Family Matters: Company Relations Extraction from Wikipedia

Artem Kuznetsov[1], Pavel Braslavski[1], Vladimir Ivanov[2]

[1] Ural Federal University
artkuznetsov.m@gmail.com, pbras@yandex.ru
[2] Innopolis University
v.ivanov@innopolis.ru

Abstract. The study described in the paper deals with the extraction of relations between organizations from the Russian Wikipedia. We experiment with two data sources for supervised methods – manual annotations made from scratch and relations from infoboxes with subsequent sentence matching, as well as different feature sets and learning methods – SVM, CRF, and UIMA Ruta. Results show that the automatically obtained training data delivers worse results than manually annotated data, but the former approach is promising due to its scalability. Evaluation of relations extracted from a subset of Wikipedia pages that are mapped to the Russian state company registry proves that external sources can enrich and complement official databases.

1 Introduction

Relation extraction (RE) between objects mentioned in text documents is an important area of information extraction. The task is not as well developed as named entity recognition (NER), which has independent significance, but is also a necessary preliminary step for RE.

RE research has made a significant progress since its advent in the 1990s; the development of the area during almost two decades can be tracked on the materials of two evaluation initiatives: MUC (1991–1997)[3] and ACE (2000–2008)[4].

The vast majority of RE research has been conducted on English data (see Section 2); there are only few studies on relation extraction for Russian. A pilot track on NER and fact extraction was organized by ROMIP in 2005[5]; however, participation was low. There has been no standard publicly available dataset suited for relation extraction task until recently. Open FactRuEval challenge[6] that has been conducted in spring 2016, partially solves this problem – organizers prepared and published a news corpus with labeled named entities (*persons* and *organizations*) and relations of four types (*commercial deal*, *meeting*, *person owning a company*, and *person employed in a company*).

[3] http://www.itl.nist.gov/iaui/894.02/related_projects/muc/
[4] https://www.ldc.upenn.edu/collaborations/past-projects/ace
[5] http://romip.ru/ru/2005/tracks/qa.html (in Russian)
[6] https://github.com/dialogue-evaluation/factRuEval-2016

Our study deals with extraction of binary hierarchical relations (parent/daughter company, ownership, founding, governance, etc.) between organizations of various kinds from the Russian Wikipedia. Wikipedia data allowed us, on the one hand, to skip the NER step, on the other – to experiment with automatically gathered data for training.

The goal of our study is twofold:

- to compare several widely used supervised approaches and different shallow features in the task of RE from Russian documents and
- to explore the potential of automatically collected data for training.

We have manually annotated 7,059 contexts with company mentions from 4,662 Wikipedia pages. We used this data for training and testing. Moreover, we collected 2,799 relations between 3,025 companies from Wikipedia infoboxes (either directly from Wikipedia dump or through DBpedia), then identified 6,962 sentences mentioning these companies. Hypothesizing that these text fragments represent relations encoded in the infoboxes, we used the data for training (obviously, this assumption does not always hold and the resulting data is essentially noisy, see discussion in Section 3). The manually annotated data created within the study is freely available for research purposes.[7]

We compared three methods of building RE classifiers: Support Vector Machines (SVM, a universal classification method used in many applications), Conditional Random Fields (CRF, a sequence classification method, a *de facto* standard for NER and RE tasks), as well as an automatic rules induction algorithm. We used a set of shallow classification features – mostly lexical and part-of-speech features – and their combinations within a window of variable size. Since we aimed at creating a baseline, we did not employed syntactic features and left this option for our future work.

Based on the evaluation results, we can conclude that a straightforward use of Wikipedia data for RE learning produces useful results ($macro\ F_1 = 57.4\%$ for two relations) at virtually zero annotation costs, but manually annotated data of higher quality provides about 20% gain in terms of F-score ($macro\ F_1 = 69.1\%$). Using a large set of shallow features does not affect the extraction quality significantly – almost identical results can be obtained using tokens and lemmata only. The quality of relation extraction increases with context length for feature calculation and reaches a plateau at window size of nine words.

At the final stage, we estimated how the relations automatically mined from Wikipedia can supplement the existing official databases. To do this, we automatically extracted ownership relations between companies from about 6K Wikipedia pages mapped to the Russian registry of legal entities[8]. A comparison of the extracted relations and those from the registry shows that the proposed method can complement and enrich existing structured data sources.

[7] `https://github.com/kriskk/OrganizationRelationRecognition`
[8] `https://egrul.nalog.ru/` (in Russian)

2 Related Work

Relation extraction tasks and methods differ from each other in terms of the type of information to be extracted. Some of the recent works [10, 16] are aimed at extraction of a particular relation type between two classes of an ontology. In this case, no instances of the relation are extracted. Other works [15, 18] focus on extraction of instances of a particular relation type. In this case RE learning requires significant manual efforts, which leads to low scalability. Other approaches [1] propose methods that extract related pairs of concepts taking into account only strength of the relation, without considering its type.

Existing English corpora for relation extraction have been manually created during a series of shared tasks and evaluation initiatives [3]. Such text corpora are crucial for evaluation of extraction methods, however they will never be sufficient for all application domains. Thus, an important part of most relation extraction methods is the approach to training data acquisition and construction. There are four directions: supervised approaches, unsupervised approaches [4, 21], semi-supervised methods [5], and distant supervision [12], or self-supervised learning. In the past decade Wikipedia was intensively used in RE studies. Semi-supervised and distant supervision approaches are most relevant in the context of our work.

In [19] a bootstrapping semi-supervised method was proposed for "Semantifying Wikipedia" and identified Wikipedia link structure, taxonomies, infoboxes, etc. as useful data for self-supervised semantic enrichment. Their system KYLIN is based on a CRF extractor trained on a set of lexical features. The system uses concepts' mentions represented in a Wikipedia page as hyperlinks. The main purpose of the method was to fill infobox fields. A very similar approach is proposed in [9]. It also uses CRF and achieves precision of 91% for the task of infobox attributes population. However, the performance was measured on all types of attributes, not just on relations between two entities.

A distant supervision approach to relation extraction was proposed by Mintz et al. [13] and provides a powerful idea to build a training set for relation extraction. The authors claimed that syntax-level features are important for relation extraction. Authors constructed a training set consisting of 800,000 pages and 900,000 relation instances from Freebase. The distant supervision means that any sentence with a pair of entities that participate in some known relation is likely to express that relation. The idea is very similar to our approach, but there are differences. First, we extract relations between page title entity and an entity mentioned in the page body. Second, our approach works with predefined types or classes of relations, and does not consider particular instances of relations.

One implicit assumption of distant supervision is that the reference database is complete. Apparently, it cannot be true in practice and leads to a high number of false negative training examples. Min et al. [12] extended the idea and proposed the Multiple-Instance Multiple Label algorithm that learns (from positive and unlabeled data) and tested the algorithm on Wikipedia.

Recent works on distant supervision usually consider web-scale relation extraction and use the Linked Open Data cloud as a source of relations instances.

[2] describes an approach to an improved distant supervision approach, where statistical techniques help to strategically select training seeds with lesser lexical ambiguity. Authors propose the following relaxation to the "one sentence – one relation" assumption: "If two entities participate in a relation, any paragraph that contains those two entities might express that relation, even if not in the same sentence, provided that another sentence in the paragraph in itself contains a relationship for the same subject" [2].

When a training dataset is provided, one should employ an appropriate machine learning method for relation extraction. Conditional Random Fields [8] and SVM [11, 6] are widely used for relation extraction tasks. A comprehensive survey of relation extraction methods can be found in [14, 7].

3 Data

In our work we use articles about organizations and companies from the Russian Wikipedia[9]. Using Wikipedia data for relation extraction allows us to skip the NER step – we consider only relations between the title company (the company the article is about) and companies mentions in the page body that are marked as anchor text of outlinks to other companies' pages. To label relations in the text of the page, we employed two approaches: 1) manual annotation and 2) automatic extraction based on information presented in Wikipedia infoboxes. Automatically labeled data and a subset of manually annotated data are used for training; both approaches are tested on the held-out 'manual' data. Fig. 1 shows an example of a Wikipedia page and relations labeled on the data preparation stage. In addition, we conducted a small experiment to find out how the relations extracted from Wikipedia pages correspond to the information presented in the official databases. We employed the JWPL library[10] for Wikipedia data processing.

3.1 Manual Annotation

To select Wikipedia pages for manual labeling, we compiled a wordlist of different organization types – *company, organization, holding, bank, factory*, etc. After that, we mined a list of Wikipedia categories containing these words and collected all the pages in these categories. Then, we selected only those pages that have links to other pages in the set; the final collection contained 10,512 Wikipedia pages.

The basic unit for annotation was a sentence containing inter-company links. The annotator was presented with the sentence and its context (±300 characters around the link) within a section of the wiki-page. The majority of sentences came from summary sections or from sections about companies' history.

[9] `https://ru.wikipedia.org/`
[10] `https://dkpro.github.io/dkpro-jwpl/`

We used *brat* tool[11] for manual annotation. The annotator's task was to link the highlighted organization to the organization in the title by one of the three relation types – *Holder*, *Subsidiary*, or *Other*. Since the annotators labeled only relations between the 'main' company and already highlighted other companies' mentions, it greatly simplified and speeded up the annotation process. The instruction required that the relation was expressed within a single sentence. For example, if a context contained a relation that required anaphora resolution, annotator was not supposed to set a link. Due to limited resources the whole annotation was performed by two annotators without overlap. This resulted in 7,154 annotated contexts in total, in particular 2,150 *Holder* relations, 992 – *Subsidiary*, and 4,012 *Other*.

3.2 Automatic Labeling

The second data source about company relations is infoboxes that represent important facts about the page subject in a structured way. As Wikipedia editor's guide states, infoboxes "are not 'statistics' tables in that they ... only summarize material from an article – the information should still be present in the main text".[12]

Similarly to the approach described in [20], we extracted ownership relations from infoboxes and then searched textual representations of them in the article body. We compiled a list of infobox fields that reflect ownership or governance relations and extracted 1,922 company pairs. Moreover, we extracted standardized company relations (*rel-parentCompany-ru, rel-owningCompany-ru, rel-parentOrganisation-ru*) from DBpedia[13], which resulted in 1,780 additional relations. The surplus is mainly due to relations presented in English pages' infoboxes that can be 'transferred' to their parallel Russian pages. After duplicates removal and normalization (inverting *Subsidiary* relations to *Holder*) we obtained 2,799 relations.

In the next step we extracted textual contexts presumably reflecting the infobox relationships. For each company in the relation we searched for an exact match of its counterpart on the corresponding page. For example, for the relation $X is\_Holder\_of Y$ we searched for mentions of Y on page X (and considered that the sought sentence expressed the *Subsidiary* relation) and vice versa – X's mentions on Y's page (assuming that these mentions expressed the *Holder* relation).

In addition, we required that the sentence was at least 30 characters long as a simple criterion for natural language sentences. We also sampled sentences with company mentions that were not members of infobox relations and regarded them as manifestations of *Other* relations (we needed them as negative class instances when training classifiers). After duplicates removal we got 6,471 contexts: 3,840 – *Holder*, 979 – *Subsidiary*, and 1,652 – *Other*.

[11] `http://brat.nlplab.org/`
[12] `https://en.wikipedia.org/wiki/Help:Infobox`
[13] `http://wiki.dbpedia.org/`

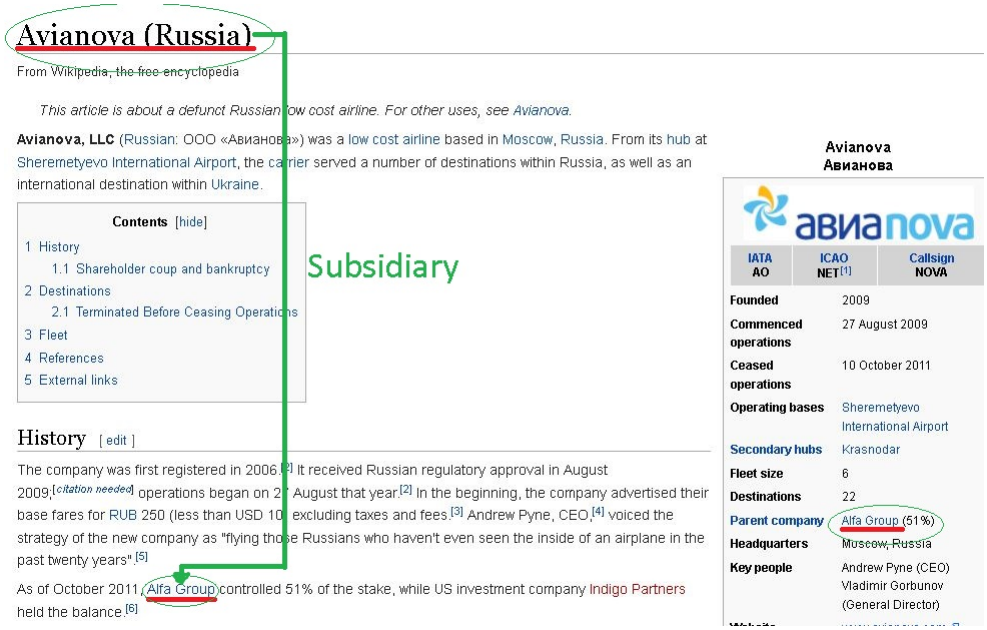

Fig. 1: Example: relation representation in the page body and infobox.

For example, infobox of the *TNK-BP* page indicates *Rosneft* as *Holder* and the page itself contains the following sentence:

> *At the end of October 2012, Rosneft has announced the acquisition of its competitor — TNK-BP oil company.*

Obviously, such automatic approach produces noisy data, for example infobox on the *Beltelecom* page mentions *Government of Belarus* as a holder, but the extracted sentence does not reflect this relation:

> *Sergei Popkov, the ex-head of Beltelecom, was appointed as Minister of Communications and Information Technology instead of Nikolai Pantelei.*

The 'manual' dataset (4,327 organizations) and 'automatic' one (3,004) have 970 entries in common. Out of 2,799 relations in 'automatic' and 2,383 in 'manual' datasets, 477 relations are presented in both. This comparison illustrates that the two approaches to data acquisition complement each other.

3.3 State Registry of Legal Entities

One of the goals of the study was to figure out to which extent the information from Wikipedia can enrich existing official databases. To this end, we used a set of 6,206 Wikipedia pages about companies that were automatically matched with records in the Russian registry of legal entities. The registry contains basic

information about companies and organizations, including data about founders and owners.

4 Relation Extraction Learning

As we stated earlier, Wikipedia data allows us to simplify the task of relations extraction and skip the NER step. We cast the relation extraction problem as classification into three classes: *Holder/Subsidiary/Other*.

4.1 Classification Methods

Linear SVM. Support Vector Machines (SVM) showed their utility in a wide variety of tasks [6]. We treat linear SVM with bag-of-words features as a baseline in our experiments. Binary feature vectors are obtained based on the 12-word-long context around the company mention.[14] We used *scikit-learn* implementation of linear SVM.[15]

Conditional Random Fields. Sequence classifiers that take into account linear sentence structure proved to be very efficient in natural language processing, in particular – in information extraction tasks. Conditional random fields (CRF) is a sequential algorithm that became a *de facto* standard for NER and RE tasks. It treats a sentence as a sequence of chunks and marks each chunk with a class label (with additional 'None' label). We used the CRFSharp implementation[16] in our experiments. CRF allows accounting both for the left and right contexts of the current token and thus introduces window size as an additional parameter. We consider symmetric windows of size $2 \cdot x + 1$, i.e. x tokens on each side of the current token. We used a much richer feature set in case of CRF (see section 4.2).

Rule induction. An alternative approach to relation extraction is example-based rule induction. We took advantage of implementation of the WHISK algorithm [17] in Apache UIMA Ruta[17]. Rule generation algorithm is implemented as "TextRuler" plug-in for the Eclipse environment [18]. Unfortunately, we encountered performance issues when applying this algorithm to our data. As a

[14] We also performed experiments with lemmatized contexts, however, it did not affected classification accuracy.

[15] `http://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html`. We also experimented with other classifiers from the same library – MultinomialNB, BernoulliNB, RidgeClassifier, Perceptron, PassiveAggressiveClassifier, KNeighborsClassifier, SGDClassifier, NearestCentroid. They produced very similar results to those by SVM.

[16] `https://crfsharp.codeplex.com/`

[17] `https://uima.apache.org/ruta.html`

[18] `https://eclipse.org/`

workaround we splitted the training set into chunks of approximately 600 contexts each; the runtime for each chunk constituted about two hours.[19] We used morphological tags (see section 4.2) as features on par with TextRuler internal labels.

UIMA Ruta produces rules of the following form (p and n indicate the number of positive and negatives outputs when applied to training set, respectively):

Org{→ MARKONCE(Holder)} Sush # SW; // p=6; n=0
Org{→MARKONCE(Subsidiary)} SPECIAL COMMA # TokenAnnotation SPECIAL; // p=7; n=0

4.2 CRF classification features

Each token in CRF method is described with the following features:

- *Token:* any alphanumeric sequence;
- *Lemma:* output of *mystem* morphological analyzer[20] in contextual disambiguation mode;
- *Script:* marks Cyrillic/Latin/Special symbols (punctuation marks and digits);
- *Part-of-speech (POS):* Due to rich Russian morphology, the standard approach is to encode part-of-speech tags (*noun, verb, adjective, adverb,* etc.) separately from the grammar tags (*number, case, animacy, person, tense, aspect*, etc.). This feature corresponds to the former notion, i.e. the very POS tags.
- *Grammar tags: mystem* output – 52 tags (*gender, case, tense,* etc.), each is a binary feature;
- *IsOrganization:* this is a binary feature that marks the organization mention – the potential relation member based on Wikipedia markup;
- *Feature bigrams:* combinations such as (*token*, *lemma*) and (*lemma*, *POS*);
- *Dictionary features:* based on a list of words that occurred frequently near company names in the training set plus synonyms and different organization names (e.g. *factory, corporation, bank*, etc.).

5 Results and Discussion

We splitted the manually labeled dataset into train (70%) and test (30%) sets, the latter was used for evaluation of all approaches. Table 1 shows the evaluation results (the cited CRF results correspond to all features and window of size 13). The table indicates quality measures for target classes only (i.e. evaluation results for *Other* class are not shown). It can be seen from the table that rule

[19] It took about a week to process the complete dataset on a commodity desktop machine. However, it resuled in much lower quality in comparison to the *divide and conquer* approach – *macro* $F_1 = 37.9$ vs. 50.2.

[20] `https://tech.yandex.ru/mystem/`

induction can deliver perfect precision, but a very low recall in case of *Holder* class. At the same time, rule induction is quite robust to the noise in the data and performs equally both with manual and automatic training data. Rule induction also delivers best recall for *Subsidiary* class and is on par with CRF in terms of general performance ($F1$) when the methods are trained on automatically gathered data. As expected, CRF outperforms other approaches, when trained on manual data.

Method	Training set	Holder			Subsidiary			macro F1
		P	R	F1	P	R	F1	
Linear SVM	Manual	66.8	63.4	65.0	55.0	28.4	37.4	51.2
CRF	Manual	82.3	**75.7**	**78.8**	72.9	**50.1**	59.4	**69.1**
UIMA Ruta	Manual	**100.0**	25.7	40.8	**100.0**	42.3	59.5	50.2
Linear SVM	Automatic	47.0	41.4	44.0	26.0	18.1	21.4	32.7
CRF	Automatic	56.2	65.6	60.5	41.9	37.9	39.8	50.2
UIMA Ruta	Automatic	**100.0**	25.4	40.5	**100.0**	42.6	**59.8**	50.2

Table 1: Evaluation results for different learning methods and training sets (in percents).

Fig. 2 illustrates that window size positively impacts F1-score that reaches a plateau at context length of nine words (the shown results are obtained without grammar features due to efficiency reasons).

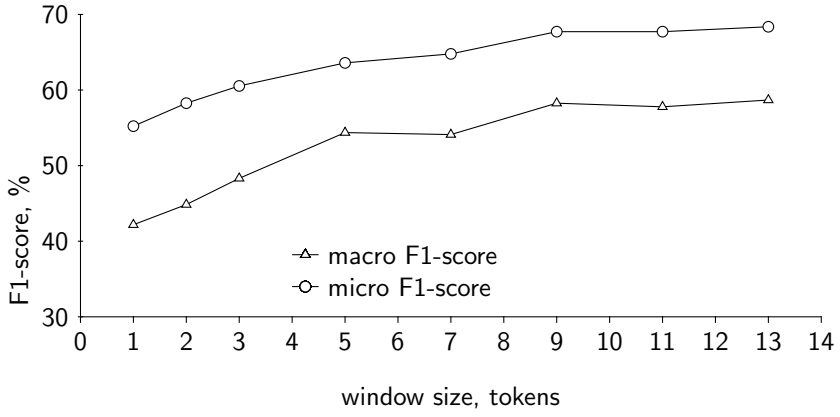

Fig. 2: Impact of context size on relation extraction quality.

Contribution of different features can be estimated based on Table 2. The table shows extraction results for CRF trained on manually obtained data with the window of nine words. The results allow us to conclude that the same extraction

quality can be achieved with tokens and lemmata as features only; richer linguistic features such as POS and grammar features do not influence the resulting quality significantly.

Feature set	F1, %
all features	67.8
w/o grammar tags	67.7
w/o dictionary	68.2
w/o bigrams	68.0
w/o script	67.3
w/o POS	67.7
tokens only	67.4
tokens and lemmata	67.5

Table 2: Contribution of different features to overall relation extraction quality.

At the final stage of our experiment we addressed the question, to what extent the automatically extracted relations can enrich the existing databases. To this end, we extracted relations from 6,206 Wikipedia pages that were automatically matched to the records of the Russian state registry of legal entities. We juxtapose the following three sets of relations: 1) Wikipedia + DBpedia — relations from infoboxes and DBpedia; 2) automatically extracted relations from Wikipeda articles; 3) relations from the registry. Fig. 3 illustrates the overlap between these three sets. The results show that Wikipedia can enrich and complement existing official databases. News can be potentially even more valuable and dynamic source for relation extraction between companies.

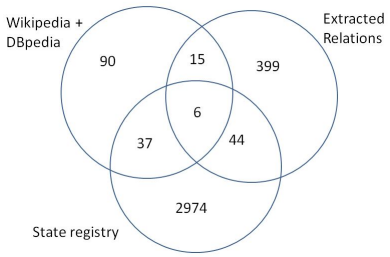

Fig. 3: Intersection of relations between 6,206 organizations from three different sources.

6 Conclusion

We conducted a pilot study aimed at extracting relations between companies from the Russian Wikipedia. We manually labeled a sizable dataset of sentences with *Holder/Subsidiary* relations and made it freely available for research purposes. We hope that these efforts will promote RE research on Russian language data.

We compared several supervised approaches to relations extraction – SVM and CRF with shallow features, as well as automatic rule generation. We also investigated automatic mining of labeled examples from Wikipedia. Rule induction, though computationally less effective, showed high precision results even when trained on noisy data. Although the automatically gathered training set was able to deliver decent results, the more elaborated manual dataset allowed for a better quality. Sequential method (CRF) outperformed SVM with bag-of-words features as expected. A wide variety of shallow features did not lead to improved results – the same quality was achieved with tokens and lemmata only as features. Rule induction, though computationally less effective, delivered high precision results even when trained on noisy data. Thus, we established several baselines for relation extraction methods from Russian documents.

Despite its size (more than 1.3 million articles), Russian Wikipedia contains relatively little information about companies and organizations, especially when compared to news stream and focusing on lesser-known organizations. In our future work we plan to transfer our methods to news data – it will include the NER step and switching from inherently unary relations to actual binary ones. We also plan to investigate the contribution of syntactic features to relation extraction for Russian.

7 Acknowledgements

The study was supported by RFBR grants #14-37-50950 "Research and development of algorithms for relation extraction from Wikipedia texts" and #15-29-01173 "Computational models and mathematical methods for big data analysis of trends and correlations in society". We want to thank Ksenia Zhagorina for providing us with Wikipedia pages matched to the state registry of legal entities, as well as Kontur.Focus[21] for the very registry data. Last, but not least we thank Olga Rogacheva and Maria Belkova for annotating the data.

References

1. Astrakhantsev, N., Fedorenko, D., Turdakov, D.: Automatic enrichment of informal ontology by analyzing a domain-specific text collection. In: Proceedings of Dialog. pp. 29–42 (2014)
2. Augenstein, I., Maynard, D., Ciravegna, F.: Relation extraction from the web using distant supervision. In: Proceedings of EKAW. pp. 26–41 (2014)

[21] https://focus.kontur.ru/

3. Doddington, G.R., Mitchell, A., Przybocki, M.A., Ramshaw, L.A., Strassel, S., Weischedel, R.M.: The automatic content extraction (ace) program – tasks, data, and evaluation. In: Proceedings of LREC (2004)
4. Etzioni, O., Banko, M., Soderland, S., Weld, D.S.: Open information extraction from the web. CACM 51(12), 68–74 (2008)
5. Etzioni, O., et al.: Web-scale information extraction in knowitall:(preliminary results). In: Proceedings of WWW. pp. 100–110 (2004)
6. Joachims, T.: Text categorization with suport vector machines: Learning with many relevant features. In: Proceedings of ECML. pp. 137–142 (1998)
7. Konstantinova, N.: Review of relation extraction methods: What is new out there? In: Proceedings of AIST. pp. 15–28 (2014)
8. Lafferty, J.D., McCallum, A., Pereira, F.C.N.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proceedings of ICML. pp. 282–289 (2001)
9. Lange, D., Böhm, C., Naumann, F.: Extracting structured information from wikipedia articles to populate infoboxes. In: Proceedings of CIKM. pp. 1661–1664 (2010)
10. Maedche, A., Staab, S.: Discovering conceptual relations from text. In: Proceedings of ECAI. pp. 321–325 (2000)
11. McNamee, P., Mayfield, J.: Entity extraction without language-specific resources. In: Proceedings CoNLL (2002)
12. Min, B., Grishman, R., Wan, L., Wang, C., Gondek, D.: Distant supervision for relation extraction with an incomplete knowledge base. In: Proceedings of HLT-NAACL. pp. 777–782 (2013)
13. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of ACL. pp. 1003–1011 (2009)
14. Nastase, V., Nakov, P., Seaghdha, D.O., Szpakowicz, S.: Semantic relations between nominals. Synthesis Lectures on Human Language Technologies 6(1), 1–119 (2013)
15. Sarawagi, S.: Information extraction. Foundations and trends in databases 1(3), 261–377 (2008)
16. Schutz, A., Buitelaar, P.: Relext: A tool for relation extraction from text in ontology extension. In: Proceedings of ISWC. pp. 593–606 (2005)
17. Soderland, S.: Learning information extraction rules for semi-structured and free text. Machine learning 34(1-3), 233–272 (1999)
18. Weikum, G., Theobald, M.: From information to knowledge: harvesting entities and relationships from web sources. In: Proceedings of PODS. pp. 65–76 (2010)
19. Wu, F., Weld, D.S.: Autonomously semantifying wikipedia. In: Proceedings of CIKM. pp. 41–50 (2007)
20. Wu, F., Weld, D.S.: Open information extraction using wikipedia. In: Proceedings of ACL. pp. 118–127 (2010)
21. Yao, L., Haghighi, A., Riedel, S., McCallum, A.: Structured relation discovery using generative models. In: Proceedings of EMNLP. pp. 1456–1466 (2011)