ИСПОЛЬЗОВАНИЕ ФОЛКСОНОМИИ ДЛЯ ОПРЕДЕЛЕНИЯ СЕМАНТИЧЕСКОЙ БЛИЗОСТИ

Клячко Е. (elenaklyachko@gmail.com)

Москва, Россия

Ключевые слова: семантическая близость, фолксономия, совместная категоризация, социальные сети

USING FOLKSONOMY DATA FOR DETERMINING SEMANTIC SIMILARITY

Klyachko E. (elenaklyachko@gmail.com)

Moscow, Russia

This paper presents a method for measuring semantic similarity. Semantic similarity measures are important for various semantics-oriented natural language processing tasks, such as Textual Entailment or Word Sense Disambiguation. In the paper, a folksonomy graph is used to determine the relatedness of two words. The construction of a folksonomy from a collaborative photo tagging resource is described. The problems which occur during the process are analyzed and solutions are proposed. The structure of the folksonomy is also analyzed. It turns out to be a social network graph. Graph features, such as the path length, or the Jaccard similarity coefficient, are the input parameters for a machine learning classifying algorithm. The comparative importance of the parameters is evaluated. Finally, the method was evaluated in the RUSSE evaluation campaign. The results are lower than most results for distribution-based vector models. However, the model itself is cheaper to build. The failures of the models are analyzed and possible improvements are suggested.

Keywords: semantic similarity, folksonomy, collaborative tagging, social networks

1. Introduction

Measuring semantic similarity is important for various natural language processing tasks, including Textual Entailment, Word Sense Disambiguation etc [1]. The aim of The First International Workshop on Russian Semantic Similarity Evaluation (RUSSE) [14] was to carry out an evaluation campaign of currently available methods for the Russian language.

The organizers provided several training sets. They also performed the evaluation on the test set.

2. Related work

2.1. Semantic similarity measurements

As described in [1], the approaches to semantic similarity measurement can be divided into knowledge-based ones or context-based ones. Knowledge-based approaches use taxonomies with pre-annotated world-relations. These taxonomies may be leveraged through collaborative tagging, for example:

1. tags made by software programmers for their projects at the FreeCode resource [18]
2. geographical tags at the Open Street Map project [3]
3. Flickr[1] image tags [16]
4. Del.icio.us[2] tags [16]

We can roughly divide the approaches to processing taxonomy data in the following groups. Naturally, features from different groups can be used jointly.

1. graph-based methods: the ontology is considered to be a graph
   a. in [1], a version of Page Rank is computed for both words, resulting in a probability distribution over the graph. Then the probability vectors are compared using cosine similarity measure
   b. in [4], path length features are used
2. ontology-based methods: these methods take into account the hierarchical structure of an ontology:
   a. in [4], the ratio of common and non-common superconcepts is calculated
   b. in [5], a feature which is based on the depth of the concepts and their least common superconcept is calculated
3. vector-space models: vectors are constructed, and their similarity is measured
   a. in [3], the vector space coordinates are words from term definitions, which were created as a part of a collaborative project.
   b. in [18], vectors of tf and idf scores are constructed. In [16], these vectors also have a temporal dimension

[1] https://www.flickr.com/

[2] https://delicious.com/

2.2. Pre-processing tags and refining tag structure

In [17], pre-processing techniques for folksonomy tags are described. These techniques involve normalizations and help cluster the tags better. In [12], the authors leverage user information in order to get a more precise understanding of tag meanings.

In [8], [10], and [15], a folksonomy is used for getting synonym and homonym relations between words. The authors reduce the dimensionality of the tag space by clustering the tags. Various measures are used, such as the Jaccard similarity coefficient, a mutual reinforcement measure, and the Jensen-Shannon divergence

In [2], lexico-syntactic patterns, which are traditionally used to get a taxonomy structure out of texts, are used to refine the taxonomy structure, which is constructed via obtaining tags from a collaborative resource.

2.3. Natural language generation

In a number of works, folksonomy structure is used in natural language generation tasks, namely for referring expression generation or text summarization [6, 13]

3. The goals of this paper

The aim of this work was to assess the contribution a folksonomy can make to word similarity measurements.

Vector-space models seem to be quite efficient for the word similarity task. However, such approaches are sometimes not easy to interpret linguistically, and using an ontology is sometimes preferrable. On the other hand, the construction of a manually-crafted ontology can take a lot of time. As a result, using a folksonomy seems to be an appropriate trade-off. The influence of various parameters of the folksonomy should also be investigated. Finally, studying the structure of a tag-based folksonomy as a quasi-natural object is quite interesting.

4. Folksonomy construction

For the RUSSE shared task, a folksonomy graph was built as a co-occurrence network of photo tags from Flickr.

The Flickr API was used to collect tags from photos in a database. The process was organized as follows:

1. start with an array of about 90,000 words (A. Zaliznyak's dictionary [19], the electronic version provided by SpeakRus[3]) and an empty graph.
2. for each *word1* in the array:
   a. get all photos tagged with *word1*

[3] http://speakrus.narod.ru/dict-mirror/

b. for each photo collected in (a):
   i. collect all other Russian-language tags from the photo. Use the number of photos to calculate the tag frequencies. As a result we get a number of (*word, frequency*) pairs.
   ii. for each *word2* with frequency *freq* (from the pairs collected in (i)) we create an edge in the graph: (*word1, word2, freq*)

Tables 1 and 2 show two fragments of the resulting co-occurrence matrix for the words "автобус" ('bus') and "ягода" ('berry'):

Table 1. A fragment of the frequency matrix for "автобус" ('bus')

word1	word2	word2 translation	frequency
автобус	природа	nature	146
автобус	улица	street	135
автобус	транспорт	transport	132
автобус	социалистически	socialist (*in Bulgarian language*)	91
автобус	комунистически	communist (*in Bulgarian language*)	90
автобус	россия	Russia	63
автобус	город	city	46
автобус	москва	Moscow	40
автобус	путешествия	travelling	40
автобус	корабль	ship	35

Table 2 A fragment of the frequency matrix for "ягода" ('berry')

word1	word2	word2 translation	frequency
ягода	россия	Russia	45
ягода	лето	summer	31
ягода	природа	nature	31
ягода	ягоды	berries	29
ягода	клубника	strawberry	28
ягода	красный	red	21
ягода	подмосковье	Moscow region	19
ягода	малина	raspberry	17
ягода	смородина	currant	16
ягода	еда	food	15
ягода	осень	autumn	15
ягода	флора	flora	15
ягода	москва	Moscow	14
ягода	вишня	cherry	13
ягода	дача	country cottage	13
ягода	черника	bilberry	13
ягода	дерево	tree	12

Language detection was the main issue at that stage. Flickr does not distinguish between the languages of the tags. The tags are also too short for a language detection tool to detect the language well enough. The Python-ported Google's detection library[4] was used for language detection. However, it soon turned out to filter some Russian words. As a result, Zaliznyak's dictionary itself was used as a source of additional checks. Probably, using a large corpus of Russian words would be a better way of detecting Russian-language words in this case. The publicly available data on the author of the tag could also be used.

The program to collect the data is a Python script available at https://github.com/gisly/word_similarity.

5. The resulting structure of the folksonomy

5.1. The folksonomy graph

The resulting folksonomy is a graph of 96,015 nodes and 1,015,992 edges. The mean node degree is approximately 21.16.

Logically speaking, the graph should be undirected because the co-occurrence relation should be symmetric. However, two problems made this impossible:

- the language detection bug described above led to the fact that sometimes *word1, word2* edge was present, but *word2, word1* was not because *word1* was not detected to be a Russian word
- the Flickr database is a not a snapshot: it is a continuously changing dataset. It means the same edge inconsistence as described above.

Naturally, the graph could have been made undirected after completing the download. However, we chose to leave it as it is and simply count for the edges' being directed.

5.2. The folksonomy graph as a complex network

What is interesting, the folksonomy graph turns out to be a complex network (in the same sense as a graph of people relations or a word co-occurrence graph; cf. [11]).

The node degree distributions fits the power-law, which is typical for a social network [11]. Fitting the power law[5], we a get a p-value of 0.99 for, which indicates the hypothesis of the power-law distribution cannot be rejected. The exponent value is 1.64.

The log node degree distribution graph is shown in fig. 1.

---

[4] https://pypi.python.org/pypi/langdetect

[5] the fit was made using the R package: http://www.inside-r.org/packages/cran/igraph/docs/power.law.fit

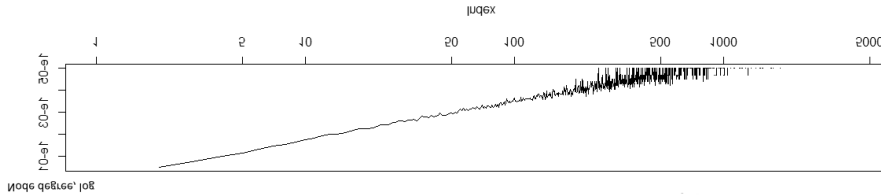

Fig. 1 Node degree distribution (log coordinates)

In table 3, top-10 words ordered by their degree are shown:

Table 3. Top-10 nodes ordered by node degree

word	translation	node degree
россия	Russia	4799
природа	nature	4096
красный	red	3875
москва	Moscow	3618
улица	street	3579
синий	blue	3543
солнце	Sun	3514
белый	white	3475
портрет	portrait	3366
отражение	reflection	3336

6. Training data

The RUSSE campaign consisted of two tasks. In the relatedness task, word relations (synonymy, hypo/hyperonymy were considered). In the association task, free associations were considered. As a part of the RUSSE evaluation campaign, several training and test datasets for each task were created by the organizers. The datasets are different in their origin. Some of were created through an online collaborative procedure, whereas others are extracted from large thesauri. A detailed description of these datasets as well as download links are given at the RUSSE website[6].

At first, these datasets contained only positive examples[7]. Therefore, we used a set of manually crafted negative examples. The negative examples were created by picking two random words from a large word set (the Wikipedia dump scores[8]), and manually excluding those which were really semantically similar to each other.

---

[6] http://russe.nlpub.ru/task/

[7] automatically generated negative examples were provided later

[8] https://s3-eu-west-1.amazonaws.com/dsl-research/wiki/wiki-cooccur-ge2.csv.bz2

During training, we mainly used the ***ae*** and ***rt*** training data, experimenting with different sizes of their subsets. ***ae*** are word association measures extracted based on an association. ***rt*** are word relatedness measures extracted from a thesaurus.

7. Features

For two words (*word1* and *word2*) the following features were calculated:

1. the existence of *word1* and *word2* nodes in the network (Y/N)
2. do *word1* and *word2* have the same part of speech[9]? (Y/N)
3. the existence of a path between *word1* node and *word2* node (Y/N)
4. path length: the number of nodes in the shortest path if the path exists (a number or NONE)
5. weighted path length (if the path exists; a number or NONE). In the shortest path, for each pair of nodes, the frequency of their joint occurrence is calculated. It is then divided by the frequencies of the individual words. The resulting measures are multiplied. Finally, a logarithm of the resulting number is taken.
6. the frequencies of the nodes in the path if the path exists (numbers or NONE). Each frequency is a separate feature.
7. the node degrees of the nodes in the path if the path exists (numbers or NONE). The degree of a node is the number of edges directly connected to the given node. Each degree is a separate feature.
8. the PageRank of the nodes in the path if the path exists (numbers or NONE)
9. the Jaccard similarity of *word1* node and *word2* node (a number). The Jaccard similarity coefficient is defined as:

   (the number of common neighbors of *word1* and *word2*)/(the size of the union of all neighbors of *word1* and *word2*)

10. the Dice similarity of *word1* node and *word2* node (a number). The Dice similarity coefficient is quite similar to the Jaccard coefficient and is defined as:

    2*(the number of common neighbors of *word1* and *word2*)/(the number of all neighbors of *word1* and *word2*)

11. the cosine similarity of the neighbor vector of *word1* and the neighbor vector of *word2*

[9] https://pythonhosted.org/pymorphy/ was used

7.1. The classification task

The classifiers were to solve the following task: each pair of words (*word1* and *word2*) should be classified as "similar" or "non-similar". Depending on the nature of the classifier, it was to produce either a binary score (0 or 1), or a number in the interval [0; 1]. In the latter case, the score was converted into the corresponding binary score:

- values ≤ 0.5 were considered to be 0
- values > 0.5 were considered to be 1

8. Machine learning algorithms

I tried several machine learning algorithms, such as Conditional Tree Inference, and Ada-Boost, implemented in the corresponding R packages (ctree[10] and ada[11]). The choice of these algorithms is mainly due to the fact that their results can be easier interpreted than the results of other algorithms.

8.1. Conditional Tree inference

A conditional tree is a kind of a decision tree. When building the conditional decision tree, the algorithm tests whether the hypothesis of the target variables's independence of the parameters can be rejected or not. If the hypothesis is rejected, it chooses the "strongest" parameter as a new node in the tree and proceeds with the other parameters [9].

In fig 2, the conditional tree which was built using the ***ae*** and ***rt*** subsets of the training data is presented.

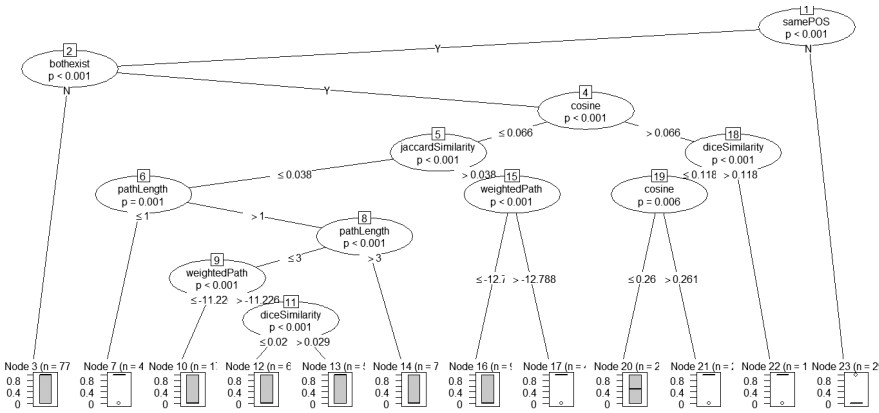

Fig. 2. The conditional tree created using the folksonomy graph and the ***ae*** training data subset

[10] http://cran.r-project.org/web/packages/partykit/vignettes/ctree.pdf

[11] http://cran.r-project.org/web/packages/ada/ada.pdf

8.2. AdaBoost

AdaBoost uses a committee of several weak classifiers (e. g., decision trees) and ends up calculating weights for these classifiers [7].

In fig 3, the variable importance plot constructed by AdaBoost is presented. The variable score shows the relative score of the variable.

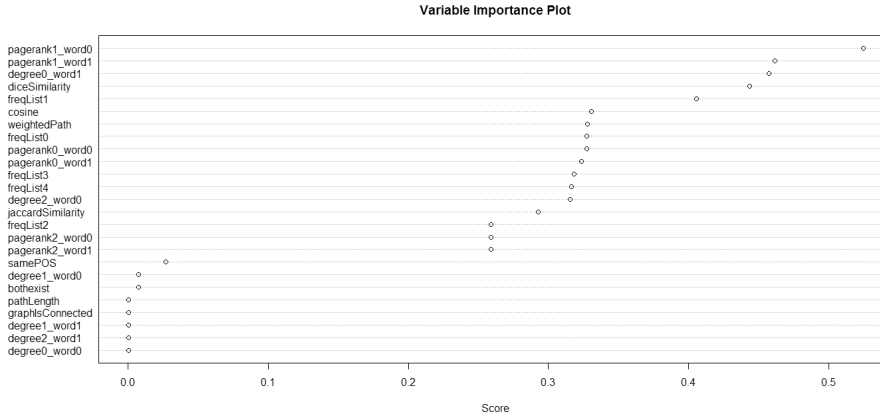

Fig. 3. The variable importance plot created by AdaBoost using the folksonomy graph and the ***ae*** training data subset

9. Evaluation

9.1. Cross-validation on the training set

I performed 4-fold cross-validation on the ae training set. The best average accuracy was 0.76 for the conditional tree model and 0.75 for the ada boost model. The best average precision was 0.73 for the conditional tree model and 0.70 for the ada boost model.

9.2. Final evaluation on the test set

Final evaluation was performed by the organizers[12]. The results for the folksonomy model are given in table[13] (model ids starting with "2-"):

---

[12] https://github.com/nlpub/russe-evaluation/tree/master/russe/evaluation

[13] from https://docs.google.com/spreadsheets/d/190qw6O_r8xAxPM2SK8q-R-0ODp2wDx-8qzh9Lr31jmSY/edit?usp=sharing

Table 4. The evaluation results for the folksonomy model provided by the organizers

HJ (human judgement for relatedness)	**RT-AVEP/ ACC** (average precision/ accuracy for ae-relatedness)	**AE-AVEP/ ACC** (average precision/ accuracy for ae associations)	**AE2-AVEP/ ACCURAC** (average precision/ accuracy for ae2 associations)	**Method Description**
0.3717	0.6815/0.5670	0.5195/0.4652	0.7282/0.6369	ctree, larger training subset
0.2490	0.7275/0.5396	0.5985/0.4795	0.7301/0.5903	AdaBoost, smaller training subset
0.2436	0.7183/0.5354	0.5802/0.5194	0.6732/0.5550	AdaBoost, larger training subset

10. Analysis

10.1. Intrinsic analysis: variable importance

From the output of AdaBoost and ctree, we can see that both algorithms consider the following parameters important:

- cosine similarity
- dice similarity
- jaccard similarity
- weighted path

Because of the structure of the network, the existence of the path itself does not mean much. Firstly, as we saw above, hubs such as “Russia”, “Moscow”, or “portrait”, which actually hold meta-information about a photo, connect most nodes with each other. Secondly, there may be an accidental connection between two words. For example, there is a photo tagged with words “egg” and “world” and it is an art representation of the world map on the eggshell. Naturally, it is an art concept and not the common truth.

Therefore, we should avoid two long paths because they may have a hub node inside. Moreover, we should avoid “accidental” paths.

The path length parameter and the weighted path parameter were thought to be the solution.

Actually, this intuition corresponds well enough with the ctree result: the larger the weighted path logarithm is, the greater is the probability of words being connected. It means the words are more likely to be related if the weighted path value is closer to one. Therefore, if the words are too frequent, we avoid considering them connected.

The conditional tree model also has two more important parameters: "both exist" and "same POS".

The scarcity of the photo tag data means that a lot of words simply lack. Therefore, the "both exist" feature simply prevents such words from being considered. However, naturally, the absence of the word in the folksonomy dictionary may only correlate with the word frequency in the everyday usage and not with its possible similarity with other words. For example, we cannot expect a folksonomy to have words like "яйцепродукты" ('egg products', a very special term from the food industry). Therefore, the parameter is perhaps useless and makes more noise than helps.

As regards the same POS feature, it is quite useful for the relatedness task because the common part of speech is usually considered to be important in the definitions of synonymy, hyponymy etc. However, it is really useless for the relatedness task.

There is also one intuitive problem with the ctree rules. According to them, if the similarity parameters are very low, but there is a direct link between the words, the words are considered to be related. In this case, the word frequencies are not analyzed at all.

10.2. Evaluation results

The algorithm performed quite consistently with the cross-validation results and considerably worse than the other competing methods.

10.2.1. Test set variations

We could expect that the photo tag similarity means association closeness and not relatedness. Moreover, we chose more ***ae*** training data as a training set. Therefore, the method was expected to work better on the association task than on the relatedness task.

Actually, the method does perform best on the ***ae2*** test set, which is a result of an online association experiment. The main reason for the poor performance on the Russian Associative Thesaurus test set is the absence of the thesaurus words in the folksonomy dictionary.

As regards the relatedness task, the method performs quite well on the RuThes relatedness subset. However, the ***hj*** (human judgment) results are poor. Why is it so that the two subsets expose different behavior?

Firstly, a subset of ***rt*** data was used for training. Secondly, in ***hj*** a finer-grained similarity score is given to word pairs, which is harder to reproduce.

10.2.2. The problems and possible solutions

In the table below, we collected several typical cases of the model's and failures. We then speculate of the possible ways of improving the model. We also mention the model's successes to show that they are not accidental.

Table 5. Error analysis

word0	word1	predicted ctree	actual	explanation	the source of the problem
true positives					
бас ('bass')	звук ('sound')	0.96	1	large cosine and dice similarities	no problem
рис ('rice')	крупа ('groats')	0.70	1	small weighted path (through the word "традиционный" 'traditional'); the photos actually depict some traditional meals.	no problem
false positives					
армия ('army')	река ('river')	0.96	0	large cosine and dice similarities	the common neighbors are mostly names of places
каланча ('watchtower/a tall person')	павильон ('pavilion')	0.96	0	large cosine similarity	perhaps, it is an annotation error. In my opinion, it is doubtful that the words are not co-hyponyms
сварщик ('welder')	изолятор ('isolation ward')	0.70	0	small weighted path through the words "синий" 'blue' and "Ярославль" 'Yaroslavl, a name of a city'	the words in the path are a hub and a name of a place
введение ('introduction')	сингл ('single (song)')	0.74	0	the word "single" does not exist in the database, so the "both exist" parameter works	the scarcity of the words and the "both exist" parameter
диагональ ('diagonal')	самолет ('airplane')	0.78	0	there is a direct link between the words.	The similarity parameters (cosine, dice etc) are very low, and in this case the path length parameter works. Perhaps, more careful analysis of negative examples could prevent such a rule from appearing in the ctree. However, such accidental links are intrinsic to the folksonomy so little can be done about it in general.
false negatives					
крыша ('roof')	ветхая ('dilapidated')	0.01	1	the word "ветхая" is not present among the tags in the given form. Furthermore, they have got different part of speech categories.	The corresponding masculine form of the adjective is present. The word can also be found in the photo descriptions
неделя ('week')	зачетная ('of exams'; the whole means 'exam week')	0.01	1	the words have different part of speech categories. Moreover, they are not connected well enough	the words co-occur in the photo descriptions but not in tags. The POS parameter is perhaps harmful

In order to improve the results, the following should be considered:

1. the hubs and place names usually contain meta-information, and do not depict the object shown in the photo. They should be filtered or somehow penalized. It can be done using geography databases and the graph statistics
2. all forms of a word should be considered. It can be achieved with a morphological analyzer.
3. photo descriptions and comments to photos should also be considered. They are accessible via the Flickr API.
4. more tags can actually be downloaded using more seed data, and adding non-vocabulary data
5. better language detection can be done (e. g., using a larger word list or simply taking all Cyrillic letter words)

10.3. Overall contribution

Although collecting the tags was inspired by the RUSSE shared task, the work has independent results, too. The way the folksonomy has been collected turns out to be valid because the resulting structure can be easily interpreted. Therefore, the method presented can be used in other natural language processing tasks (e. g., natural language generation, recommending services). Moreover, as far as we know, there are no similar publically shared open folksonomies for the Russian language

However, the problems we faced show that the data is very noisy and that we should pay more attention to normalizing it. Firstly, we should have paid more attention to the language detection problem. Secondly, the origin of the data should have taken into account. As the tags are connected with photos, they contain a lot of extra-linguistic information, which should be dealt with.

References

1. *Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., and Soroa, A.* (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In Proceedings of NAACL-HLT 2009, pages 19–27.
2. *Almoqhim, Fahad, David E. Millard, and Nigel Shadbolt* (2014) "Improving on Popularity as a Proxy for Generality When Building Tag Hierarchies from Folksonomies." Social Informatics. Springer International Publishing. 95–111.
3. *Ballatore, Andrea, David C. Wilson, and Michela Bertolotto.* (2013) "Computing the semantic similarity of geographic terms using volunteered lexical definitions." International Journal of Geographical Information Science 27.10: 2099–2118.
4. *Banea, Carmen,* et al. (2012) "Unt: A supervised synergistic approach to semantic text similarity." Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. Association for Computational Linguistics.

5. *Batet, Montserrat, et al.* (2013) “Semantic similarity estimation from multiple ontologies.” Applied intelligence 38.1: 29–44.
6. *Boydell, Oisin, and Barry Smyth* (2007) “From social bookmarking to social summarization: an experiment in community-based summary generation.” Proceedings of the 12th international conference on Intelligent user interfaces. ACM.
7. *Culp, Mark, Kjell Johnson, and George Michailidis* (2006) “ada: An r package for stochastic boosting.” Journal of Statistical Software 17.2: 9.
8. *Eynard, Davide, Luca Mazzola, and Antonina Dattolo* (2013) “Exploiting tag similarities to discover synonyms and homonyms in folksonomies.” Software: Practice and Experience 43.12: 1437–1457.
9. *Hothorn T.* et al. (2011) Package ‘party’: A laboratory for recursive partitioning.
10. *Mousselly-Sergieh, Hatem, et al.* (2013) “Tag Similarity in Folksonomies.” INFORSID.
11. *Newman, Mark* (2008) “The physics of networks.” Physics Today 61.11: 33–38.
12. *Niebler, Thomas,* et al. (2013) “How tagging pragmatics influence tag sense discovery in social annotation systems.” Advances in Information Retrieval. Springer Berlin Heidelberg. 86–97.
13. *Pacheco, Fabián, Pablo Ariel Duboue, and Martín Ariel Domínguez.* (2012) “On the feasibility of open domain referring expression generation using large scale folksonomies.” Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies. Association for Computational Linguistics.
14. *Panchenko A., Loukachevitch N. V., Ustalov D., Paperno D., Meyer C. M., Konstantinova N.* (2015) “RUSSE: The First Workshop on Russian Semantic Similarity”. In Proceeding of the Dialogue 2015 conference. Moscow, Russia
15. *Quattrone, Giovanni,* et al. (2012) “Measuring similarity in large-scale folksonomies.” arXiv preprint arXiv:1207.6037.
16. *Radinsky, Kira,* et al. (2011) “A word at a time: computing word relatedness using temporal semantic analysis.” Proceedings of the 20th international conference on World wide web. ACM.
17. *Solskinnsbakk, Geir, and Jon Atle Gulla.* (2011) “Mining tag similarity in folksonomies.” Proceedings of the 3rd international workshop on Search and mining user-generated contents. ACM.
18. *Wang, Shaowei, David Lo, and Lingxiao Jiang.* (2012) “Inferring semantically related software terms and their taxonomy by leveraging collaborative tagging.” Software Maintenance (ICSM), 2012 28th IEEE International Conference on. IEEE.
19. *Zaliznyak, A.* (1977). Grammaticeskij Slovar’ Russkogo Jazyka. Slovoizmenenie (‘The Grammatical Dictionary of the Russian Language. Inflection’), Moscow: Russkij Jazyk.