Keywords: named entity recognition, Latvian and Lithuanian languages, bootstrapping

1. Introduction

Named entity recognition (NER) has been actively researched for over 20 years. Most of the research has, however, been focussed on resource rich languages, for instance, English French and Spanish. The scope of this paper covers the task of named entity recognition for two under-resourced languages – Latvian and Lithuanian. The author presents an open source freely available toolkit named TildeNER that makes use of existing supervised learning methodology (for instance, the Stanford NER conditional random field classifier (Finkel et al., 2005)) enriched with heuristic refinement methods in order to bootstrap NER models using unlabelled data, thus, creating a “highly supervised” semi-supervised named entity recognizer.

Latvian and Lithuanian are the state languages of two European Union member countries - Latvia and Lithuania. Both languages feature rich morphology with high morphological ambiguity and a relatively free order of constituents in sentences, thus, making the task of named entity recognition more difficult than, for instance, for English.

The current dominant approach to developing named entity recognition systems is supervised learning (Nadeau and Sekine, 2007). This, however, means that a prerequisite for NER model training is a large named entity (NE) annotated data corpus. For resource rich languages this is not an issue, but for under-resourced languages (for instance, the Baltic languages) is. For Latvian and Lithuanian there has been very little previous research in the field of named entity recognition. Most of the existing research has dealt with only toponym recognition, for instance, Skadiņa (2009) describes toponym recognition from image annotations using lexicons and patterns. Also the lack of annotated named entity corpora for both languages does not allow (without significant financial input for corpora creation) the development of a truly supervised NER system. Because of the available resource constraints, for Latvian and Lithuanian a semi-supervised NER system development approach was selected, more precisely, bootstrapping. The systems presented in the paper are, therefore, the first multi-class NER tools created for Latvian and Lithuanian. The main reason for the development of the Latvian and Lithuanian NER systems has been to tag NEs in comparable corpora for further bilingual NE alignment using NE mapping methods in the ACCURAT project. It is also planned to use the NER systems as a pre-processing step in machine translation in order to create NE-aware translations.

The next chapter gives a description of the NE-annotated corpora followed by a section on the design and methods applied in TildeNER and evaluation in section four. The paper is finalized with conclusions and a discussion of future work.

2. Annotated Corpora

For the task of named entity recognition relatively small NE annotated corpora was created. The corpora for both languages consists of IT localization (software reviews, manuals and other IT related articles), news (current news from news web portals) and Wikipedia articles in equal proportions. The first two parts were acquired using comparable corpora web crawling tools developed within the ACCURAT project. The corpora statistics is shown in Table 1. For the annotation task, NE mark-up guidelines were prepared. The guidelines are mostly compliant with the MUC-7 (Chinchor, 1998) NE annotation guidelines (adaptation to Latvian and Lithuanian was performed as

---

1 Report on information extraction from comparable corpora, Tools for building comparable corpus from the Web, public deliverable of the project ACCURAT, 2011. Published as part of TildeNER in the „Toolkit for multi-level alignment and information extraction from comparable corpora“, public deliverable of the project ACCURAT, 2011.
well as minor contradictions were resolved). The following NE categories were annotated: organization, person name, location, product, date, time, money.

| Document count | Latvian | Lithuanian |
|----------------|---------|------------|
| Seed           | 40      | 37         |
| Development    | 25      | 33         |
| Test           | 66      | 55         |
| Total          | 131     | 125        |

| Word count      | Latvian | Lithuanian |
|-----------------|---------|------------|
| Seed            | 20,959  | 18,852     |
| Development     | 10,053  | 17,827     |
| Test            | 41,208  | 36,239     |
| Total           | 72,220  | 72,918     |

Table 1: Latvian and Lithuanian corpora statistics.

The corpora were annotated by two annotators and disagreements were resolved by a third annotator for both languages. The inter-annotator agreement between the first two annotators using the Cohen’s Kappa statistic (Cohen, 1968) is 0.885 for Latvian and 0.822 for Lithuanian. This score, however, represents the overall complexity of the corpora including non-entities strictly classified as non-entities by both annotators. This score does not represent the actual NE annotation complexity and difficulties in NE border detection; that is, adding or removing non-entity data (tokens/sentences) will result in respectively higher or lower inter-annotator agreement. Therefore, separate NE category and NE border detection inter-annotator agreement scores are given in Table 2. The token level agreement scores do not consider cases where both annotators annotated a token as a non-entity.

| NE Type    | Seed      | Development | Test      |
|------------|-----------|-------------|-----------|
| DATE       | 498       | 249         | 843       |
| LOCATION   | 682       | 479         | 1,453     |
| MONEY      | 123       | 18          | 148       |
| ORGANIZATION | 464     | 219         | 966       |
| PERSON     | 267       | 172         | 601       |
| PRODUCT    | 381       | 103         | 382       |
| TIME       | 200       | 46          | 107       |
| Total      | 2,615     | 1,286       | 4,500     |

Table 3: Latvian and Lithuanian NE annotated corpora statistics.

The NE annotated data is stored in plaintext format containing MUC-7 style NE tags. A format sample is given in Figure 2. This format is also used when TildeNER performs automatic NER on user provided plaintext documents.

Figure 1: Disambiguation view of NESimpleAnnotator

After annotation both corpora were split in seed, development and test sets. The development set is used in refinement method parameter tuning and feature function selection processes and the test set is used for final evaluation. The NE statistics in the disambiguated corpora is shown in Table 3 for both Latvian and Lithuanian.
3. System Design

TildeNER is a named entity recognition toolkit that consists of multiple workflows for NER model training, NE tagging and evaluation. In training and tagging as a machine learning (ML) component TildeNER uses the conditional random field classifier StanfordNER (Finkel et al., 2005), which contains a large set of feature functions required in a supervised NER system (and does not require inventing a wheel a second time). The TildeNER system is developed in Perl and the StanfordNER system is a Java application. Both systems run on Windows and Linux operating systems.

3.1 Feature Function Selection

The feature functions for both Latvian and Lithuanian were selected using iterative minimum error-rate training. The method starts with a seed feature function set and in each iteration trains multiple (depending on the number of altering feature functions) NER models with altered (set to “true” or “false” or assigned a different value) feature functions where each model has a different feature function altered. The feature function set of the model, which increases the F-measure the most is selected as the base set for the next iteration.

Although such an iterative approach allows finding only the local maxima it is sufficient to select good performance feature functions. In the authors experiments in every iteration 85 different models were trained and the performance on Latvian development data increased from a token level F-measure of 63.29 to 69.47, which gives a significant increase on the system’s performance (although, on development data).

3.2 Data Pre-processing

The human annotated data and unlabelled data that is used in NER model training or tagging is pre-processed using a maximum entropy based morpho-syntactic tagger (Pinnis and Goba, 2011), which tokenizes, lemmatizes and morpho-syntactically tags the data. The tag is positional and contains 28 categories (for instance, part of speech, verb tense and mode, gender, number, case, required number and case agreement, etc.). The output of the tagger is tab-separated as shown in Figure 3. After morpho-syntactic tagging, positional information is added in order to trace every token from the tab-separated document back to its positions in the plaintext input document. In the case of gold annotated data also NE categories are assigned to each token. As introduced in the CoNLL 2002 conference (Tjong Kim Sang, 2002) the author also uses the BIO scheme for annotation of non-entity tokens and NE tokens (for instance, “B-ORG” and “I-ORG” for first and further tokens of an organization).

The data pre-processing step introduces a new feature function – the value of the morpho-syntactic tag. This feature function has been integrated in the StanfordNER conditional random field (CRF) classifier used by TildeNER. It can be used as additional feature to describe the context around a token in the range from one to N (depending on the configuration) tokens to the left and to the right from each token. The whole positional tag is used as a feature.

3.3 NER Model Bootstrapping

The NE annotated corpora for Latvian and Lithuanian are relatively small compared to data sets that are used, for instance, for English NER system development and model training. Therefore, TildeNER features a NER model bootstrapping module, which employs a bootstrapping method similar to Liao and Veeramachaneni (2009).

In order to bootstrap a NER model the system requires a set of seed, development and test data (human annotated data). Additionally to the human annotated data unlabelled data is required (for instance, in author’s experiments articles from Wikipedia and Web news were used as sources of unlabelled data). All four sets have to be pre-processed in order to run the bootstrapping workflow. The overall bootstrapping design, including pre-processing steps, is shown in Figure 4. Once all data is available, the bootstrapping system iteratively:

- Trains a NER model. In the first iteration only seed data is used as training data. In further iterations, additionally to the seed data, new training data, which...
is extracted in previous iterations, is used.

- Evaluates the trained model on development and test data. The system provides also functionality that enforces only positive iteration usage (specified in the configuration), dropping all iterations that decrease performance on the development set. Iteration is considered positive if it increases either precision, recall or F-measure (also defined through the configuration).

Figure 4: Design of the bootstrapping workflow
• Tags the unlabelled data with the newly trained NER model. In the case if the configuration requires only positive iteration data propagation and the current trained model decreases performance, unlabelled data is tagged with a model from the last positive iteration.
• Extracts new training data. After the unlabelled data is tagged with the trained NER model, new training data is extracted. Sentences that contain NEs, which have been annotated with the heuristic and statistical refinement methods, are ranked and the top N sentences of each NE category are selected as new training data. It is important in this step to use good refinement methods that are able to tag new and unseen by the supervised classifier NEs. If the raw data that the NER classifier outputs is used, the bootstrapping learns only the cases that it already knows as the supervised classifier’s performance on unseen data is unreliable.
• Extracts new gazetteer data from the newly tagged unlabelled data. This step is optional, but can be used in automatic gazetteer bootstrapping. The system also allows using the extracted NE lists in training of further iteration NER models.

3.4 Refinement Methods
In NER model bootstrapping as well as tagging, TildeNER applies refinement methods in order to improve upon the NE classification results produced by the StanfordNER CRF classifier. During bootstrapping the refinements help finding new unseen data examples and in tagging refinements allow achieving either better precision or recall (depending on the configuration of the refinement methods).
Refinement methods are functions that analyse a document and re-classify tokens or sequences of tokens as named entities or non-entities. The following refinements have been implemented so far in TildeNER:
• Removal of unlikely NEs. Named entities that are classified by the CRF classifier below a configured threshold are re-classified as non-entities (increases precision).
• Consolidation of equal lemma sequences. In NER a common assumption is to classify equal NEs with the same category (one sense per discourse rule). This method analyses such cases and decides whether for certain NEs, which are classified as being of multiple categories, one category, which is the most likely, can be identified. Misclassified entities in such situations are re-classified (increases precision). This method is important as the CRF classifier does not observe the whole context, but rather a limited window and is not able to realise the one sense per discourse rule.
• Enforcing equal lemma sequences to be tagged (increases recall). Similarly as in the previous method, the CRF classifier tends not only misclassify, but also miss some NEs in different contexts (mostly in contexts unknown to the NER model). This method classifies lemma sequences that are misclassified as non-entities if there exists a NE that is classified with a confidence score of over a configurable threshold and has the same lemma sequence as the non-entity sequence. This refinement method also enforces the one sense per discourse rule.
• NE border correction for entities, which contain an odd number of quotation marks or brackets (increases both precision and recall). When bootstrapping, the new training data tends to contain classified sequences that lack, for instance, a bracket or a quotation mark, because the classifier’s confidence has been too low to tag the misclassified token as part of the NE. This issue occurs mostly for NEs spanning over five and more tokens. If not controlled, such cases decrease system’s performance over bootstrapping iterations. Therefore, this method tries to expand or reduce the NEs containing bracketing and quotation mistakes.
• Artefact removal methods (increase precision). Applying the NER system to different domains, some in-domain artefacts (for instance, hyperlinks in web crawled documents, some leftover mark-up from corpora processing, etc.) can occur in texts.
• Person name analysis (increases recall). As person names may consist of multiple tokens (first name, middle name, last name, title, etc.), the refinement method splits all person NEs, which CRF classifier’s confidence score is above a configurable threshold, in separate tokens and tags non-entity tokens that match with the NEs respective tokens.
• Sentence beginning classification validation (increases recall). Sentence beginnings have proven to be difficult cases for NER as the capitalized tokens may be misleading. If the CRF classifier classifies a token as a NE, but it can be found elsewhere in the same document as a common (lowercased) word and lowercased, the sentence beginning misclassified NE is re-classified as a non-entity.
Refinement methods can be applied in any required sequence by passing a “refinement order definition string” when running TildeNER. This allows boosting the system’s performance by either recall or precision (and in some cases by both).

4. Evaluation
4.1 Non-comparative Evaluation
As a baseline the author uses the supervised system (without bootstrapping and refinements) trained with only the StanfordNER CRF classifier using the feature functions selected in the iterative minimum error rate training. Table 4 shows the baseline performance with an F-measure of 54.28 for Latvian and 62.70 for Lithuanian on full NEs (border detection and equal categories).
An obvious question is: “Why is there such a huge difference?” The answer is quite simple – the test sets and training sets vary in content complexity. For instance, the Lithuan texts feature automatically web crawled data, which includes also extracted tables with vague structure (space or tab separated), many short fragments with missing context, as well as many fragments with comma separated NEs.
The Lithuanian corpora, on the other hand, is manually selected and extracted from news portals, Wikipedia and other sources, therefore, features less complex structures. All these points result in lower Latvian results on the test set and if comparison between the two system evaluations is done, test data complexity has to be taken into account. Once the baseline systems were prepared, the refinement method parameters and the refinement method application sequence were tuned on the development set and if comparison between the two system evaluations total of 75% of errors are caused by missing NE’s in the tagged data, 15% are caused by incorrect border detection and the remaining 10% are wrong category classification mistakes.

The baseline CRF based system are given in Table 4. Using bootstrapped models (with the respective refinement configurations), precision and F-measure can be increased by up to 4.92% over the refined supervised results for full NEs and up to 16.55% for precision and up to 5.91% for F-measure over the baseline systems. For comparison, Czech (Kravalová and Žabokrtský, 2009), who also feature a morphologically rich language with different NE capitalization rules as in English, achieve an F-measure of 0.71 using 10 NE categories and a corpus twice as large). In the precision bootstrapped NER model for Latvian a total of 75% of errors are caused by missing NE’s in the tagged data, 15% are caused by incorrect border detection and the remaining 10% are wrong category classification mistakes.

### 4.2 Experimental Comparative Evaluation

In order to better understand the performance figures and to be able to better compare results to different language NER systems, for experimental purposes a comparative evaluation on parallel and strongly comparable corpora was performed. The reasoning, why parallel and strongly comparable corpora is used, is such that in parallel (and also strongly comparable) documents NE coverage and the document structural complexity is the same (or at least very close) for both languages, thus the system performance on the data, even if from two different languages, can be compared. As TildeNER relies on the StanfordNER CRF classifier, for comparative evaluation a Stanford NER model⁵ that

| System                                           | Latvian (Only CRF Classifier) | Lithuanian (CRF + refinement methods) tuned for precision | Lithuanian (CRF + refinement methods) tuned for F-measure |
|--------------------------------------------------|-------------------------------|-----------------------------------------------------------|-----------------------------------------------------------|
| System                                           | Precision | Recall | Accuracy | F-measure | Precision | Recall | Accuracy | F-measure |
| Token                                            | 74.70     | 56.23  | 91.02    | 64.16     | 74.44     | 63.54  | 92.30    | 68.56     |
| Full NE                                          | 62.43     | 48.01  | -        | 54.28     | 67.42     | 58.60  | -        | 62.70     |
| **Baseline (CRF + refinement methods) tuned for precision** | **Token** | **Recall** | **Accuracy** | **F-measure** | **Precision** | **Recall** | **Accuracy** | **F-measure** |
| Token                                            | 86.47     | 41.51  | 88.86    | 56.09     | **84.04** | 53.74  | 91.53    | 65.56     |
| Full NE                                          | 75.61     | 35.05  | -        | 47.90     | **77.01** | 49.63  | -        | 60.36     |
| **Baseline (CRF + refinement methods) tuned for F-measure** | **Token** | **Recall** | **Accuracy** | **F-measure** | **Precision** | **Recall** | **Accuracy** | **F-measure** |
| Token                                            | 74.63     | 57.15  | 91.17    | 64.73     | 76.31     | 63.50  | **92.47** | 69.32     |
| Full NE                                          | 62.32     | 49.66  | -        | 55.27     | 68.57     | 59.39  | -        | 63.65     |
| **Bootstrapped (CRF + refinement methods + bootstrapping) for better precision** | **Token** | **Recall** | **Accuracy** | **F-measure** | **Precision** | **Recall** | **Accuracy** | **F-measure** |
| Token                                            | 87.27     | 45.17  | 89.57    | 59.53     | -         | -      | -        | -         |
| Full NE                                          | 79.18     | 41.85  | -        | 54.76     | -         | -      | -        | -         |
| **Bootstrapped (CRF + refinement methods + bootstrapping) for better F-measure** | **Token** | **Recall** | **Accuracy** | **F-measure** | **Precision** | **Recall** | **Accuracy** | **F-measure** |
| Token                                            | 75.55     | **61.34** | **91.86** | **67.71** | 76.90     | **63.77** | **92.42** | **69.72** |
| Full NE                                          | 64.98     | **56.06** | -        | **60.19** | 71.32     | **59.91** | -        | **65.12** |

Table 4: Evaluation results on test data.
achieves an F-measure of 93.0 for English on the “CoNLL 2003 testa” data set⁶ was selected. For the comparative evaluation a set of 10 documents (5 parallel and 5 strongly comparable) was selected. The comparable documents are Wikipedia articles and European Commission bilingual news articles, but the parallel documents are legal documents. NEs in both languages were annotated by a human annotator in order to create a reference (gold) data set for evaluation. The corpora statistics is shown in Table 5.

| NE Type     | English | Latvian |
|-------------|---------|---------|
| ORGANIZATION| 441     | 404     |
| LOCATION    | 291     | 329     |
| PERSON      | 113     | 148     |
| Total       | 845     | 881     |

Table 5: Comparative evaluation corpora statistics for English-Latvian.

The NE types were limited to organization, person and location. The evaluation results are shown in Table 6.

|                      | Precision | Recall | F-measure |
|----------------------|-----------|--------|-----------|
| StanfordNER          |           |        |           |
| LOCATION             | 37.5      | 31.91  | 34.48     |
| PERSON               | 37.12     | 45.37  | 40.83     |
| ORGANIZATION         | 60.89     | 70.89  | 65.51     |
| Latvian bootstrapped for better precision |           |        |           |
| LOCATION             | 76.47     | 39.63  | 52.21     |
| PERSON               | 76.27     | 30.41  | 43.48     |
| ORGANIZATION         | 93.16     | 44.14  | 59.90     |
| Latvian bootstrapped for better F-measure |           |        |           |
| LOCATION             | 63.85     | 50.61  | 56.46     |
| PERSON               | 54.08     | 71.62  | 61.63     |
| ORGANIZATION         | 77.82     | 56.86  | 65.71     |

Table 6: English-Latvian comparative evaluation results.

The comparative evaluation results suggest that even if the results of TildeNER are lower than state-of-the-art English NER system results, those cannot be compared without taking test set characteristics into account. The results also suggest that TildeNER for Latvian performs slightly better for location and person name NEs on the 10 document comparative evaluation scenario.

One important note when analysing the results has to be also taken into account – the test set of the comparative evaluation is more in favour of the TildeNER Latvian NER system as that has been trained on a mixed set of documents including also Wikipedia articles, which are out of domain articles for the StanfordNER English model. Nevertheless, the methodology of bilingual comparative evaluation is a means to compare NERs from different languages.

5. Conclusion

In this paper the author presented TildeNER - a NER system developed for two Baltic languages for which supervised and semi supervised ML methods for NER had not been applied before. Although, the results show improvements in F-measure using raw data refinement methods as well as F-measure targeted bootstrapping, the methods have to be improved in order to make a significant increase over the supervised learning models. Refinement methods and their capability in finding new and unseen data is one of the most important requirements for a successful NER model bootstrapping system that is based on supervised learning-based classification.

The toolkit TildeNER offers large configuration possibilities for various NER tasks (aid in question answering, automatic gazetteer extraction, machine translation, keyword extraction, etc.) where different requirements for higher precision or higher F-measure can be set.

TildeNER is released under the Apache 2.0⁷ licence and can be freely acquired through the Toolkit for multi-level alignment and information extraction from comparable corpora, a public deliverable of the ACCURAT project (http://www.accurat-project.eu/index.php?p=toolkit).

Future work on TildeNER will involve more fine-grained Latvian and Lithuanian morpho-syntactic feature integration in the CRF classifier. Currently the whole morpho-syntactic tag is used as a single feature function, ignoring that some of the properties within the positional tag may be independent and can be used, for instance, in NE border disambiguation, category classification, etc. Also much can be done with refinement methods in order to find better candidates in bootstrapping as well as to improve tagging quality in terms of precision and recall.

6. Acknowledgements

The research within the project Accurat leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013), grant agreement no 248347.

7. References

Chinchor, N. (1998), MUC-7 Named Entity Task Definition. In: Proceedings of the Seventh Message Understanding Conference (MUC-7).

Cohen, J. (1968), Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit, Psychological Bulletin 70 (4) (October, 1968), pp. 213–220.

Finkel, J., Grenager, T., and Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In: Proceedings of the 43rd Annual Meeting on Association for

⁶ As reported by University of Stanford in: http://nlp.stanford.edu/software/crf-faq.shtml (point 11)

⁷ Apache 2.0 licence: http://www.apache.org/licenses/LICENSE-2.0.html
Kravalová, J. and Žabokrtský, Z. (2009). Czech named entity corpus and SVM-based recognizer. In: Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration, Association for Computational Linguistics, pp. 194–201.

Liao, W. and Veeramachaneni, S. (2009). A simple semi-supervised algorithm for named entity recognition. In: Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing, Association for Computational Linguistics, pp. 58–65.

Nadeau, D. and Sekine, S. (2007). A survey of named entity recognition and classification. Lingvisticae Investigationes, Vol. 30, No. 1. (January, 2007), pp. 3–26.

Pinnis M. and Goba K. (2011). Maximum Entropy Model for Disambiguation of Rich Morphological Tags. In: Proceedings of the Second Workshop on Systems and Frameworks for Computational Morphology, Communications in Computer and Information Science, Vol. 100, Springer, pp. 14–22.

Skadiņa, I. (2009). Jaunas iespējas attēlu meklēšanā: ģeotelpiskajā informācijā un valodu tehnoloģijās balsīta attēlu meklēšanas platforma TRIPOD. In: Latvijas nacionālās bibliotēkas zinātniskie raksti, National Library of Latvia.

Tjong Kim Sang, E.F. (2002). Introduction to the conll-2002 shared task: Language independent named entity recognition. In: Proceedings of CoNLL-2002, pp. 155–158.