Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2016"

Moscow, June 1–4, 2016

NAMED ENTITY NORMALIZATION FOR FACT EXTRACTION TASK

Popov A. M. (hedgeonline@gmail.com)[1],
Adaskina Yu. V. (adaskina@gmail.com)[2],
Andreyeva D. A. (andreyeva.pavlodar@gmail.com)[1],
Charabet Ja. K. (jkharabet@gmail.com)[1],
Moskvina A. D. (moskvina.anya@gmail.com)[1],
Protopopova E. V. (protoev@gmail.com)[1],
Yushina T. A. (iushina.tatiana@gmail.com)[1]

[1]St. Petersburg State University, St. Petersburg, Russia
[2]InfoQubes, Moscow, Russia

The paper describes our approach to the task of information extraction within FactRuEval, an independent evaluation of Named Entity Recognition and Fact Extraction tools. We took part in the three subtasks of the evaluation: Named Entity Recognition per se, Entity Normalization and Fact Extraction.

We chose a rule-based approach to the task. The three subtasks correspond to the modules of 'Hurma' parser, the tool we have developed. In addition to traditional lexicon and regular expressions based rules, it allows creating elaborate rules to mine and normalize different kinds of entities with regard to specific challenges such language as Russian presents to the researchers. For Fact Extraction, we used skip-gram based algorithm with no dependencies in order to overcome the problem of data sparsity.

Preliminary results show that our Entity Extraction and Normalization methods score reasonably high and our Fact Extraction score is high enough, taken into account that that our expected maximum F-measure is relatively low due to the specifics of the Gold Standard.

Key words: Information Extraction, Named Entity Recognition, Named Entity Normalization, Fact Extraction, skip-grams

ОСОБЕННОСТИ НОРМАЛИЗАЦИИ ИМЕНОВАННЫХ СУЩНОСТЕЙ ДЛЯ ЗАДАЧИ ИЗВЛЕЧЕНИЯ ФАКТОВ

Попов А. М. (hedgeonline@gmail.com)[1],
Адаскина Ю. В. (adaskina@gmail.com)[2],
Андреева Д. А. (andreyeva.pavlodar@gmail.com)[1],
Москвина А. Д. (moskvina.anya@gmail.com)[1],
Протопопова Е. В. (protoev@gmail.com)[1],
Харабет Я. К. (jkharabet@gmail.com)[1],
Юшина Т. А. (iushina.tatiana@gmail.com)[1]

[1]СПбГУ, Санкт-Петербург, Россия
[2]InfoQubes, Москва, Россия

Ключевые слова: извлечение информации, извлечение именованных сущностей, извлечение фактов, нормализация именованных сущностей

1. Introduction

Named Entity Recognition (NER) is a crucial task for most NPL platforms, both commercially and academically oriented. There has been a lot of research on rule-based and statistical approaches to this problem. Within the task, it is often necessary to perform so called Named Entity Normalization (NEN). One can define Normalization as finding a standardized name form for an entity or attributing it to an identifier. We can define Entity Normalization in the narrow sense as a task of setting grammatical forms of its constituents to a certain “normal form”, so that the whole entity is transformed into a dictionary entry, and thus can be used as such. NEN is vital for morphologically rich languages, such as Russian, as grammatical forms of individual constituents can be changed significantly and their morphological lemma is often not the desirable “normal” entry. Usually those “normalized” entities are used for Fact Extraction—another important subtask of Information Extraction. Facts are templates representing a certain type of situations, and the facts’ fields correspond to the situation attributes, such as its participants, time, place, etc. As in case of any other challenge of Natural Language Processing, evaluation is a ubiquitous methodological problem. FactRuEval is an open competition for the Russian language platforms dealing with all three tasks mentioned above: Named Entity Recognition, Named Entity Normalization and Fact Extraction.

The aim of this paper is twofold. Firstly, we describe our take on the tasks of FactRuEval, which is the use of a multi-module system employing a rule-based approach. We believe that our method can yield a high precision without extensive dictionaries of such entities as organizations and locations. Secondly, we discuss in detail our approach to the task of Entity Normalization, a very challenging task for Russian.

The paper is organized as follows. Section 2 describes prior work on general methods of NLP platforms evaluation. Chapter 3 provides an outline of our parser. In Chapter 4 and 5 we discuss our approaches to the evaluation tasks, with special attention to Normalization. Relevant background research is briefly discussed in each chapter. Evaluation results are presented in Chapter 6, while Section 7 provides a conclusive discussion along with ideas for future work.

2. Related Work on Evaluation

Named entity recognition, linking and Fact Extraction has been widely studied during last 15 years thanks to numerous competitions such as held by MUC (Message Understanding Conference). The results for several European languages (English, German etc.) are presented in (Grishman, Sundheim 1996), among others.

However, the task applied to Russian language lacks thorough research especially concerning the issue of normalization. ROMIP evaluation campaign organized in 2005 (ROMIP 2005) included NER and Fact Extraction tracks. The majority of participants used ontology or rule-based approaches. For instance, RCO system used entity dictionaries and syntactic rules as well as a coreference resolution algorithm (Ermakov 2005) based on partial coincidence or known synonymic relations. Fact Extraction systems are based either on the results of syntactic parsing (Kiselev 2004) or on shallow morphosyntactic patterns (see Gareev et al. 2013).

Several more recent results incorporate anaphoric processing (Ermakov 2007) or rule-based approach to extracting implicit factual information (Kuznetsov 2012). The Named Entity Recognition task for Russian has not been widely discussed recently, with (Gareev et al. 2013) as a notable exception, where the authors introduce baselines for rule-based and statistical Russian NER.

3. ‘Hurma’ Parser Description

Our text processing system, ‘Hurma’, represents a classic pipeline of level-by-level analysis of natural language. It consists of a number of modules, which are applied in a sequential manner and produce relevant output. Our system is implemented in Java; currently it consists of the following modules:

- Tokenizer;
- Morphological analyzer;
- Gazetteer;
- Pattern search engine;
- Fact extractor.

Tokenizer is a regular expression based module, which provides means for tokenization and segmentation of input paragraph and for classifying tokens. Tokenization rules are defined in a tabular form, where the token-extraction pattern is paired with a token-type tag. Tokenization and segmentation can be performed in a number of different ways (Dale et al. 2000). Each rule assigns some tags to extracted tokens, for example, we tag tokens that are end-of-sentence markers: they are later used for sentence segmentation.

Morphological analyzer module is used for attributing each wordform a set of pairs of normal form and corresponding morphological tag via a dictionary or, in case of an unknown word, for predicting possible normal forms and tags. Our morphological dictionary is represented as Minimal Finite State Automaton and is compiled from a set of wordfrom files. It is a straightforward implementation of the algorithm described in (Daciuk et al. 2000), which allows us to reduce memory consumption and maintain high performance. Our dictionary is imported mainly from the Dialing project (Sokirko 2001) augmented with extra names and organizations from the Open Corpora project[1].

Gazetteer module is used for tagging words and phrases with semantic and morphological tags by means of various dictionaries: including exact forms and normal forms dictionaries, as well as regular expression patterns, word sequences and synonym sets ones.

Pattern search engine module is the main tool for entity extraction. We follow a rule-based approach for named entity recognition (Gareev et al. 2013). Our module is designed to execute regular-expression-like rules, which are described as token sequences and can be restricted either by lexical, grammatical or semantic constraints (such token sequences are usually called “N-grams”). Elements of such rules can be matched with the tokens of the sentence as one-to-one, one-to-many or as optional elements, which may not be present in the sequence. Only non-intersecting entities are permitted, so if some entities intersect, the most appropriate one is selected using manually set rule priority and entity boundaries information. Each rule provides full control of the normalization process of each N-gram element: which wordform should the element be normalized to and which named field should the element be assigned to (i.e. we can assign labels to tokens, e.g. we can separate organization type form the name).

Fact extractor module uses rules very similar to *pattern search engine*: we can describe a set of linearly ordered tokens in terms of lexical and grammatical restrictions with additional range constraints, those sets are often referred to as “skip-grams”. Skip-grams are used for various tasks of language processing and language modeling, but perhaps their most famous application is word2vec, where they are one of the methods of learning vector representations of words, introduced in (Mikolov 2013). The research proves that they outperform standard N-grams for many tasks; moreover, they can be helpful in overcoming the problem of data sparsity. Each element of such skip-gram can be assigned to a named field that corresponds to the fact field, that we are about to extract. One skip-gram element can be marked as “head” and any element can be marked as “restricted”; this means that if more than one fact with the same head is extracted, or more than one fact of the same type with intersecting restricted tokens is extracted, only one of them will remain, and all others will be discarded. The fact discarding process is introduced for precision control and it utilizes several heuristics to determine which fact should be left out. The priority parameter also contributes to the process of multi-rule case scenario resolution.

[1] Available at: http://opencorpora.org/dict.php

4. Normalization Techniques

The first task in normalizing an entity is extracting one. The core idea behind the proper normalization process is bringing up correct and relevant extraction patterns. Different entities can be clustered into groups according to how their constituents should be normalized, so when introducing a new extraction rule the operator must already know how exactly each element of described pattern should be normalized. Thus, all we need to do is to describe each element in terms of normalization directives. We have the following set of directives implemented:

- Default—word is normalized to the dictionary lemma (i.e. verbs, participles, etc. are normalized to the infinitive, nouns and adjectives are normalized to nominative singular etc.);
- Self—word is left in the exact form it occurred in text;
- Explicit—we can describe a set of grammemes of the desired wordform;
- Implicit—we can describe the target grammemes of the desired wordform as a grammatical agreement between specific words in the phrase;
- Conditional—we can add some grammemes to the desired wordform if and only if these grammemes are present in the tag of the source wordform in the text.

It turns out, that this set of five directives is enough to normalize nearly any wordform, because those explicit, implicit and conditional directives can be combined in various ways. In the following sections we will see how we use this apparatus to compose rules.

4.1. Locations

Locations, as almost any type of named entities, can be divided into two major groups:

- Single-word entities, such as “Россия”, “Москва”, “Нева” etc., which have no context to define other normalization techniques other than default, i.e. the normal form of such entities always will be the morphological lemma of its single constituent;
- Multiword entities, such as “Российская Федерация”, “Московская область”, “Соединенные Штаты Америки” etc., which have such a context and should be normalized according to some rules.

First, let us demonstrate the long way of the default normalization process of such entities to a morphological lemma:

- Российская Федерация → Российский Федерация;
- Московская область → Московский область;
- Соединенные Штаты Америки → Соединить Штат Америка.

We can see that in the first two examples the adjective is normalized to the default lemma, which has masculine gender with disregard of the noun gender. The third example shows us that all three words should remain in plural, the first word “Соединенные” should remain participle and the third word should remain in genitive. Furthermore, we can say that the third word never changes at all, no matter what grammatical form the whole entity is. Therefore, we can define the conditional directive “Participle” on the first word and the “Self” directive on the last. As for the first two examples, we can use an Implicit “Gender” directive, so that the adjective in both locations is left in agreement with the noun.

4.2. Persons

Persons again can be divided into two groups by the number of constituents:

- Single name or surname—in general there can be names that suit both masculine and feminine gender, moreover, all surnames (excluding invariable) have different forms in masculine and feminine gender in Russian, this means that we cannot always know for sure, how to normalize a single-word person entity;
- Multiword persons—sometimes the ambiguity can only be resolved if the patronymic is present—in Russian the patronymic forms are always different in masculine and feminine genders.

So, given the level of gender ambiguity in Russian, the optimal choice appears to be the following: normalize and preserve all possible combinations of normal forms for persons until the ambiguity is resolved in the future. For example, the person “Александра Иванова” can be normalized in two different ways:

- “Александр Иванов”—masculine, accusative, singular;
- “Александра Иванова”—feminine, nominative, singular.

4.3. Organizations

The organizations named entity type is probably the most challenging type both for recognition and for normalization. The first issue is that often there are no strict boundaries, so extractable entities can be quite long, decreasing the chance of correct normalization. Another problem is that organizations’ names are often not present in the morphological dictionary but still can be morphologically inclined, so this offers additional challenge for recognition and normalization systems. Let us consider some examples; again, demonstrating the difference between default and controllable normalization techniques:

- Агентству военных новостей—Агентство военный новость / Агентство военных новостей
- Высшей национальной школе изящных искусств—Высокий национальный школа изящное искусство / Высшая национальная школа изящных искусств

The latter case is perhaps the most challenging, since it requires nearly all available directives to be used in a single pattern:

Table 1. Normalization example

Source	Высшей	национальной	школе	изящных	искусств
Default	Высокий	национальный	школа	изящный	искусство
Controllable	Высшая	национальная	школа	изящных	искусств
Directives	highest?	agr(gender, 3)	Lemma	Self	Self

Where:

- "highest?" denotes a conditional on grammeme "highest" (specially introduced for such cases where the degree of comparison should be preserved);
- "agr(gender, 3)" denotes implicit normalization to the common gender of agreeing words (the current and the third, i.e. "школа");
- "Lemma" denotes normalization to a dictionary form;
- "Self" denotes leaving the current form as it is.

5. Fact Extraction

Fact Extraction is a term commonly referring to a subtask of Information Extraction where the output unit is 'a fact', i.e. the representation of a certain type of situation and its participants. Another term would be template relation, or relations among entities, see (Cunningham 2005). So, a fact is a template structure that is filled in during the process of text analysis, the fields of the template are determined by the fact type. The classic example of the fact is Deal, the set of fields including Seller, Buyer, Object, Place and Time. The Fact Extraction algorithm could parse the text *I bought some donuts at the Dunkin Donut's store on Roosevelt Boulevard yesterday* and extract the fact Deal with the following attributes: Seller (Dunkin Donut's), Buyer (I), Object (Donuts), Place (Roosevelt Boulevard), Time (Yesterday).

Today's methods of Fact Extraction include ontology-based approaches, rule-based approaches and machine learning methods, sometimes the latter is combined with either of the former. Rules or patterns for information extraction can be handwritten or automatically learned by a system. Either way, the rules can contain only word tags (Hearst 1992), or include dependencies (Yangaber 2002, Akbik 2013), i.e. syntactic information.

Following (Hearst 1992) among others, we used a manual dependency-free approach based on a skip-gram model for the task of Fact Extraction. Skip-grams are a technique, which, as opposed to traditional N-grams, allows for non-adjacent word sequences, that is, some nodes are 'skipped'. The Fact Extraction module has access to all the information acquired from the morphological analyzer, the gazetteers and NER modules. It parses skip-gram sequences, where the nodes are referred to by their morphological and/or semantic tags. Alongside the general semantics, we used special lexical classes for each fact type, that is nouns and verbs denoting purchases, company merges, employment etc.

6. Evaluation

We evaluate our system within the FactRuEval-2016 competition, participating in all three tasks:

1. Named Entity Recognition—extract the boundaries of the entities and define their type;
2. Named Entity Normalization—bring entities extracted in the previous step to one of the possible normal forms;
3. Fact Extraction—extract facts and fill in their fields with normalized names of entities extracted in the previous task and define the fact type.

The dataset consisted of two subsets: a development set (122 documents, approx. 31K tokens) and a test set (132 documents, approx. 59K tokens); the entity and fact types were distributed as follows:

Table 2. Dataset statistics

		Test set		**Development set**	
Type	**Subtype**	**Count**	**%**	**Count**	**%**
Named entity	Persons	1,347	32	728	31
	Organizations	1,537	37	661	28
	Locations	1,283	31	943	41
Fact	Ownership	141	23	17	7
	Occupation	336	54	180	78
	Meeting	45	7	5	2
	Deal	102	16	29	13

Table 2 illustrates that the development set is quite unbalanced, especially as for the facts section, and that results in significant quality drop within the third task.

The first task is a good test for our pattern-based NER system in general, if sufficient quality is achieved in this task, so the extracted entities can be further normalized. We evaluate our system by means of the comparators and the Gold Standard provided by the competition organizers:

Table 3. Preliminary results for Entity Extraction task

Entity type	**Precision**	**Recall**	**F-measure**
Persons	0.9300	0.8403	0.8829
Locations	0.9535	0.8361	0.8910
Organizations	0.8181	0.5450	0.6542
OVERALL	0.9038	0.7301	0.8077

We can see that in general our entity recognition results are above 80% of F-measure with persons and locations approaching 90%, and with expectable decrease with organizations as the most controversial named entity type. Table 4 contains the results for normalized entities, illustrating the decrease in quality as normalization is added to extraction:

Table 4. Preliminary results for Entity Normalization task

Entity type	Precision	Recall	F-measure	Relative F-measure
Persons	0.8024	0.8433	0.8223	0.9313
Locations	0.9017	0.7741	0.8330	0.9349
Organizations	0.6490	0.5760	0.6103	0.8359
OVERALL	0.7725	0.7173	0.7439	0.9210

We calculate “relative F-measure” to see how our normalization engine works regardless of NER quality in general; we calculate it as F-measure from the second table divided by the F-measure from the first one. We can see that our overall relative score for normalization is over 0.9, which means that we can successfully normalize 90% of all extracted entities.

The third task, Fact Extraction, is quite challenging due to the FactRuEval restriction that fact fields should be filled with normal forms of the entities participating in the situation. Fact participants are mainly persons and organizations and most facts involve at least two participants, so with our Recall for both types of entities we can estimate our maximum Recall for the task roughly as a product of persons Recall and organizations Recall in the normalization task, i.e. about 0.5. Table 5 contains detailed results for every fact type in the Fact Extraction task.

Table 5. Preliminary results for Fact Extraction task

Fact type	Precision	Recall	F-measure
Ownership	0.3465	0.0916	0.1449
Occupation	0.6612	0.3620	0.4679
Meeting	0.6667	0.1481	0.2424
Deal	0.2342	0.0698	0.1076
OVERALL	0.5586	0.2377	0.3335

We can see that our actual recall is even lower than the estimated one. This can be explained by high deviation of distribution of certain fact types between the development and test sets: fact types that have poor representation in the development set have extremely low quality, leaving only the Occupation type as more or less adequate with nearly 47% of F-measure.

7. Conclusion and future work

We have developed a rule-based Information Extraction system for Russian and have successfully participated in all three tracks of FactRuEval, an independent

Information Extraction competition. We introduced a technique for normalizing complex entities for a morphologically rich language and scored considerably high in both Named Entity tracks yielding over 80% F-measure in Extraction task and over 74% in Normalization task. The relative score of our normalization engine, calculated regardless of the extraction quality, is about 92% of F-measure. In future, we are going to address yet unresolved issues, especially in the areas of ambiguity resolution and predictions to polish our normalization techniques. We also plan to introduce a syntax-based module to improve our results in Fact Extraction.

References

1. *Akbik A., Konomi O., Melnikov M.* (2013), Propminer: A workflow for interactive information extraction and exploration using dependency trees. Proc. of ACL (Conference System Demonstrations), Sofia, pp. 157–162.
2. *Daciuk, J., Mihov, S., Watson, B. W., Watson, R. E.* (2000), Incremental construction of minimal acyclic finite-state automata. Computational Linguistics, 26(1), 3–16.
3. *Dale, R., Moisl, H., Somers, H.* (2000), Handbook of Natural Language Processing. Taylor & Francis.
4. *Cunningham H.* (2005), Information Extraction, Automatic. Encyclopedia of language and linguistics, Second Edition, vol. 5, Oxford, pp. 665–677.
5. *Ermakov A.* (2005), Persons and organizations names' reference in Russian media: empirical patterns for computational analysis [Referentsija oboznachenij person i organizatsij v russkojazychnykh tekstakh SMI: empiricheskie zakonomernosti dl'a kompjuternogo analiza], Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2005" [Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii "Dialog 2005"], Bekasovo, 131-135.
6. *Ermakov A.* (2007), Fact extraction from dossier texts: anaphoric relation problems [Izvlechenie faktograficheskikh dannykh iz testov dosje: problemy anaforicheskogo ustanovlenija sv'azej], Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2007" [Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii "Dialog 2007"], Bekasovo, 172-178.
7. *Gareev, R., Tkachenko, M., Solovyev, V., Simanovsky, A., Ivanov, V.* (2013), Introducing Baselines for Russian Named Entity Recognition. In A. Gelbukh (Ed.), Computational Linguistics and Intelligent Text Processing, vol. 7816, Springer, Berlin, pp. 329–342.
8. *Grishman R., Sundheim B.* (1996), Message Understanding Conference—6: A Brief History. In: Proceedings of the 16th International Conference on Computational Linguistics (COLING), I, Kopenhagen, pp. 466–471.
9. *Hearst M. A.* (1992), Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th International Conference on Computational linguistics, COLING '92, Stroudsburg, pp. 539–545.

10. *Kiselev S., Ermakov A., Pleshko V.* (2004), Fact extraction from texts based on network description [Poisk faktov v tekste estestvennogo jazyka na osnove setevykh opisanij]. Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialogue 2004" [Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii "Dialog 2004"], Bekasovo, 282-285.
11. *Kuznetsov I.* (2013), The methods of discovery of objects and their links presented implicitly in texts. In Proceedings of the International Conference "Dialogue 2012" [Komp'yuternaya Lingvistika i Intellektual'nye Tekhnologii: Trudy Mezhdunarodnoy Konferentsii "Dialog 2012"], Bekasovo. Available at: http://www.dialog-21.ru/digests/dialog2012/materials/pdf/Кузнецов_И_П.pdf
12. *Mikolov T., Sutskever I., Chen K., Corrado G., Dean J.* (2013), Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, Harrahs and Harveys, pp. 3111-3119.
13. Proceedings of the 3rd ROMIP workshop [Trudy tret'ego rossijskogo seminara po otsenke metodov informatsionnogo poiska], Saint Petersburg, 2005.
14. *Sokirko A.* A short description of Dialing Project. Available at http://aot.ru/docs/sokirko/sokirko-candid-eng.html
15. *Yangarber R., Lin W., Grishman R.* (2002), Unsupervised learning of generalized names. In Proceedings of the 19th International Conference on Computational Linguistics, COLING '02, Taipei, pp. 1359–1141.