Supporting Web-based Address Extraction with Unsupervised Tagging

Berenike Loos[1] and Chris Biemann[2]

[1] European Media Laboratory GmbH, Schloss-Wolfsbrunnenweg 33, 69118 Heidelberg, Germany berenike.loos@eml-d.villa-bosch.de
[2] University of Leipzig, NLP Department, Johannisgasse 26, 04103 Leipzig, Germany biem@informatik.uni-leipzig.de

Abstract. The manual acquisition and modeling of tourist information as e.g. addresses of points of interest is time and, therefore, cost intensive. Furthermore, the encoded information is static and has to be refined for newly emerging sight seeing objects, restaurants or hotels. Automatic acquisition can support and enhance the manual acquisition and can be implemented as a run-time approach to obtain information not encoded in the data or knowledge base of a tourist information system. In our work we apply unsupervised learning to the challenge of web-based address extraction from plain text data extracted from web pages dealing with locations and containing the addresses of those. The data is processed by an unsupervised part-of-speech tagger (Biemann, 2006a), which constructs domain-specific categories via distributional similarity of stop word contexts and neighboring content words. In the address domain, separate tags for street names, locations and other address parts can be observed. To extract the addresses, we apply a Conditional Random Field (CRF) on a labeled training set of addresses, using the unsupervised tags as features. Evaluation on a gold standard of correctly annotated data shows that unsupervised learning combined with state of the art machine learning is a viable approach to support web-based information extraction, as it results in improved extraction quality as compared to omitting the unsupervised tagger.

1 Introduction

When setting up a Natural Language Processing (NLP) system for a specific domain or a new task, one has to face the acquisition bottleneck: creating resources such as word lists, extraction rules or annotated texts is expensive due to high manual effort. Even in times where rich resource repositories exist, these often do not contain material for very specialized tasks or for non-English languages and, therefore, have to be created ad-hoc whenever a new task has to be solved as a component of an application system. All methods that alleviate this bottleneck mean a reduction in time and cost. Here, we demonstrate that unsupervised tagging substantially increases performance in a setting where

only limited training resources are available. As an application, we operate on automatic address extraction from web pages for the tourist domain.

1.1 Motivation: Address Extraction from the Web

In an open-domain spoken dialog system, the automatic learning of ontological concepts and corresponding relations between them is essential as a complete manual modeling of them is neither practicable nor feasible due to the continuously changing denotation of real world objects. Therefore, the emergence of new entities in the world entails the necessity of a method to deal with those entities in a spoken dialog system as described in Loos (2006).

As a use case to this challenging problem we imagine a user asking the dialog system for a newly established restaurant in a city, e.g. ("How do I get to the *Auerstein*"). So far, the system does not have information about the object and needs the help of an incremental learning component to be able to give the demanded answer to the user. A classification as well as any other information for the word "Auerstein" are hitherto not modeled in the knowledge base and can be obtained by text mining methods as described in Faulhaber et al. (2006). As soon as the object is classified and located in the system's domain ontology, it can be concluded that it is a building and that all buildings have addresses. At this stage the herein described work comes into play, which deals with the extraction of addresses in unstructured text. With a web service (as part of the dialog system's infrastructure) the newly found address for the demanded object can be used for a route instruction.

Even though structured and semi-structured texts such as online directories can be harvested as well, they often do not contain addresses of new places and do, therefore, not cover all addresses needed. However, a search in such directories can be used in combination with the method described herein, which can be used as a fallback solution.

1.2 Unsupervised Learning Supporting Supervised Methods

Current research in supervised approaches to NLP often tries to reduce the amount of human effort required for collecting labeled examples by defining methodologies and algorithms that make a better use of the training set provided. Another promising direction to tackle this problem is to empower standard learning algorithms by the addition of unlabeled data together with labeled texts. In the machine learning literature, this learning scheme has been called semi-supervised learning (Sarkar and Haffari, 2006). The underlying idea behind our approach is that syntactic and semantic similarity of words is an inherent property of corpora, and that it can be exploited to help a supervised classifier to build a better categorization hypothesis, even if the amount of labeled training data provided for learning is very low. We emphasize that every contribution to widening the acquisition bottleneck is useful, as long as its application does not cause more extra work than the contribution

is worth. Here, we provide a methodology to plug an unsupervised tagger into an address extraction system and measure its contribution.

2 Data preparation

In our semi-supervised setting, we require two different data sets: a small, manually annotated dataset used for training our supervised component, and a large, unannotated dataset for training the unsupervised part of the system. This section describes how both datasets were obtained. For both datasets we used the results of Google queries for places as restaurants, cinemas, shops etc. To obtain the annotated data set, 400 of the resulting Google pages for the addresses of the corresponding named entities were annotated manually with the labels: `street`, `house`, `zip` and `city`, all other tokens received the label `O`.

As the unsupervised learning method is in need of large amounts of data, we used a list with about 20,000 Google queries each returning about 10 pages to obtain an appropriate amount of plain text. After filtering the resulting 700 MB raw data for German language and applying cleaning procedures as described in (Quasthoff et al., 2006) we ended up with about 160 MB totaling 22.7 million tokens. This corpus was used for training the unsupervised tagger.

3 Unsupervised Tagging

3.1 Approach

Unlike in standard (supervised) tagging, the unsupervised variant relies neither on a set of predefined categories nor on any labeled text. As a tagger is not an application of its own right, but serves as a pre-processing step for systems building upon it, the names and the number of syntactic categories is very often not important.

The system presented in Biemann (2006a) uses Chinese Whispers clustering (Biemann, 2006b) on graphs constructed by distributional similarity to induce a lexicon of supposedly non-ambiguous words with respect to part of speech (PoS) by selecting only safe bets and excluding questionable cases from the category building process. In this implementation two clusterings are combined, one for high and medium frequency words, the other collecting medium and low frequency words. High and medium frequency words are clustered by similarity of their stop word context feature vectors: a graph is built, including only words that are endpoints of high similar pairs. Clustering this graph of typically 5,000 vertices results in several hundred clusters, which are subsequently used as PoS categories. To extend the lexicon, words of medium and low frequency are clustered using a graph that encodes similarity of significant neighbor co-occurrences (as defined in Dunning, 1993). Both

clusterings are mapped by overlapping elements into a lexicon that provides PoS information for some 50,000 words.

For obtaining a clustering on datasets of this size, an effective algorithm like Chinese Whispers is crucial. Increased lexicon size is the main difference between this and other approaches (e.g. (Schütze, 1995), (Freitag , 2004)), that typically operate with 5,000 words. Using the lexicon, a trigram tagger with a morphological extension is trained, which can be used to assign tags to all tokens in a text. The tag sets obtained with this method are usually more fine-grained than standard tag sets and reflect syntactic as well as semantic similarity. In Biemann (2006a), the tagger output was directly evaluated against supervised taggers for English, German and Finnish via information-theoretic measures. While it is possible to relatively compare the performance of different components of a system or different systems along this scale, it does only give a poor impression on the utility of the unsupervised tagger's output. Therefore, an application-based evaluation is undertaken here.

3.2 Resulting Tagset

As described in Section 2, we had a relatively small corpus in comparison to previous work with the same tagger, that typically operates on about 50 million tokens. Nonetheless, the domain specifity of the corpus leads to an appropriate tagging, which can be seen in the following examples from the resulting tag set (numbers in brackets give the words in the lexicon per tag):

1. Nouns: *Verhandlungen, Schritt, Organisation, Lesungen, Sicherung,...* (800)
2. Verbs: *habe, lernt, wohnte, schien, hat, reicht, suchte...* (191)
3. Adjectives: *französischen, künstlerischen, religiösen...* (142)
4. locations: *Potsdam, Passau, Innsbruck, Ludwigsburg, Jena...* (320)
5. street names: *Bismarckstr, Leonrodstr, Schillerstr, Ungererstr...* (150)

On the one hand, big clusters are formed that contain syntactic tags as shown for the example tags 1 to 3. Items 4 and 5 show that not only syntactic tags are created by the clustering process, but also domain specific tags, which are useful for an address extraction. Note that the actual tagger is capable of tagging all words, not only words in the lexicon – the number of words in the lexicon are merely the number of types used for training. We emphasize that the comparatively small training corpus (usually, 50M–500M tokens are employed) leaves room for improvements, as more training text showed to have a positive impact on tagging quality in previous studies.

4 Experiments and Evaluation

This section describes the supervised system, the evaluation methodology and the results we obtained in a comparative evaluation of either providing or not providing the unsupervised tags.

4.1 Conditional Random Field Tagger

We perceived address extraction as a tagging task: labels indicating `city`, `street`, `house` number, `zip` code or other (`O`) from the training set are learned and applied to unseen examples. Note that this is not comparable to a standard task like Named Entity Recognition (cf. Roth and van den Bosch, 2002), since we are only interested in labeling the address of the target location, and not other addresses that might be contained in the same document. Rather, this is an instance of Information Extraction (see Grishman, 1997). For performing the task, we train the MALLET tagger (McCallum, 2002), which is based on Conditional Random Fields (CRFs, see Lafferty et al. 2001). CRFs define a conditional probability distribution over label sequences given a particular observation sequence. CRFs have been proven to have equal or superior performance at tagging tasks as compared to other systems like Hidden Markov Models or the Maximum Entropy Framework. The flexibility of CRFs to include arbitrary, non-independent features allows us to supply unsupervised tags or no tags to the system without changing the overall architecture. The tagger can operate on a different set of features ranging over different distances. The following features per instance are made available to the CRF:

- word itself
- relative position to target name
- unsupervised tag

We experimented with different orders as well as with different time shifts.

CRF Order

The order of the CRF defines how many preceding labels are used for the determination of the current label. An order of 1 means that only the previous label is used, order 2 allows for the usage of two previous labels etc. As higher orders mean more information, which is in turn supported by fewer training examples, an optimum at some small order can be expected.

Time Shifting

Time shifting is an operation that allows the CRF to use not only the features for the current position, but also features from surrounding positions. This is reached by copying the features from surrounding positions, indicating what relative position they were copied from. As with orders, an optimum can be expected for some small range of time shifting, exhibiting the same information/sparseness trade-off. For illustration, the following listing shows an original training instance with time shift 0, as well as the same instance with time shifts -2, -1, 0, 1, 2, for the scenario with unsupervised tags. Note that relative positions are not copied in time-shifting because of redundancy. The following items show these shifts:

- **shift 0:**
  - Extrablatt 0 T115 `O`
  - 53 1 T215 `house`
  - Hauptstr 2 T64 `street`
  - Heidelberg 3 T15 `city`
  - 69117 4 T215 `zip`
- **shift 1:**
  - 1 -1:Extrablatt -1:T115 0:53 0:T215 1:Hauptstr 1:T64 `house`
  - 2 -1:53 -1:T215 0:Hauptstr 0:T64 1:Heidelberg 1:T15 `street`
- **shift 2:**
  - 1 -2:Cafe -2:T10 -1:Extrablatt -1:T115 0:53 0:T215 1:Hauptstr 1:T64 2:Heidelberg 2:T15 `house`
  - 2 -2:Extrablatt -2:T115 -1:53 -1:T215 0:Hauptstr 0:T64 1:Heidelberg 1:T15 2:69117 2:T215 `street`

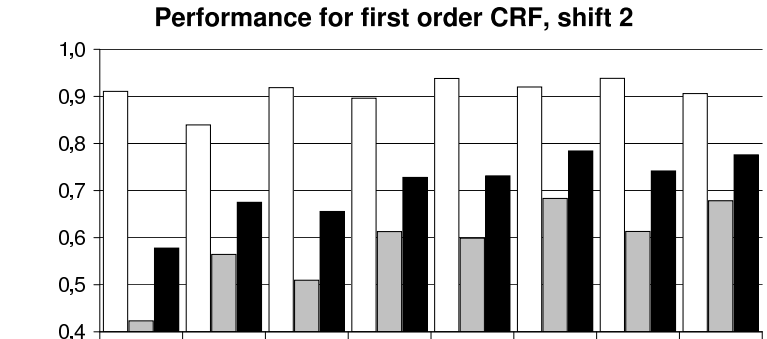
In the example for shift 0 a full address with all features is shown: word, relative position to target "Extrablatt", unsupervised tag and classification label. For exemplifying shifts 1 and 2, only two lines are given, with -2:, -1:, 0:, 1: and 2: being the relative position of copied features. In the scenario without unsupervised tags all features "T<number>" are omitted.

4.2 Evaluation Methodology

For evaluation, we split the training set into 5 equisized parts and performed 5 sub-experiments per parameter setting and scenario, using 4 parts for training and the remaining part for evaluation in a 5-fold-cross-validation fashion. The split was performed per target location: locations in the test set were never contained in the training. To determine our system's performance, we measured the amount of correctly classified, incorrectly classified (false positives) and missed (false negatives) instances per class and report the standard measures Precision, Recall and F1-measure as described in Rijsbergen (1979). The 5 sub-experiments were combined and checked against the full training set.

4.3 Results

Our objective is to examine to what extent the unsupervised tagger influences classification results. Conducting the experiments with different CRF parameters as outlined in Section 4.1, we found different behaviors for our four target classes: whereas for `street` and `house` number, results were slightly better in the second order CRF experiments, the first order CRF scored clearly higher for `city` and `zip` code. Restricting experiments to first order CRFs and regarding different shifts, a shift of 2 in both directions scored best for all classes except `city`, where both shift 0 and 1 resulted in slightly higher scores. The best overall setting, therefore, was determined to be the first order CRF with



	street		house		zip code		city	
	untagged	tagged	untagged	tagged	untagged	tagged	untagged	tagged
Precision	0,911	0,840	0,919	0,896	0,938	0,920	0,938	0,906
Recall	0,423	0,564	0,510	0,613	0,599	0,683	0,613	0,678
F1	0,578	0,675	0,656	0,728	0,731	0,784	0,742	0,776

Fig. 1. Results in precision, recall and F1 for all classes, obtained with first order CRF and a shift of 2.

a shift of 2. For this setting, Figure 1 presents the results in terms of precision, recall and F1.

What can be observed not only from Figure 1 but also for all parameter settings is the following: Using unsupervised tags as features as compared to no tagging leads to a slightly decreased precision but a substantial increase in recall, and always affects the F1 measure positively. The reason can be sought in the generalization power of the tagger: having at hand syntactic-semantic tags instead of merely plain words, the system is able to classify more instances correctly, as the tag (but not the word) has occurred with the correct classification in the training set before. Due to overgeneralization or tagging errors, however, precision is decreased. The effect is strongest for `street` with a loss of 7% in precision with a recall boost of 14%.
In general, unsupervised tagging clearly helps at this task, as a little loss in precision is more than compensated with a boost in recall.

5 Conclusion and Further Work

In this research we have shown that the use of large, unannotated text can improve classification results on small, manually annotated training sets via building a tagger model with unsupervised tagging and using the unsupervised tags as features in the learning algorithm. The benefit of unsupervised tagging is especially significant in domain-specific settings, where standard pre-processing steps such as supervised tagging do not capture the abstraction granularity necessary for the task, or simply no tagger for the target language is available. For further work, we aim at combining the possibly several addresses per target location. Given the evaluation values obtained with

our method, the task of dynamically extracting addresses from web-pages to support address search for the tourist domain is feasible and a valuable, dynamic add-on to directory-based address search.

References

BIEMANN, C. (2006a): Unsupervised Part-of-Speech Tagging Employing Efficient Graph Clustering. *Proc. COLING/ACL-06 SRW*, Sydney, Australia.

BIEMANN, C. (2006b): Chinese Whispers - an Efficient Graph Clustering Algorithm and its Application to Natural Language Processing Problems. *Proceedings of the HLT-NAACL-06 Workshop on Textgraphs*, New York, USA.

DUNNING, T. (1993): Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics 19(1), pp. 61–74.*

FAULHABER A., LOOS B., PORZEL R., MALAKA, R. (2006): Towards Understanding the Unknown: Open-class Named Entity Classification in Multiple Domains. *Proceedings of the Ontolex Workshop at LREC*, Genova, Italy

FREITAG, D. (2004): Toward unsupervised whole-corpus tagging. *Proceedings of the 20th International Conference on Computational Linguistics*, Geneva, Switzerland

GRISHMAN, R. (1997): Information Extraction: Techniques and Challenges. In Maria Teresa Pazienza (ed.) *Information Extraction.* Springer-Verlag, Lecture Notes in Artificial Intelligence, Rome

LAFFERTY, J. and McCALLUM, A. K. and PEREIRA, F. (2001): Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *Proceedings of ICML-01*, pp. 282–289.

LOOS, B. (2006): On2L A Framework for Incremental Ontology Learning in Spoken Dialog Systems. *Proc. COLING/ACL-06 SRW*, Sydney, Australia

MCCALLUM, A. K. (2002): MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu.

QUASTHOFF, U., RICHTER, M. and BIEMANN, C. (2006): Corpus Portal for Search in Monolingual Corpora. *Proceedings of LREC-06*, Genoa, Italy

ROTH, D. and VAN DEN BOSCH, A. (Eds.) (2002): Proceedings of the Sixth Workshop on Computational Language Learning (CoNNL-02), Taipei, Taiwan.

SARKAR, A. and HAFFARI, G. (2006): Inductive Semi-supervised Learning Methods for Natural Language Processing. *Tutorial at HLT-NAACL-06*, NYC, USA.

SCHÜTZE, H. (1995): Distributional part-of-speech tagging. *Proceedings of the 7th Conference on European chapter of the Association for Computational Linguistics*, Dublin, Ireland

VAN RIJSBERGEN, C. J. (1979): *Information Retrieval, 2nd edition.* Dept. of Computer Science, University of Glasgow.