Collective Ontology-based Information Extraction using Probabilistic Graphical Models

Slavko Žitnik[1,2]

[1] University of Ljubljana, Faculty of Computer and Information Science,
Tržaška cesta 25, SI-1000 Ljubljana
[2] Optilab d.o.o., Teslova 30, SI-1000 Ljubljana,
slavko.zitnik@fri.uni-lj.si
This doctoral research is being led under the supervision of prof. dr. Marko Bajec[1].

Abstract. Information Extraction (IE) is a process of extracting structured data from unstructured sources. It roughly consists of subtasks named entity recognition, relation extraction and coreference resolution. Researchers have primarily focused just on one subtask or their combination in a pipeline. In this paper we introduce an intelligent collective IE system combining all three subtasks by employing conditional random fields. The usage of same learning model enables us to easily communicate between iterations on the fly and to correct errors during iterative process execution. In addition to the architecture we introduce novel semantic and collective feature functions. The system's output is labelled according to an ontology and new instances are automatically created during runtime. The ontology as a schema encodes a set of constraints, defines optional manual rules or patterns and with instances provides semantic gazetteer lists. The proposed framework is being developed during ongoing PhD research. It's main contributions are intelligent iterative interconnection of the selected subtasks, extensive use of context-specific features and parameterless system that can be guided by an ontology. Some preliminary results combining just two subtasks already show promising results over traditional approaches.

Keywords: information extraction, named entity recognition, relation extraction, coreference resolution, ontology

1 Introduction

Machine understanding of textual documents has been challenging since early computer-era. Information Extraction (IE) is a subfield of Information Retrieval that attempts to analyze text and extract its structured semantic contents. Main IE tasks consist of named entity recognition (e.g. extraction of person names, locations, organizations), relation extraction (i.e. identification of relations among entities) and coreference resolution (i.e. clustering of mentions to an entity).

Early IE systems were naive and rule-based, then (semi-) automatic approaches such as wrapper generation, seed expansion and rule induction were introduced and recently machine learning techniques gained popularity. In contrast to standard multi-label and regression classifiers, sequence taggers such as Hidden Markov Models and Conditional Random Fields have become most successful. Especially latter as they support rich feature function generation. Ontology-based IE systems have also emerged to assist the development of Semantic Web [13]. Here ontologies are extensively used to support the IE process by its knowledge representation.

To further improve IE systems, we propose a Collective Ontology-based IE system that iteratively combines all three main IE subtasks. Those are named entity recognition (NE) - identification of entity types (e.g. person names), relation extraction (RE) - identification of relation types between entities (e.g. [PERSON] *lives in* [LOCATION]) and coreference resolution (COREF) - clustering of mentions to the same entity (e.g. to link *John* and *He*). We employ linear-chain conditional random fields for NE and RE and skip-chain for COREF task. The use of same learning techniques enables us to reuse feature functions across tasks and by machine learning learn the incorporation of intermediate results during iterations. Next to labeling tasks we include entity resolution technique for matching and merging of coreferent mentions during IE process. Furthermore we identify additional semantic and iterative feature functions, taking into account multiple possible labelings in order to build error-robust system. The system is completely parameterless but results can be affected by ontology manipulation or by runtime as ontology can be automatically populated. Therefore the proposed system is not just a mashup of some already existing techniques, but intelligently interconnects all of the components.

The rest of the paper is structured as follows. Section 2 overviews related work, next in Section 3 we point out motivation for the research and present a toy example showing lack of current methods. Section 4 briefly describes data representation, conditional random fields classifier and then gives an architecture overview of proposed system. Lastly, we discuss current achievements and introduce future work and then conclude the paper in Section 6.

2 Related Work

There has been a lot of work done for specific IE subtasks separately [11]. The most often researched NER [3] task is relatively well solved and state-of-the-art approaches can achieve 90% or more F-score on general datasets. In contrast RE [7] and COREF [2] tasks are not yet well solved as 70% or more F-score is best what state-of-the-art algorithms can achieve on general data. We believe we can get better results using our approach. The work on IE is also driven by challenges at MUC[3], CoNLL[4] conferences and ACE[5] program as they define specific IE task and also provide data.

[3] Message Understanding Conference

[4] Conference on Computational Language Learning

[5] Automatic Content Extraction

The term collective IE was to our knowledge first used by Bunescu and Mooney [1]. They focused only on iterative NER exploiting mutual influence between possible extractions. Later Nellec and Nazarenko proposed Ontology-based IE [8] that in a cyclic process combines NER and RE with knowledge integration using an ontology. The proposed system was completely rule based, but it pointed into the right direction. The most recent system, Felix [9], was presented by Niu et. al.. It is a general IE system based on logical and statistical rules that use Markov Logic Networks. The authors focused on scaling it to large datasets and definition of generally applicable rules. The interesting part is, that their task scheduler can combine all three IE tasks, but different algorithms are used for each task and system is mostly rule based.

Conditional random fields (CRF), a sequence modeling framework, were first presented by Lafferty et. al. [4] and have been since used on various sequence labeling tasks. Using proper text labeling and feature induction they were successfully applied to the task of NER [3], RE [7] and COREF [12].

3 Motivation

Nowadays applications try to combine data from various sources and apply logic over it, but there are a number of useful unstructured sources that, due to complex information extraction systems or low quality results remain unused (e.g. experiments on automatic software testing using specification documents). There are also many barriers in building IE systems like accuracy, efficiency, multilinguality or reuse [6] that decrease general IE use.

We believe best results in IE could be only achieved by using as much data context as possible and jointly work with subtasks of NE, RE and COREF. Apart from input data context and iteration generated context, system should have its own knowledge base and use third party (semi-) structured sources. Many research projects combine the subtasks in linear fashion or just focus on only one of the subtasks by taking others for granted.

The IE system should be easy to use and if possible completely parameterless. There should also be minimal effort to change system's knowledge domain if needed. It must also be modular in order to inter-change specific implementation or adapt it to another natural language. Results should be clearly understandable and also presented as a semantic graph according to the underlying ontology. The knowledge base should be automatically populated with labelled data on new documents. User should be able to add additional labelings or repair mislabelings through the system's graphical user interface. Such data would be used as training data at system's classifiers re-learning. These are next to problems of low accuracy performance on general data some ideas that we believe future IE systems should have.

Now we will show an example how to achieve better performance in IE. Let analyze the following toy example: "*Dr. Missikof and Kurowski work as researchers. They will give keynote at CAISE in Gdansk.*" We show possible labeling in Fig. 1. The traditional pipeline approach of NER, RE and COREF

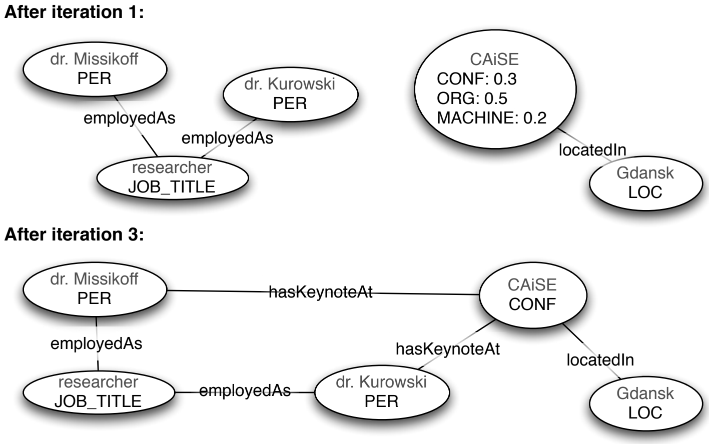

Fig. 1. A toy example of annotation improvement over iterations.

would output a result similar to a labeling after first iteration (see Fig. 1). As we see, the labeling of *CAISE* may be wrong due to missing additional information. Furthermore *They* was not yet co-referenced with person type and therefore relation *hasKeynotAt* was not identified. By enabling the system to iterate and take previous labelings into account, we would be able to use pronoun connection *They* and identify corresponding relation in second iteration. During third iteration *CAISE* would be successfully classified as a conference by using second labeling of corresponding relation and their argument types.

To efficiently solve identified problem, we designed a collective architecture which is in more detail defined in Section 4.3.

4 Framework proposal

In this section we present data representation, briefly introduce conditional random fields with feature functions and outline our architecture with training. Data representation presents data labeling that is needed to initially train CRFs and to mutually account intermediate labelings. CRF is a classifier that we have chosen to use for learning labeling models and is therefore crucial part in our architecture.

4.1 Data representation

The proposed architecture takes a raw natural text as an input and returns semantically annotated text as an output. We treat the tasks of NER, RE and COREF as sequence labeling tasks.

Let $\overline{x}^{k_i} = \{x_1^{k_i}, x_2^{k_i}, ..., x_n^{k_i}\}$ denote the sequence of observable tokens. Index k_i stands for input words w_i or additional attributes such as part-of-speech (lexical category) tags, phrase boundaries, entity cluster identifier or pre-calculated values. Each observable sequence is associated with corresponding labeling sequence $\overline{y}^{l_i}$ where $l_i \in \{NE, REL, COREF\}$ is defined for named entity, relation and coreference label tags.

We use common IOB notation [10] for NER and RE tasks. It defines tags starting with "B-" to denote start of a label type, "I-" the successor of the same type and "O" for other types. An example label tag set for person named entities is {B-PER, I-PER, O}. For relations we use labels {B-REL, I-REL, O}.

Coreference mentions can be represented as clusters, each referencing an entity. We therefore label each mention with corresponding cluster number.

Our problem is now finding the most probable labeling $\hat{y}^l$ for each of defined subtasks.

4.2 Conditional Random Fields

Conditional random fields (CRF) [5] are discriminative models and model a single joint distribution $p(\overline{y}|\overline{x})$ over the predicted sequence $\overline{y}$ conditioned on $\overline{x}$. Observable sequence $\overline{x}$ typically contains also a number of attributes that can be used when modeling feature functions. Training labels $\overline{y}$ relative to position i inside feature functions define the structure of graphical model which can in general be arbitrary.

Training CRF means looking for a weight vector w that assigns best possible labeling $\hat{y}$ given $\overline{x}$ for all training examples:

$$\hat{y} = \arg\max_{\overline{y}} p(\overline{y}|\overline{x}; w), \tag{1}$$

using conditional distribution

$$p(\overline{y}|\overline{x}; w) = \frac{\exp(\sum_{j=1}^{J} w_j \sum_{i=1}^{n} f_j(\overline{y}, \overline{x}, i))}{Z(\overline{x}, w)} \tag{2}$$

($Z(\overline{x}, w)$ is a normalization constant over all possible labelings of $\overline{y}$). When distance between two addressing labels y_i, y_j inside feature functions f_k is long, exact inference is intractable due to exponential number of partial sequences and thus approximate algorithms must be used. We therefore use feature functions that depend on single label (y_i) and two consecutive labels (y_{i-1}, y_i). This type of CRF is also known as linear chain CRF (LCCRF) which underlying graphical structure forms a chain and have been rather successfull in IE tasks. Using LCCRF, training and inference can be easily solved using forward–backward method and Viterbi algorithm. For COREF task better results are achieved if a few more distant labels are observed and therefore we use skip-chain CRF (SCCRF) [5] that are similar to LCCRF with additional long-distance edges. Training and inference is little harder but still faster than using arbitrary structure.

The modeling of feature functions is a crucial part when training CRFs. We divide used feature functions into the following groups:

Preprocessed: functions using bootstrap labelings (e.g. parse tree length, consecutive POS tags)
String: word shape features (e.g. suffix of length 2, upper case word, followed by a special symbol)
Semantic: ontology-derived features (e.g. entity type is argument of a relation, word contained as instance in knowledge base, is the same gender constraint)
Iteration: features taking into account intermediate labelings of NER, RE and COREF (e.g. relation argument possible types, entity has relation of type X)

4.3 System Architecture

In this section we show proposed system architecture and describe their components. Next, we introduce notion of iterative learning and labeling and define classes of used feature functions.

The proposed Collective Ontology-based IE system is presented in Fig. 2. The notion collective is used as all three subtasks iteratively refine the results and use each other's intermediate outputs. Main components of the system are the following:

Input consists of a set of textual documents.
Bootstrapping initially processes input data–i.e. splits it into tokens, sentences, lemmatizes it, performs Part-of-Speech tagging and full dependency parsing. This step initializes data with common labelings and enables the execution of iterative method.
Iterative method includes NER, RE, COREF and Entity merging and matching algorithms. The iteration continues until results converge or maximum number of iterations (will be statically defined by running some experiments) is reached. At each iteration, classifications from previous iterations are used by newly proposed feature functions. The convergence is achieved when the classifications over two consecutive iterations remain the same. All four methods tag data with additional annotations and take previous labelings into account. Entity matching and merging merges coreferent entities using relational entity resolution using attribute, relationship and semantic similarity measures [14]. This gives matched entities additional context information that is used by feature functions like other labelings.
Data sources provide structured data and are divided into two parts:
- **Ontology** defines concepts, constraints, rules and underlying instances in separate data store. It can be directly manipulated by the user. Additional feature functions we will design are going to use it. The use of instances will be the same as using gazetteer lists (list of known instances for an entity type). Additionally, ontology database will be populated at runtime with newly extracted instances.

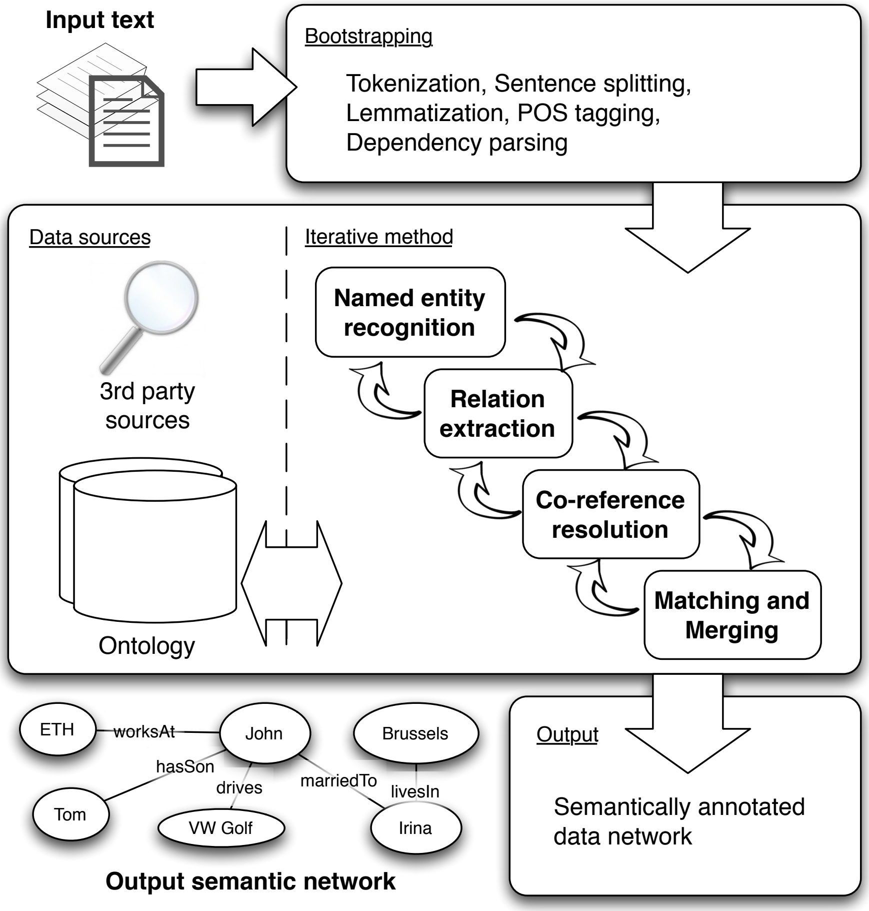

Fig. 2. Proposed Collective Ontology-based Information Extraction architecture.

- **3rd party sources** provide access to public services–e.g. phonebook, social network APIs. When extracted data types matches specific source parameters, extracted data is automatically enriched with response data (e.g. retrieval of twitter feeds if twitter username was extracted).

Output consists of semantically annotated data and semantic network according to system's ontology.

We show a high level implementation of training in Algorithm (1). Training for each task's classifier is done separately and in iterative manner to learn weights for intermediate labelings. At beginning of each iteration, feature function values are re-initialized and classifiers are re-learned. As we employ matching and merging technique during training execution, distant labelings from merged clusters become visible locally and therefore there is no need to use arbitrary structured CRF for NE and RE tasks. Labeling is done analogous to training algorithm using already built classifiers.

Algorithm 1 Collective IE Training

```
Input: $\overline{x}^k$, $\overline{y}^l$, maxIter
Output: classifiers (cNE, cREL, cCOREF)
Initialize coreferent clusters as $C = \emptyset$
$i \leftarrow 0$
while $i <$ maxIter and prevScoreDiff() $< \varepsilon$ do
    Initialize feature functions
    cNE $\leftarrow$ LCCRF($\overline{x}^k, \overline{y}^{NE}$)
    cREL $\leftarrow$ LCCRF($\overline{x}^k, \overline{y}^{REL}$)
    cCOREF $\leftarrow$ SCCRF($\overline{x}^k, \overline{y}^{COREF}$)
    $C \leftarrow$ matchingAndMerging($\overline{x}^k, \overline{y}^l$)
    $\overline{x}^{I\text{-}NE} \leftarrow$ cNE.tag($\overline{x}^k$)
    $\overline{x}^{I\text{-}REL} \leftarrow$ cREL.tag($\overline{x}^k$)
    $\overline{x}^{I\text{-}COREF} \leftarrow$ cCOREF.tag($\overline{x}^k$)
    $i \leftarrow i + 1$
end while
return (cNE, cREL, cCOREF)
```

The iterative algorithm is language independent. To support specific language, we need to have prebuilt bootstrap algorithms and labeled text corpus, following the representation described in section Section 4.1.

5 Current stage, Contribution and Future work

At the moment of writing this paper we have already developed matching and merging algorithm [14], based on attribute, relationship and semantic similarities. To show possible improvements, we have tested iterative method in combination NER and RE using Preprocessed and String features only. The results showed little improvement over pipeline approach and therefore we decided to continue building the proposed framework. Source code with some additional work is available in public repository[6].

The main contribution of the work is intelligent architecture for information extraction. It accepts unstructured text as an input and returns semantically tagged network as a result. Ontology is the only element that the user can change in order to impact the results. Furthermore, the core of the contribution will be the iterative method by new ways to learn and classify with iterative classifiers along with newly introduced feature functions. For Slovene language, the contribution will be the new annotated dataset and evaluation of proposed method on that data.

[6] `https://bitbucket.org/szitnik/iobie/`

Future work will focus on full framework implementation, evaluation on publicly available datasets and support at least one non-english language (e.g. Slovene).

6 Conclusion

In this paper we presented our in-progress PhD research. We proposed an intelligent collective ontology-based information extraction architecture, combining tasks of named entity recognition, relation extraction and coreference resolution.

A new parameterless IE architecture is based on an ontology which also updates at runtime. Main contributions are intelligent interconnection of main IE tasks, iterative training and labeling procedure, new possible feature functions and incorporation of matching and merging techniques into iterative method in order to use faster linear-chain instead of arbitrary CRFs. We also showed an example where proposed system could achieve significantly better results, mentioned little preliminary improvements over traditional approach and are continuously update publicly available sources.

Future work will include full implementation and evaluation of proposed iterative system on multi-language data.

7 Acknowledgments

The work has been supported by the Slovene Research Agency ARRS within the research program P2-0359 and part financed by the European Union, European Social Fund.

References

1. R. Bunescu and R. J. Mooney. Collective information extraction with relational markov networks. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.
2. J. Cai and M. Strube. End-to-end coreference resolution via hypergraph partitioning. In *Proceedings of the 23rd International Conference on Computational Linguistics*, page 143–151, 2010.
3. J. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, page 363–370, 2005.
4. J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.
5. J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

6. J. L. Leidner. Current issues in software engineering for natural language processing. In *Proceedings of the HLT-NAACL 2003 workshop on Software engineering and architecture of language technology systems-Volume 8*, page 45–50, 2003.
7. Y. Li, J. Jiang, H. Chieu, and K. Chai. Extracting relation descriptors with conditional random fields. pages 392–400, Thailand, 2011. Asian Federation of Natural Language Processing.
8. C. Nedellec and A. Nazarenko. Ontologies and information extraction. *CoRR*, abs/cs/0609137, 2006.
9. F. Niu, C. Zhang, C. Ré, and J. W. Shavlik. Felix: Scaling inference for markov logic with an operator-based approach. *CoRR*, abs/1108.0294, 2011.
10. L. Ramshaw and M. Marcus. Text chunking using transformation-based learning. In *Proceedings of the Third ACL Workshop on Very Large Corpora*, page 82–94, 1995.
11. S. Sarawagi. Information extraction. *Foundations and Trends in Databases*, 1(3):261–377, 2008.
12. M. L. Wick, A. Culotta, K. Rohanimanesh, and A. McCallum. An entity based model for coreference resolution. In *SDM*, pages 365–376, 2009.
13. D. C. Wimalasuriya and D. Dou. Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science*, 36(3):306–323, Mar. 2010.
14. S. Žitnik, L. Subelj, D. Lavbič, O. Vasilecas, and M. Bajec. Contextual data matching and mmrging using semantics, trust and ontologies. *Informatica - in review*, 2012.