Verb-Noun Collocation and Government Model Extraction from Large Corpora

Vladislav Tushkanov, Oksana Dereza

National Research University Higher School of Economics,
v.tushkanov@outlook.com, oksana.dereza@gmail.com

Abstract. Knowing the government model, or argument structure, of a verb is crucial for many NLP tasks. In this article, a method of automatic extraction of verbs from large annotated corpora is devised. This method allows to computationally efficiently extract government models and particular arguments for every verb using a simple window-based approach by iterating through each sentence with a window of fixed size and applying frequency filters to filter out noise.

Keywords: collocations, verb government models, argument structure, corpus methods, parsing

1 Introduction

Automated acquisition of lexical knowledge is one of the important tasks on which many natural language processing tasks depend. Knowing semantic and syntactic properties of the words in a given language is crucial for such tasks as e.g. natural language generation, machine translation [1] and word sense disambiguation [2]. This is especially true as far as verbs are concerned [3].

One important property of a verb is its government model, or argument structure. It is a property that is inherent to a particular verb and prescribes the grammatical (case, number) and semantic properties of a noun which are necessary for it to be subordinated to this verb [4]. Setting the semantic properties aside, we also consider prepositions to be part of the government model, trying to exhaustively enumerate all the frequent nouns which are used with any given verb and abide by that government model. In a nutshell, the task which we undertake in this article is to answer the following question: *what can we do with what (and in what way)*?

The aim now is to automatically extract collocations of the kind verb + subordinate noun/pronoun from a large collection of texts. By collocation we mean any combination of a verb and a noun, possibly supported by a preposition, which is possible as far the language use is concerned. In other words, the words form a collocation if it is "okay" to use them together and their combination "feels natural".

2 Experiment

We use a simple bag of words approach with a window of size [-5, +5] to learn the collocations and the way a verb governs its objects, including the preposition and case. This window size is optimal due to the fact that it is the mode of the distribution of distances between a verb and its subordinate noun [5]. Then we try to give some quantitative expression to the "normalness" and "okay-ness" of the extracted collocations. Another approach would be to infer a syntactic tree from a sentence and use it to find the collocations [7]; however, being much more laborious, this approach does not seem to be superior to the simpler bag-of-word method [5].

When trying to extract collocations and filter out noise, instead of calculating PMI [5], which seems less relevant for smaller corpora, we rely on the following heuristic assumptions:

1. An object is a noun, a substantival pronoun or a cardinal numeral.
2. If a verb has an object, it has a noun inside the window of size n.
3. If the object follows a verb or precedes it, there are no other verbs between them.
4. If the object follows a verb or precedes it, given the third assumption is satisfied, it assumes a non-nominative case.
5. If a following noun or pronoun is identified as an object under assumptions 2-4, and there is a preposition between them, this preposition is part of the verb's government model.
6. If a preceding noun or pronoun is identified as an object under assumptions 2-4, and there is a preposition before the noun, this preposition is part of the verb's government model.
7. If there are two candidates to be recognized as an object, we prioritize the noun in the accusative case over the other one. If their cases coincide, we treat them as two distinct collocations.
8. The window does not cross a sentence boundary.

The window object search model works as follows. First, we use a stack to represent the contents of the window (a verb in the centre and l words before and after it) and start popping elements from its end (which is logical because it is more natural for an object to come after the verb). When we encounter a noun, we make it a candidate for being an object. We memorize it, as well as its case and lemma, and continue popping. If having encountered a noun we pop a preposition, it becomes a candidate for being part of the government model. If we encounter another verb, we clear the memory. In case another noun is found, we compare their cases and act according to the 7th assumption. When we reach the verb in the centre of the window, if no nouns are in memory, we continue the iteration over the stack, memorizing nouns and prepositions, or stop and return the nouns and prepositions. Having encountered a verb, we stop the iteration and return empty list of collocations, as nouns, if found, are most likely objects to this verb than to the verb in the centre of the window. We estimate the

complexity of this method to be $\mathcal{O}(mn)$, where m is the number of words in the corpus and n is the number of verbs. Subjectively, the whole parsing procedure takes two to three minutes on a low-tier notebook.

After we parse the corpus utilizing the method devised above, we store the data as a pandas dataframe. It is used to store collocations and contains the following columns: verb, noun, case, preposition, number, and the number of occurrences of the collocation. The last one will later be used to calculate statistics and, moreover, allows us to somewhat scale down the data. Table 1 shows *verb + noun* collocations from our dataframe, derived by processing Russian National Disambiguated Corpus [9] of 6 million tokens, with frequency count > 100. We are not particularly interested in the token of a noun, but the case is important, as it will help to recreate the form later using some morphological inflector (e.g. pymorphy2).

Table 2 contains the collocations with the verb *забить 'to hit in'* with the relevant ones in italics to be compared with [5] (these are different due to a different corpus and a slightly different task).

After parsing the whole corpus, we can extract the information on most probable government model for each verb by simply choosing the most frequent model across all collocations for a given verb. We skip the verbs with the collocation count less than two for each model we have discovered in the corpus to rid the resulting data of noise.

As a result of this experiment, we obtain two separate databases. The first one is all the possible candidates for being an argument to a given verb, the other one is a list of the most probable government models for 10000 verbs. Frequency counts for both of them suggest that the entries are distributed according to Zipf's law, which does make sense, as this law is usually applicable to most data in linguistics where frequency of phenomena is concerned [6].

The list of government models with frequency counts is given in Table 3, while Table 4 shows most probable government models for 20 random verbs. The distribution of top-1000 collocations and most probable government models can be seen in Figures 1 and 2 respectively.

This dataset was generated as part of a larger project, i.e. automatic generation of grammar tasks for people suffering from aphasia [8]. With most frequent verbs being the most useful for it, we quantify the "normalness" of a collocation as its absolute frequency compared to absolute frequencies of other collocations this verb forms. As there is no golden standard to measure its quality against, an expert asssessment was used. We randomly sampled 500 collocations with frequencies of 5 and more and assessed them to either be true collocations or random noise (like *думать ему*, 'to think to him'. Using this approach, we estimated the accuracy to be 91.4%.

3 Conclusion

The method of extracting government models for a verb using a window-based approach by iterating through each sentence with a window of fixed size and

Verb	Object	POS	Case	Number	Preposition	Count	Model	Translation
быть	год	S	gen	pl	None	298	gen	‘to be... years (ago)’
покачать	голова	S	ins	sg	None	281	ins	‘to shake one’s head’
махнуть	рука	S	ins	sg	None	252	ins	‘to wave one’s hand’
обратить	внимание	S	acc	sg	None	223	acc	‘to pay attention’
быть	человек	S	ins	sg	None	221	ins	‘to act human’
пожать	плечо	S	ins	pl	None	220	ins	‘to shrug’
иметь	право	S	acc	sg	None	201	acc	‘to have right’
поднять	голова	S	acc	sg	None	177	acc	‘to raise one’s head’
закрыть	глаз	S	acc	pl	None	167	acc	‘to shut one’s eyes’
пить	чай	S	acc	sg	None	157	acc	‘to drink tea’
сидеть	стол	S	ins	sg	за	156	ins,за	‘to sit at the table’
кивнуть	голова	S	ins	sg	None	151	ins	‘to nod’
протянуть	рука	S	acc	sg	None	149	acc	‘to give a hand’
играть	роль	S	acc	sg	None	142	acc	‘to play role’
открыть	глаз	S	acc	pl	None	138	acc	‘to open one’s eyes’
иметь	право	S	gen	sg	None	135	gen	‘(not) to have right’
иметь	дело	S	acc	sg	None	133	acc	‘to have business’
прийти	голова	S	acc	sg	в	130	acc,в	‘to cross one’s mind’
обращать	внимание	S	gen	sg	None	128	gen	‘(not) to pay attention’
иметь	значение	S	acc	sg	None	124	acc	‘to be significant’
быть	человек	S	gen	pl	None	124	gen	‘to be (no) people’
принять	участие	S	acc	sg	None	121	acc	‘to take part (pf)’
делать	вид	S	acc	sg	None	118	acc	‘to pretend’
принимать	участие	S	acc	sg	None	118	acc	‘to take part (impf)’
принять	решение	S	acc	sg	None	109	acc	‘to make a decision’
иметь	возможность	S	acc	sg	None	101	acc	‘to have an opportunity’

Table 1. The most frequent collocations

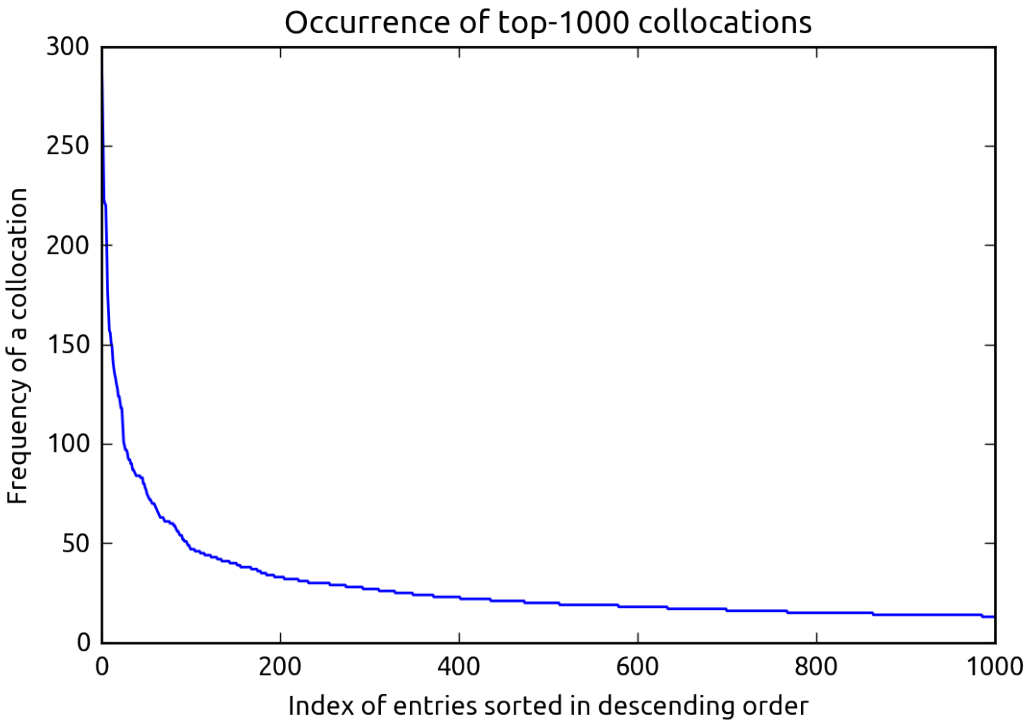

Fig. 1. Occurrence of the top-1000 collocations

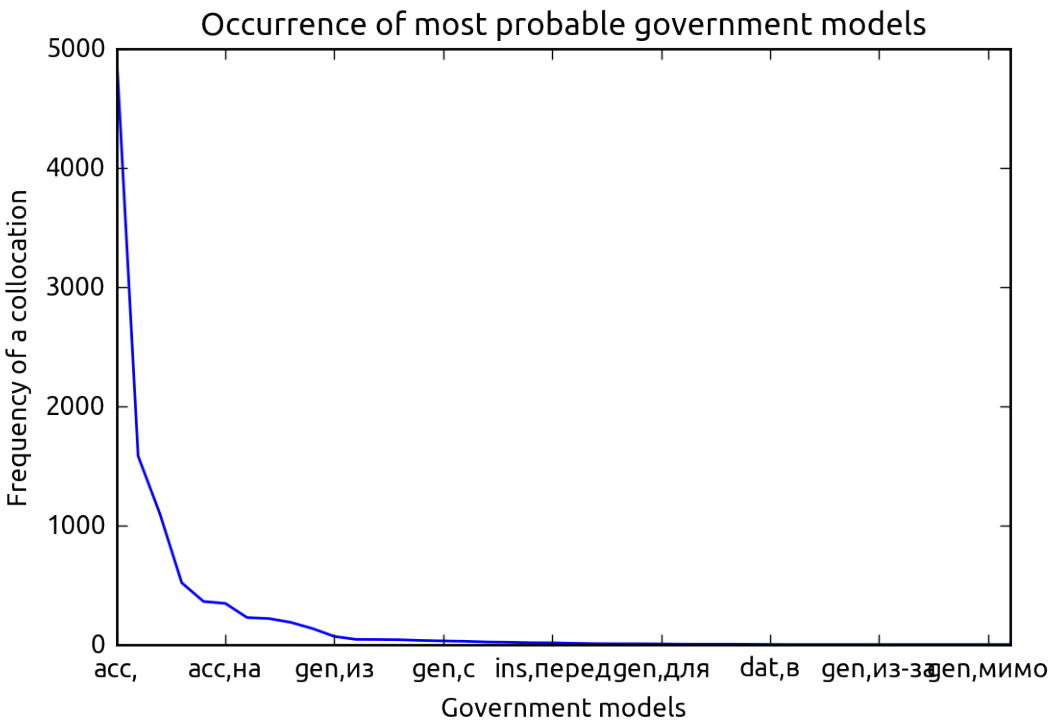

Fig. 2. Occurrence of most probable government models for 1000 verbs

Collocation	Model	POS	Preposition	Count	Translation
забить тревога	acc,sg	S	None	4	'to sound the alarm'
забить мяч	acc,sg	S	None	2	'to score'
забить гол	acc,sg	S	None	12	'to score'
забить раз	acc,sg	S	None	2	'to score... times'
забить козел	acc,sg	S	None	2	'to slaughter a goat'
забить голова	acc,sg	S	None	2	'to head a goal'
забить себя	dat	S-PRO	None	2	'to score an own goal '
забить мяч	gen,pl	S	None	2	'(not) to score'
забить мяч	gen,sg	S	None	3	'(not) to score'
забить смерть	gen,sg	S	до	2	'to beat dead'
забить гол	gen,sg	S	None	2	'to score'
забить автомобиль	ins,pl	S	None	2	'to obstruct with cars'
забить машина	ins,pl	S	None	2	'to obstruct with cars'
забить крыло	ins,pl	S	None	2	'to begin to flacker (of birds)'
забить вещь	ins,pl	S	None	2	'to fill up with stuff'
забить фанера	ins,sg	S	None	3	'to nail up with plywood'

Table 2. Collocations with the verb *забить 'to hit in'*

Model	Count	Model	Count	Model	Count
acc	4988	ins, над 'above'	43	ins, под 'under'	7
gen	1582	gen, в 'in'	41	gen, для 'for'	6
ins	1097	acc, за 'behind'	35	gen, вокруг 'around'	4
acc, в 'in'	519	gen, с 'with'	31	gen, без 'without'	4
dat	362	ins, за 'behind'	28	acc, сквозь 'through'	4
acc, на 'on'	346	gen, у 'at'	22	gen, против 'against'	3
ins, с 'with'	227	gen, на 'on'	20	dat, в 'in'	2
dat, к 'to'	219	acc, через 'through'	16	ins, со 'with'	1
gen, от 'from'	187	ins, перед 'in front of'	15	ins, между 'between'	1
dat, по 'on'	135	acc, о 'about'	11	acc, про 'about'	1
gen, из 'from'	69	acc, под 'under'	8	gen, мимо 'past'	1
gen, до 'till'	44	gen, о 'about'	7		

Table 3. Automatically derived government models for Russian verbs: frequencies

applying frequency filters to filter out noise proved efficient, and the results obtained from the experiment are reasonable from the linguistic point of view. This data can be used for language generation tasks (which was the primary idea behind this piece of research), as well as many other tasks mentioned earlier. The

Verb	Model	Verb	Model
вычислять ‘to calculate’	acc	сорвать ‘to pluck, to pick’	acc
демонстрировать ‘to demonstrate’	acc	обнаруживаться ‘to appear’	gen
дать ‘to give’	acc	подвязать ‘to tie up’	ins
свалиться ‘to fall’	acc, в ‘in’	покормить ‘to feed’	acc
являться ‘to appear’	gen	набрать ‘to gather, to recruit’	gen
получаться ‘to result’	gen	рассказывать ‘to tell’	dat
определяться ‘to take shape’	gen	учиться ‘to study’	gen
протянуть ‘to stretch’	acc	тыкнуть ‘to poke’	acc, в ‘in’
упасть ‘to fall’	acc, на ‘on’	предавать ‘to betray’	acc
клясться ‘to swear’	ins	подмигивать ‘to wink’	dat

Table 4. Automatically derived government models for Russian verbs: examples

source code in Python is available on GitHub: https://github.com/mamamot/vncollocations.

References

1. Orliac B., Dillinger M.: Collocation extraction for machine translation. Proceedings of Machine Translation Summit IX, New Orleans, LA, USA, 292–298 (2003)
2. Толдова С.Ю., Кустова Г.И., Ляшевская О.Н.: Семантические фильтры для разрешения многозначности в национальном корпусе русского языка: глаголы. Труды конференции «Диалог 2008», 522—529 (2008) // Toldova S., Kustova G., Lyashevskaya O.: Using Semantic Filters for Disambiguation in Russian National Corpus: Verbs
3. Merlo, P., Stevenson, S.: Automatic verb classification based on statistical distributions of argument structure. Computational Linguistics, 27(3), 373–408 (2001)
4. Кочеткова, Н. А.: Метод автоматической генерации модели управления глаголов русского языка. Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012 (16-20 октября 2012 г., г. Белгород, Россия): Труды конференции, Т. 1., 227 (2012) // Kochetkova N.: A Method of Automatic Government Model Generation for Russian Verbs
5. Акинина, Ю.С., Кузнецов, И.О., Толдова, С.Ю.: Влияние синтаксической структуры на извлечение коллокаций-существительных при глаголах. Труды конференции "Диалог-2013" (2013) // Akinina, Y., Kuznetsov I., Toldova, S.: The impact of syntactic structure on verb-noun collocation extraction
6. Zipf G.K.: Human Behavior and the Principle of Least Effort. Addison-Wesley Press, 484–490 (1949)
7. Гельбух А.: Разрешение синтаксической неоднозначности и извлечение словаря моделей управления из корпуса текстов. Материалы VIII Международной конференции KDS-99 (1999) // Syntactic Disambiguation and Model Dictionary Extraction from a Text Corpus
8. Tushkanov V., Drozd O, Davoian A. (2016, July) . Poster session presented at the Summer Neurolinguistics School at National Research University Higher School of Economics, Moscow, Russia.
9. Национальный корпус русского языка [Электронный ресурс]. / Электрон. дан. - Институт рус. яз. им. В. В. Виноградова РАН