Extraction of Hypernyms from Dictionaries with a Little Help from Word Embeddings

Maria Karyaeva[1], Pavel Braslavski[2], and Yury Kiselev[3]

[1] Yaroslavl State University, Yaroslavl, Russia
mari.karyaeva@gmail.com,
[2] Ural Federal University, Yekaterinburg, Russia
pbras@yandex.ru,
[3] Yandex, Yekaterinburg, Russia
ykiselev.loky@gmail.com

Abstract. The paper investigates several techniques for hypernymy extraction from a large collection of dictionary definitions in Russian. First, definitions from different dictionaries are clustered, then single words and multiwords are extracted as hypernym candidates. A classification-based approach on pre-trained word embeddings is implemented as a complementary technique. In total, we extracted about 40K unique hypernym candidates for 22K word entries. Evaluation showed that the proposed methods applied to a large collection of dictionary data are a viable option for automatic extraction of hyponym/hypernym pairs. The obtained data is available for research purposes.

Keywords: hypernymy · semantic relations · thesaurus · word2vec

1 Introduction

Hypernymy relations such as *dog* $\prec$ *mammal* and *roadster* $\prec$ *car* are essential for organization of electronic thesauri, ontologies and taxonomies for natural language processing. Building such resources manually is a labor-intensive task. Manually built resources have an inherently narrow scope and need to be constantly maintained and kept up-to-date. Therefore, automatic methods for constructing linguistic resources have been demanded and actively explored over the last two decades. Pioneering experiments on the extraction of hypernymy relations between words from text corpora in early 1990s employed hand-crafted templates. Later, attempts were made to automatically expand the initial set of templates, as well as to combine them with distributional semantics representations. Nowadays neural networks have been actively using to detect hypernym/hyponym word pairs.

In this paper, we address the task of noun hypernym extraction from dictionary definitions in Russian. Earlier work on template-based extraction of hierarchical relations showed that encyclopedic data is best suited for this approach. Template-based methods require that both words (hyponym and hypernym) occur in the same sentence, which is not very common in arbitrary texts. Dictionaries often define a word through a more generic concept, for example:

– *Car*: a wheeled vehicle that moves independently, powered mechanically, steered by a driver.
– *Convertible*: a car whose roof can be removed or folded.[4]

The main advantage of this approach is its simplicity, but it has also additional benefits. When employing dictionary definitions, we naturally solve the problem of distinguishing hypernyms corresponding to different word senses expressed in turn in separate definitions. For example, both pairs $bass \prec fish$ and $bass \prec musical\ instrument$ are expected to be presented in a thesaurus. Dictionaries make it possible to find hypernyms even for infrequent words, which is rather hard in case of purely statistical methods. In addition, we use multiple dictionaries simultaneously, which potentially improves both precision and recall of the resulting pairs. As a supplemental method, we reproduce experiments on hypernym/hyponym pairs detection based on pre-trained word embeddings and supervised classification.

The study resulted in 47,831 (40,298 unique) hypernym candidates extracted for the 21,957-word initial list by three methods. Evaluation showed that the precision of the simple template-based methods lies in the range of 0.57–0.64 (see Section 5 for details). Classification-based methods using word embeddings demonstrate lower quality, but still can be considered as a complementary solution.

2 Related work

There are two main approaches to extraction of hyponymy/hyperonymy relations: 1) lexico-syntactic templates and 2) semantic vector representations. In addition, there are methods for building linguistic resources based on semi-structured data from Wiktionary and Wikipedia, as well as combinations of different approaches.

Researchers have been engaged with the task of automatic detection of hierarchical relations between words since early 1990s. In her pioneering work, Hearst [5] proposed a set of manually crafted templates aimed at extraction of hypernyms and co-hyponyms from a large text corpus, for example: x_1, x_2, x_3 *and other* y, where x_i and y are noun phrases representing co-hyponyms and a hypernym, respectively. Sabirova and Lukanin [17] applied a similar method to a corpus of texts in Russian. The studies showed that this kind of approach works best with encyclopedic data as a source. Kiselev et al. [8] followed this line of research and elaborated extraction of hyponymy/hypernymy relations from dictionary definitions in Russian using lexico-syntactic patterns. In the current work, we improve on the approach by using several dictionaries in parallel, extracting multiwords in addition to single-word candidates, as well as compliment

[4] A different, though less common approach is to define a word trough its synonyms: *car – a motorcar or automobile,* or through cognate words: *running – the action of the verb to run.*

the method with a simple approach based on pre-trained semantic vector representations.

In the works [2,3] initial template-based methods were improved by taking into account semantic similarity of words based on conjunctive structures and LSA representations. Snow et al. [20] elaborated a method for automatic extraction of such patterns, followed by classification-based hyponymy/hyperonymy detection.

Word embeddings [12] proved to be a good semantic representation of words in many natural language processing tasks. Thus, attempts have been made to use pre-trained models for the task of hypernymy detection. Several studies [1,16,21] investigated supervised classification and vector representations of words for hypernymy detection task: word pairs are presented either as concatenation or difference of their vector representations, and classifiers are trained on positive and negative word pairs. We reproduce these approaches in the current work. Fu et al. [4] elaborated this approach: first, vector differences are clustered (clusters are expected to reflect different types of hierarchical relations), then a separate projection is learned for each cluster using a training sample from an existing thesaurus. After space partitioning and projections learning, each word pair is assigned to a positive or a negative class. Shwartz et al. [19] combined syntactic and semantic information to address the task of hyponym/hypernym pair extraction: each candidate word pair is represented as concatenation of individual words' vectors and averaged vector representation of the paths in the dependency tree between words.

Many studies explore the possibility of building structured semantic resources from Wikipedia data, see for example [15,6,13].

3 Data

3.1 Dictionaries

We use six dictionaries of Russian as the main source of data in the current study:

(EFR) T. F. Efremova, New dictionary of Russian, 2000.
(BAS) S. A. Kuznetsov, Comprehensive Dictionary of the Russian Language, 1998.
(MAS) A. P. Evgenyeva, Small Academic Dictionary, 1957.
(BAB) L. Babenko, Dictionary of synonyms of the Russian Language, 2011.
(USH) D. N. Ushakov, Explanatory Dictionary of the Russian Language, 1935.
(OZH) S. I. Ozhegov, Explanatory Dictionary of the Russian Language, 1949 (1992).

These dictionaries span 75 years and reflect well the diversity of Russian lexicography, in particular – different approaches to definitions. Five of them are explanatory dictionaries, BAB is a relatively small vocabulary of synonyms, with definitions given for the whole synsets. EFR differs through its more 'analytical' definitions. As shown in [7], MAS and BAS contain many similar definitions.

We collected all noun entries along with all definitions from the dictionaries and retained only items presented in the list of 60,000 most frequent words in the Russian National Corpus (RNC)[5]. This resulted in 21,957 unique nouns. Main statistics of the dictionary dataset can be found in Table 1.

Table 1. Main statistics of the data used in the study: **entries** – noun entries from six dictionaries; **defs** – their definitions; **entries@60K** and **defs@60K** – entries and their definitions after filtering out infrequent word entries; **WIKT** – words in **entries@60K**, for those hypernyms are presented in Wiktionary; **WIKT_pairs** – words in **WIKT**, whose definitions contain hypernyms from Wiktionary; unique definitions – unique strings after lowercasing and punctuation removal.

	entries	defs	entries@60K	defs@60K	WIKT	WIKT_pairs
EFR	55,499	84,072	18,812	37,918	5,085	3,031
BAS	29,877	42,229	14,888	24,812	4,579	2,604
MAS	33,664	51,182	17,922	32,627	4,794	2,899
BAB	1,810	2,011	1,510	1,689	569	217
USH	33,005	50,615	16,465	30,410	4,602	2,536
OZH	18,709	26,961	13,428	20,914	4,278	2,484
Unique	66,784	233,718	**20,312**	**134,460**	5,230	4,253

3.2 Reference Hyponym/Hypernym Pairs

Wiktionary[6] is a large online dictionary, whose

content is curated by community members. At the time of writing, Russian Wiktionary contains more than 173,000 entries. A subset of dictionary entries contains semantic relations – synonymy, antonymy, hypernymy, and hyponymy. We collected 59,582 hyponym/hypernym noun pairs from Wiktionary. Then, we used the same 60K frequency list from RNC to filter out pairs containing rare words. Proper nouns, names, and narrow domain terms were removed, e.g. *Альза* $\prec$ *река* (*Alza* $\prec$ *river*), *Муцал* $\prec$ *имя* (*Muzzal* $\prec$ *name*), *алюмогидрид* $\prec$ *соединение* (*aluminumhydride* $\prec$ *compound*). This resulted in 10,826 pairs for 7,124 unique hyponyms.

We also matched these reference pairs with dictionary data (hyponym as a dictionary entry, one of whose definitions contains hypernym), see the last column in Table 1. It can be seen that about 39% of entries have hypernyms in one of their definitions.

3.3 Clustering of Definitions

In contrast to previous studies dealing with relation extraction from text, we deal with semi-structured data from several dictionaries. The multitude of data

[5] http://ruscorpora.ru
[6] https://ru.wiktionary.org/

sources increases coverage and provides additional evidence in case of frequent items. At the same time, many definitions from different dictionaries are very similar [7]. To mitigate the redundancy problem we first cluster definitions from different sources corresponding to the same word entry. In our initial dataset, 93% of word entries have more than one definition.

We apply graph-based clustering that proved to be efficient for many NLP tasks [22,14,9]. All definitions corresponding to a word entry are vertices; there is an edge between them if cosine similarity is greater than 0.15. Table 2 shows clustering of definitions for a word *пиявка (leech)* as an example.

As a result, 134,460 definitions are clustered into 48,739 clusters, 21,199 of them contain a single definition.

Table 2. Example of clustered definitions for a word *пиявка (leech)*.

♢ червь класса кольчатых // *a worm that belongs to the class of annelid worms* ♢ пресноводный червь, питающийся кровью животных, к телу которых он присасывается // *a freshwater worm sucking blood from their host animal*
♢ о жадном и жестоком человеке, живущем за счет других // *a greedy and cruel person who extorts profit from others* ♢ тот, кто живёт за счёт чужого труда и ведёт паразитический образ жизни // *somebody who lives a parasitic life*
♢ щелчок, удар пальцами по телу человека, причиняющий острую жгучую боль // *a finger flip on a person's body causing acute burning pain*

4 Methods

4.1 Single Words as Hypernyms

As mentioned above, we use the method for hypernym extraction described in the paper [8] as a starting point for this study. The approach is very simple: the first noun occurring in definition is extracted, with a short list of exception words, for example *kind, species, section,* etc. The method uses data from a single dictionary. In our case, we use definitions from several dictionaries, previously combined into clusters that presumably correspond to different word senses. The modification of our method is that we extract a single most frequent candidate from a cluster. If there is a tie and we cannot make a choice based on the frequency , then the candidate with the highest RNC frequency is selected. The example below demonstrates how this strategy works in case of four definition clusters and extracted candidates in case of the word entry *искусство (art)*:

1. *система (system)* ×2, ***отрасль (domain)*** ×3 – most frequent entry;
2. ***умение (skill)*** ×3, *знание (knowledge)* ×1 – most frequent entry;

3. *отражение (reflection)* $freq_{RNC} = 5,789$*, воспроизведение (reproduction)* $freq_{RNC} = 1,323$*,* ***деятельность (activity)***$freq_{RNC} = 46,298$ – RNC frequency;
4. ***дело (occupation)*** – single candidate.

Using this approach, we extracted 58,834 hypernym candidates from 48,739 definition clusters. In some cases identical hypernym candidates were extracted from different definition clusters. After eliminating duplicate pairs we ended up with 39,252 pairs in total. In case of 17% entries the method did not extract any candidate. 28% out of 39,252 extracted pairs are present in the list of hyponym/hypernym pairs extracted from the Wiktionary (see Section 3.2).

4.2 Multiwords as Hypernyms

If we look at single-word hypernym candidates extracted on the previous stage, we can notice that many of them are very abstract and though connected to their intended hyponyms, should probably reside not immediately above, but on a higher taxonomy level. For example: *адмирал* $\prec$ *чин* (*admiral* $\prec$ *rank*), *ил* $\prec$ *масса* (*silt* $\prec$ *mass*), *нездоровье* $\prec$ *состояние* (*ailment* $\prec$ *condition*). When inspecting source definitions, we noticed that it is possible to extract more immediate hypernyms, when considering multiwords, for example *военно-морской чин (navy rank)*, *вязкая масса (viscous mass)*, and *болезненное состояние (painful condition)*.

Multiword expressions (MWE) have been underrepresented in traditional dictionaries [7], but became an essential part of modern electronic linguistic resources. For example, about 41% of Princeton WordNet entries are multiword [18]; MWE constitute about 47% of total 115,000 RuWordNet entries [11].

To extract verbose hypernym candidates from definitions, we used several morphosyntactic patterns, see Table 3.

These patterns have been used in many previous MWE extraction studies. Table 4 illustrates the productivity of patterns by dictionary. According to these figures, **EFR** seems to have a somewhat different definition structure.

All the patterns correspond to nominal phrases, thus the extracted expressions potentially refine the single-word candidates obtained by the previous method. Note that patterns are nested: we can extract several multiwords from the same definition. A possible downside of these more complicated structures is that they can lead to extraction of non-lexicalized expressions or extraction of almost entire definitions. This is especially obvious in case of 4-grams, for example: *рецептура – способ изготовления лекарственных веществ (formulation – a method of manufacturing pharmaceutical substances); энциклопедист – представитель группы передовых мыслителей (encyclopaedist – a representative of a group of advanced thinkers); стоянка – место поселения первобытного человека (settlement site – a settlement of the primitives).* Nevertheless, even extracted items of this kind can be useful as glosses – many thesauri allow “empty” synsets containing a definition only that serve as a means for a more balanced hierarchy.[7]

Table 3. Multiword extraction patterns and their productivity (A – adjective; N – noun; N_G – noun, genitive case; Pr – preposition; in AN patterns adjective and noun must be coordinated).

Pattern	Example			#candidates
AN	аул (aul)	::	горное селение (a mountain village)	41 891
NN_G	вихор (cowlick)	::	прядь волос (a strand of hair)	23 145
AAN	баккара (baccara)	::	азартная карточная игра (a gambling card game)	3 328
NAN_G	хронология (chronology)	::	последовательность исторических событий (a sequence of historical events)	7 935
ANN_G	пядь (span)	::	русская мера длины (a Russian unit of length)	2 469
NN_GA	арахис (peanut)	::	растение семейства бобовых (a plant of the family of bean)	1 268
$NPrN$	ванна (bathtub)	::	сосуд для купания (a vessel for bathing)	9 629
NN_GAN_G	ватт (watt)	::	единица измерения активной мощности (a unit of active power)	1 123

Table 4. Percentage of definitions by dictionary, where the pattern has fired.

pattern	BAS(%)	EFR(%)	OZH(%)	BAB(%)	MAS(%)	USH(%)
N	92	77	95	88	80	84
AN	37	28	35	31	31	30
NN_G	19	20	17	23	15	14
AAN	3	2	3	3	2	2
NAN_G	6	4	4	5	4	4
ANN_G	2	2	2	2	1	1
NN_GA	1	1	0	1	0	0
NN_GAN_G	1	1	1	1	1	0
$NPrN$	5	4	5	3	4	5

As with the single-word hypernyms, our goal was to extract one expression from each cluster of definitions. In order to filter out less reliable bigrams, we used two additional criteria. We required that the candidate is 1) selected from at least two clusters of our dataset and 2) present in the RNC frequency list of

[7] See for example GermaNet, `http://www.sfs.uni-tuebingen.de/GermaNet/`.

bigrams.[8] The former criterion can be seen as a simple sanity check: a hypernym is expected to have several 'children'; while the latter one is a light lexicalization test. Extracted candidates that meet both criteria are championed. Candidates that fail to meet both conditions, are discarded. Since the number of extracted 3– and 4–gram is much lower, we kept all of them for subsequent evaluation.

For example, for the entry *сыч (athene)* 11 multiword hypernym candidates are extracted, including *сумеречная птица (twilight bird)* and *хищная птица (a predatory bird).* The latter one is championed, since it is potentially very productive as a hypernym. It is extracted from definitions of 17 entries, including *сип (sip), стервятник (vulture), кречет (gyrfalcon)*, etc.

In total, 61,134 bigrams and 17,628 3- and 4-grams were extracted. After applying our two criteria we ended up with 19,880 (18,360 bigrams and 1,520 3- + 4-grams) multiword hypernym candidates for 11,623 word entries.

4.3 Hypernym Extraction Based on Word Embeddings

An interesting property of word embeddings is that they capture not only semantic similarity of words, but also other relations between them. A famous example cited by Mikolov et al. [12] is that the closest vector to $vector(king) - vector(man) + vector(woman)$ is $vector(queen)$. There were several attempts to use pre-trained word embeddings for hypernymy detection, as well as learning dedicated embeddings for the task (see Section 2).

In this study we use pre-trained word2vec models from the RusVectōrēs project [10]. In particular, we employ 300-dimensional word vectors for Russian trained on Wikipedia and RNC (600 million tokens in total) using continuous skip-gram model.[9] We use `gensim` library[10] to find closest vectors. First, we trained a classifier using reference hyponym/hypernym pairs (see Section 3.2) as positive examples and pairs from a dataset originating from a semantic similarity shared task as negative examples. Then, using the classifier we extracted best candidate from the words closest to each item in our initial list of 21,957 words. The main difference of our approach from the similar experiments reported in the literature is that we aim at searching hypernyms not within a closed set of annotated word pairs, but 'in the wild'.

As positive examples we took 10,826 reference pairs, for 8,496 out of which pre-trained vectors are available. Out of these 8.5K reference pairs only in case of fewer than 3K a hypernym can be found as a similar entry in 1K words of word2vec outputs close to a given hyponym; 1.9K (23%) of them can be found in top-200 words. We sampled the same amount of negative pairs from 'unrelated

[8] The list `http://ruscorpora.ru/corpora-freq.html` contains 6.8 million bigrams with frequency above 3. We also matched the extracted multiwords with Wikipedia titles, but there were fewer than 9% matches, and we did not use this as a selection criterion.

[9] The embeddings can be downloaded from `http://rusvectores.org/models/`, model `ruwikiruscorpora_upos_skipgram_300_2_2018`.

[10] `https://radimrehurek.com/gensim/models/word2vec.html`

examples' of the RuThes dataset[11]. We experimented with vector concatenations (ConVec) and vector differences (DiffVec). We trained a classifier using SVM with RBF kernel implemented in `scikit-learn`[12]. Three-fold cross-validation resulted in 0.83 and 0.85 accuracy for ConVec and DiffVec, respectively, which is comparable with scores for similar experiment on closed-set hypernymy classification reported in the literature. To obtain hypernym candidates for initial word list not participating in training, we retain nouns from the top-200 words for each list item and feed them to the classifier. In addition, we require that the candidate words are present in RNC60K frequency list. The positively classified pair farthest from the separating hyperplane is considered the best match.

5 Evaluation

For evaluation we randomly sampled 300 words from the 14,833 entries from the initial list of 21,957 words excepting 7,124 unique entries from the Wiktionary hyponym/hypernym pairs. For these 300 words, we collected all hypernym candidates generated by all approaches. These pairs were evaluated by three assessors. The assessors judged each pair as correct (1) or incorrect (0). Cohen's kappa is 0.51, which indicates a moderate agreement. In addition, assessors could provide more fine-grained annotations:

– senses are *too distant* (**dist_sen**);
– the words is semantically similar, but reflect a *different relation* – synonymy, antonymy, co-hyponyms, association (**diff_rel**);
– expression is *grammatically incorrect*, expression is *incomplete*, *cognate* words, *inverse* relation, *definition* instead of hypernym (collapsed to **others** below).

Individual judgments were converted to common scores by applying an agreement rule – at least two assessors marked the pair as correct. Table 5 summarizes the evaluation results.

Table 5. Evaluation results.

method	# candidates	precision	diff_rel	dist_sen	others
1-word	485	0.57	0.04	0.04	0.28
2-word	236	**0.64**	0.03	0.00	0.23
3-word	14	0.57	0.00	0.00	0.43
ConVec	219	0.30	0.12	0.02	0.46
DiffVec	284	0.46	0.29	0.01	0.27
1w+DiffVec	399	**0.62**			
3|2w+(1w+DiffVec)	323	**0.61**			

It can be seen from the Table that candidate bigrams outperform unigrams in terms of precision, but have a lower coverage. Trigrams are represented scarcely

[11] `http://russe.nlpub.org/downloads/`
[12] `http://scikit-learn.org/stable/modules/svm.html`

in the evaluated sample, though these few candidates demonstrate decent precision. DiffVec outperforms ConVec in terms of coverage and precision. Expectedly, embeddings-based methods often return semantically similar words that manifest a relation different from hypernymy. We can slightly improve the methods by their combination. For example, intersection of single-word extraction (1w) and DiffVec (i.e. the extracted noun is assigned to the positive class by DiffVec) increases precision, though decreases coverage. A simple heuristics 'take a longer candidate if present, a DiffVec-filtered single-word candidate otherwise' delivers about the same precision. The combined approach allows to extract candidates of different word length without losing accuracy.

Figure 1 illustrates relation between the positions of the extracted single-word candidates in definitions and their scores. By design, the majority of the extracted candidates are located towards the beginning of the definitions. It can be seen that the precision of the extracted hypernyms drops after the fifth position, though there are still correct hypernyms quite distant from the word being defined. Table 6 shows that two two-word patterns have approximately equal productivity; the examples of three-word candidates are too few to make reliable conclusions.

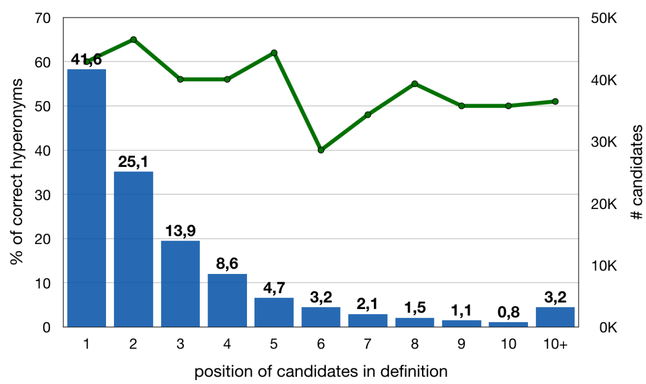

Fig. 1. Positions of extracted single-words candidates in original definitions (bars, right axis) and precision of each position (line, left axis). If the candidate is present in several definitions for the same entry, all positions are accounted for.

Table 6. Productivity of patterns.

	AN	NN_G	AAN	NAN_G	$NPrN$
#pairs	156	80	2	5	3
precision	0.62	0.69	1.0	0.60	0.67

6 Conclusion and Future Work

In this paper, we presented several methods of extraction of hyponym/hypernym relations based on morphosyntactic templates, simple heuristics, and word embeddings. Manual evaluation demonstrated feasibility of the methods and their combinations. Table 7 demonstrate that our methods are able to produce at least a single candidate for 97% of the RNC frequently list.

Our methods are efficient for practical use and we believe that our findings will be applied to different natural language processing tasks and obtained data will extend existing lexicographic resources. The obtained candidate hyponym/hypernym pairs are freely available for research purposes[13].

Table 7. Coverage of methods: percentage of the most frequent nouns, for which at least one candidate hypernym is extracted.

	1-word	2-word	3-word	ConVec	DiffVec	Total
# entries	18,111	12,304	1,496	19,307	20,428	**21,330**
RNC60K, %	82	56	7	88	93	**97**

In the future work we plan to address the following issues:

- In the current study we restricted the initial list of word entries by the RNC frequency list to be able to use dictionary- and embeddings-based methods in parallel. We will investigate the opportunity to extract less frequent hyponym/hypernym pairs from dictionaries thus increasing the coverage of the method.
- We did not evaluate the quality of clustering, used ad-hoc parameters for clustering and selected only one candidate from each cluster. We will investigate the impact of clustering on the overall extraction quality, as well as take a closer look at the ranking/selection of candidates from a cluster. Multiple candidates can be useful for matching the extracted pair with existing synsets, enriching existing synsets, and increasing the overall quality of the method.
- To train the classifier we applied pairs of unrelated words as negative examples. The study [19] suggests that using other relations (synonymy, part/whole, etc.) as negative examples can increase the specificity of the classifier. We plan to investigate this hypothesis in our future work.
- We restricted evaluation to top-1 candidate of each classifier. For a better understanding of the methods and their potential we plan to evaluate top-5 candidates.

Acknowledgments. MK was supported by RFBR grant #15-37-50912, PB and YK were supported by RFH grant #16-04-12019.

[13] https://github.com/YARN-semantic-relations/hyponymic-relationship

References

1. Baroni, M., Bernardi, R., Do, N.Q., Shan, C.c.: Entailment above the word level in distributional semantics. In: EACL. pp. 23–32 (2012)
2. Caraballo, S.A.: Automatic construction of a hypernym-labeled noun hierarchy from text. In: ACL (1999)
3. Cederberg, S., Widdows, D.: Using LSA and noun coordination information to improve the precision and recall of automatic hyponymy extraction. In: CoNLL. pp. 111–118 (2003)
4. Fu, R., Guo, J., Qin, B., Che, W., Wang, H., Liu, T.: Learning semantic hierarchies via word embeddings. In: ACL. pp. 1199–1209 (2014)
5. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: COLING. pp. 539–545 (1992)
6. Hu, J., Fang, L., Cao, Y., Zeng, H.J., Li, H., Yang, Q., Chen, Z.: Enhancing text clustering by leveraging wikipedia semantics. In: SIGIR. pp. 179–186 (2008)
7. Kiselev, Y., et al.: Russian lexicographic landscape: a tale of 12 dictionaries. In: Dialog (2015)
8. Kiselev, Y., Porshnev, S., Mukhin, M.: Method of extracting hyponym-hypernym relationships for nouns from definitions of explanatory dictionaries (*in Russian*). Software Engineering **10**, 38–48 (2015)
9. Kozareva, Z., Riloff, E., Hovy, E.: Semantic class learning from the web with hyponym pattern linkage graphs. In: ACL-HLT. pp. 1048–1056 (2008)
10. Kutuzov, A., Kuzmenko, E.: Webvectors: A toolkit for building web interfaces for vector semantic models. In: AIST. pp. 155–161 (2017)
11. Loukachevitch, N., Lashevich, G.: Multiword expressions in Russian thesauri RuThes and RuWordnet. In: AINL. pp. 1–6 (2016)
12. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: NAACL. pp. 746–751 (2013)
13. Navigli, R., Ponzetto, S.P.: BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial Intelligence **193**, 217–250 (2012)
14. Navigli, R., Velardi, P., Faralli, S.: A graph-based algorithm for inducing lexical taxonomies from scratch. In: IJCAI. vol. 11, pp. 1872–1877 (2011)
15. Ponzetto, S.P., Strube, M.: Deriving a large scale taxonomy from Wikipedia. In: AAAI. vol. 7, pp. 1440–1445 (2007)
16. Roller, S., Erk, K., Boleda, G.: Inclusive yet selective: Supervised distributional hypernymy detection. In: COLING. pp. 1025–1036 (2014)
17. Sabirova, K., Lukanin, A.: Automatic extraction of hypernyms and hyponyms from russian texts. In: AIST. pp. 35–40 (2014)
18. Shudo, K., Kurahone, A., Tanabe, T.: A comprehensive dictionary of multiword expressions. In: ACL-HLT. pp. 161–170 (2011)
19. Shwartz, V., Goldberg, Y., Dagan, I.: Improving hypernymy detection with an integrated path-based and distributional method. In: ACL. pp. 2389–2398 (2016)
20. Snow, R., Jurafsky, D., Ng, A.Y.: Learning syntactic patterns for automatic hypernym discovery. In: NIPS. pp. 1297–1304 (2005)
21. Vylomova, E., Rimell, L., Cohn, T., Baldwin, T.: Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In: ACL. pp. 1671–1682 (2016)
22. Widdows, D., Dorow, B.: A graph model for unsupervised lexical acquisition. In: COLING. pp. 1–7 (2002)