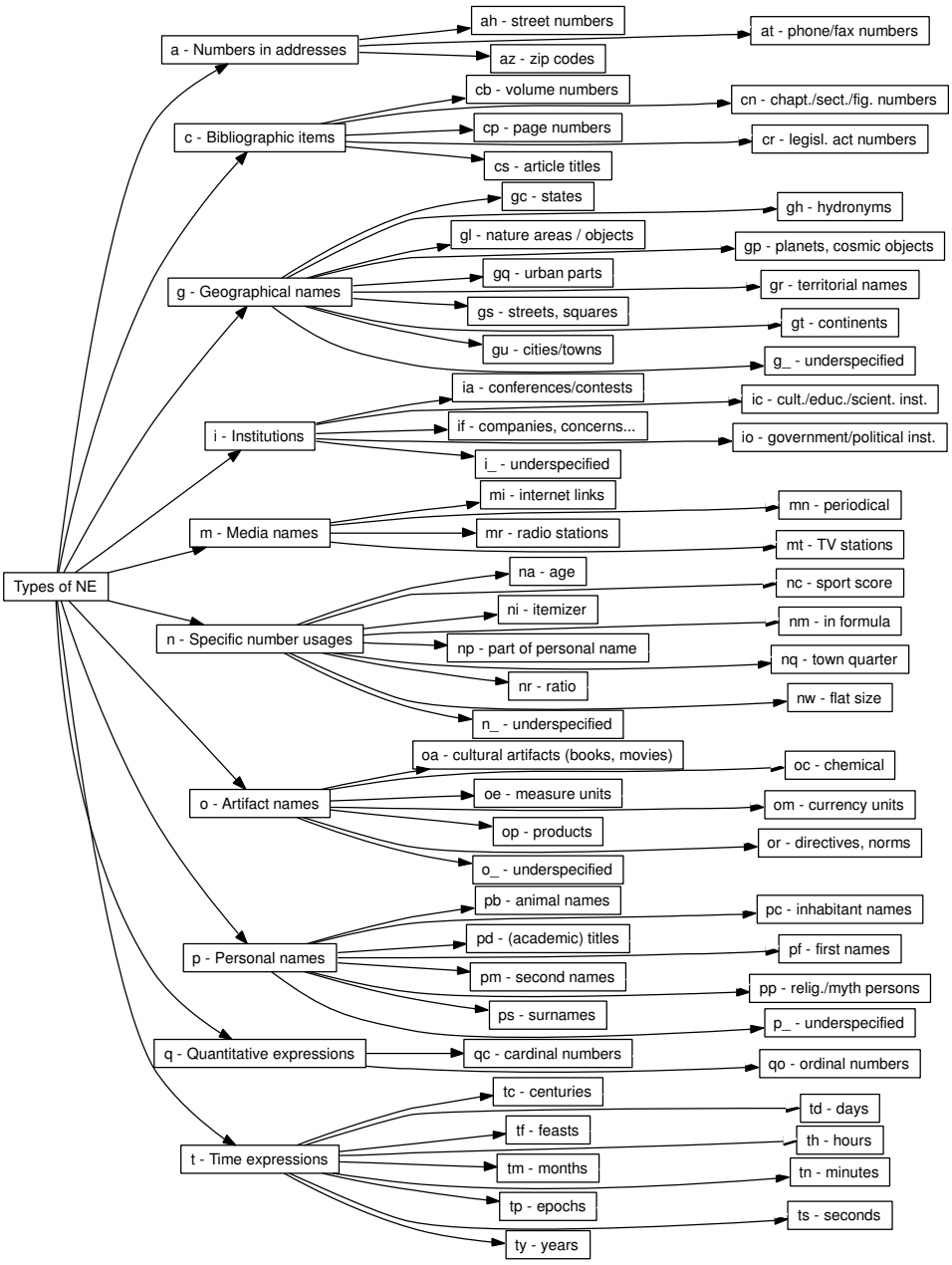

Fig. 1. Proposed two-level classification of NEs in Czech. Note that the (detailed) NE types are divided into two columns just because of the space reasons here.

(this query makes the relative frequency of NEs in the selection higher than the corpus average, which makes the subsequent manual annotation much more effective, even if it may slightly bias the distribution of NE types).
2. The data (simple line-oriented plain-text files, editable in any text editor) have been manually annotated (i.e., enriched with starting and ending symbols and tags for NE types or NE containers) by two annotators in parallel. Differently annotated instances have been checked and decided by a third person. 11,644 NE instances have been detected in the sample.
3. The sentences have been enriched with morphological tags and lemmas using Jan Hajič's tagger shipped with Prague Dependency Treebank 2.0 ([5]).
4. The data have been divided into training, development test and evaluation test parts (8:1:1), and converted into pairs of XML files (source sentences enriched with morphological attributes are stored in an XML file with the same format as m-files in PDT 2.0, whereas the NE instances are represented in a separated XML file in the form of triples (i) reference to the first token of the given NE instance, (ii) reference to the last token of the instance, (iii) type of the NE instance according to the two-level classification; see [8] for more detail).

After the last step, the data have been 'frozen' and prepared for experiments with NE taggers described in Section 4 below.[3]

Sample of annotated text (after the second step):

```
Britský mediální umělec <p_ Sting> , vlastním jménem <P<pf Gordon>
<ps Sumner>> , který má vystoupit <T<td 14 .> <tm června>> v pražské
<ic Sportovní hale> , bude s největší pravděpodobností bydlet se svým
devětadvacetičlenným týmem pod krycím jménem v některém z pražských
hotelů .
<P<pd Ing .> <pf Karel> <ps Hennhofer> , <pd PhD>> , zastupující
ostravskou divizi <ic Technické inspekce " DOM-ZO <n_ 13> "> popsal
nové pojetí systému managementu jakosti podle normy <or ČSN EN ISO
<nr 9001:2001>> .
```

4 Development of NE tagger

4.1 Task Definition

Our goal was to create a program for automatic processing of NEs in Czech texts enriched with morphological annotation. It has been subdivided into two 'subgoals': (i) the word or the span of words belonging to the NE should be delimited, (ii) a type/container tag chosen from a given set (cf. Section 2) should be assigned to the detected NE instance.

A NE instance is correctly recognized if and only if both the span was correctly detected and the correct tag was assigned. However, we also present results based on less strict rules (only the supertype is taken into consideration) to provide the reader with more insight into the principles of the NE tagger.

[3] The data are available also for other researchers by request from the authors.

4.2 Decomposition of the Task

The two proposed subgoals are inherently different: the first one (i.e., detecting the span of a NE instance) involves searching the input text for tokens or sequences of tokens forming NE instances (such a sequence can be of an arbitrary length, limited only by the length of the sentence) whereas the second one (i.e., assigning the tag) is the assignment of a class to the detected NE instance.

Concerning the first subgoal, a considerable simplification can be reached by limiting the length of a NE instance we attempt to recognize. With such an approach, we can tackle the NE instance detection as a classification task as well. The price is that we will not be able to detect NE instances longer than the given upper bound any more. For our system, a bound of two words has been set.[4]

These two subgoals have been solved separately for single-word NE instances (see subtask 1) and for two-word NE instances (subtask 2). As for multi-word NE instances, only one type of such NE instances has been detected, namely multi-word names of Czech towns (subtask 3; i.e., only one tag can be assigned). Thus, there are three subtasks to be solved:

1. Detection and classification of single-word NE instances.
2. Detection and classification of two-word NE instances.
3. Detection and classification of one type of multi-word NE instances.

The first two subtasks have been approached as a feature-based classification. The third one has been solved by specially crafted algorithms. The feature sets are different for each subtask.

4.3 Implementation

The system has been implemented in Perl, it uses the *Rulequest c5.0* classifier. Firstly, the training of the system will be described, then the analysis will be briefly sketched.

Training is divided into (i) the preparation of the data for the classifier and (ii) the use of the classifier to construct a prediction method. The data sets for each subtask are prepared in the same way, except for the different feature sets.

We distinguish categorial and boolean features. Features used for detection – and subsequently also for classification – of single-word NE instances are the following:

- Categorial: How many times the lemma occurs in training data.
- Boolean: The word form is capitalized.
- Categorial: The NE type denoted by the technical lemma suffix as used at the morphological layer of PDT 2.0.

[4] NE instances of length up to two words cover more than 87 % of all NE instances in the training data.

- Boolean: The token is the only capitalized word (first word excluded) or the only number in the sentence.
- Boolean: The form is a single capital letter.
- Boolean: The lemma denotes a month.
- Boolean: The form is a number adjacent to another number.
- Boolean: The form matches a simple time expression pattern (e.g., *16:30*).
- Categorial: The lemma; `OTHER` value for non-frequent lemmas.
- Boolean: The lemma is included in the list of Czech town names.
- Tag-based features: In the positions of the morphological tag used in PDT 2.0, the part-of-speech information, information concerning the gender, number etc. is encoded. Values on these tag positions are treated as categorial features.
- Contextual features: Features representing presence or absence of trigger words in the immediate neighborhood; the list of around 600 trigger words (words that signal the beginning or the end of a NE instance, such as *president*) was semimanually extracted from the training data.

Features used for detection of two-word NE instances as well as for their classification are the following:

- Categorial: Part-of-speech pattern:
  Two-letter symbol formed by concatenation of the parts of speech of the two words in the scope (again, `OTHER` is used for infrequent combinations).
- Categorial: Single-word prediction pattern:
  The pair of NE types predicted by the system for the individual words of the bigram. In case of rare pairs, `OTHER` is used instead.
- Categorial: Capitalization pattern:
  Each word of the bigram is classified as (i) being the first word in the sentence, (ii) being capitalized or (iii) neither of the above. The couple of these categories is then used as the feature value.
- Boolean: The words agree in number, gender and case.
- Categorial: How many times the lemmas occur in the training data next to each other.

The **analysis** (i.e., detection and classification of NE instances in the unseen data) is performed for each sentence in three phases, as mentioned in Subsection 4.2 above.

For each word, features for single-word NE instance detection are evaluated. Based on these features, the prediction method decided whether the word is a NE instance. If so, then features used for classification of single-word NE instances are evaluated. The prediction method assigns a tag to the NE instance. For each bigram, the same procedure is executed as for each word, using feature sets for two-word NE instances. The last phase involves searching the sentence for an occurrence of a multi-word Czech town name (as the only one multi-word NE instance to be detected). In case such NE instance is found, the words are marked as a NE instance of type `gu`.

	Correct type	Correct supertype	Correct span
Precision	0.16	0.29	0.68
Recall	0.16	0.29	0.68
F-measure	0.16	0.29	0.68

Table 1. Results of the capitalization-based baseline classifier.

	Correct type	Correct supertype	Correct span
Precision	0.54	0.57	0.59
Recall	0.33	0.34	0.36
F-measure	0.40	0.43	0.45

Table 2. Results of the baseline classifier based on the in-data occurrence (the precision and recall results seem to be identical only because of rounding).

	All NE inst.	One-word NE inst.	Multi-word NE inst.
Correct type	0.74 / 0.54 / 0.62	0.72 / 0.69 / 0.70	0.93 / 0.22 / 0.35
Correct supertype	0.81 / 0.59 / 0.68	0.79 / 0.76 / 0.78	0.95 / 0.22 / 0.36
Correct span	0.88 / 0.64 / 0.75	0.87 / 0.84 / 0.86	0.98 / 0.23 / 0.37

Table 3. Final results of the developed system (precision/recall/F-measure).

4.4 Results and Evaluation

We evaluate the results using *precision*, *recall* and *f-measure*. We present results based on three definitions of a correctly detected (and classified) NE instance:

- The span of the NE instance was detected correctly.
- The span of the NE instance was detected correctly and a correct supertype tag (i.e., the first character of the NE type tag) was assigned.
- Both the span of the NE instance was detected correctly and a correct NE type tag was assigned.

Two baselines have been suggested. The first one recognized every capitalized word (excluding sentence-first words) as a NE instance of the most frequent type (`ps`, i.e. a surname; see Table 1). The second baseline (see Table 2) checked each word and bigram for presence in the training data (effectively evaluating the corresponding feature described above) and marked it as a NE instance if an occurrence has been found. The type of the NE instance was denoted as the type of one of the NE instances found in training data.

The final results are shown in Table 3. The F-measure equal to 0.62 seems to be a rather low number, but it is necessary to take into account the very high number of employed NE types (and thus very low baselines). Restricting the task in any dimension (i.e. the kind of entities sought, the number of types, etc.) improves the performance considerably. The weakest part is the recall in

recognizing multi-word NE instances, which is no surprise as such entities are frequent in real-world data and we have no satisfactory method to deal with them yet. The results are comparable with those of HAREM competitors ([9]) and of Sassano and Utsuro ([10]).

5 Conclusion

We believe that the contributions of our work are the following: (i) we have introduced a detailed two-level NE classification verified on authentic corpus data, (ii) we have manually annotated a substantive sample of Czech sentences with the proposed NE tags, and (iii) we have developed and evaluated a NE tagger for Czech. To our knowledge, the presented work is novel for Czech in all three aspects.

As for future work, we plan to use also unlabeled data for the development of NE taggers, and to study the status of NE instances at more abstract layers of linguistic representation, especially at the tectogrammatical layer as implemented in PDT 2.0.

References

1. Grishman, R., Sundheim, B.: Message Understanding Conference - 6: A Brief History. In: Proceedings of the 16th International Conference on Computational Linguistics (COLING). Volume I. (1996) 466–471
2. Sekine, S.: Named Entity: History and Future. http://www.cs.nyu.edu/~sekine/papers/NEsurvey200402.pdf (2004)
3. Collins, M., Singer, Y.: Unsupervised Models for Named Entity Classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC). (1999) 189–196
4. Talukdar, P.P., Brants, T., Liberman, M., Pereira, F.: A Context Pattern Induction Method for Named Entity Extraction. In: Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL-X). (2006) 141–148
5. Hajič, J., Panevová, J., Hajičová, E., Sgall, P., Pajas, P., Štěpánek, J., Havelka, J., Mikulová, M., Žabokrtský, Z., Ševčíková, M.: Prague Dependency Treebank 2.0 (2006)
6. Fleischman, M., Hovy, E.: Fine Grained Classification of Named Entities . In: Proceedings of the 19th International Conference on Computational Linguistics (COLING). Volume I. (2002) 267–273
7. Sekine, S.: Sekine's Extended Named Entity Hierarchy. http://nlp.cs.nyu.edu/ene/ (2003)
8. Ševčíková, M., Žabokrtský, Z., Krůza, O.: Zpracování pojmenovaných entit v českých textech. Technical report, ÚFAL MFF UK, Praha (2007)
9. Santos, D., Seco, N., Cardoso, N., Vilela, R.: HAREM: An Advanced NER Evaluation Contest for Portuguese. In: Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC). (2006) 1986–1991
10. Sassano, M., Utsuro, T.: Named Entity Chunking Techniques in Supervised Learning for Japanese Named Entity Recognition. In: Proceedings of the 18th International Conference on Computational Linguistics (COLING). Volume II. (2000) 705–711