Клячко Елена,

Факультет филологии, Компьютерная лингвистика, 2 курс

1. **Данные:**

   Часть подкорпуса интернет-корпуса[1] (5 тысяч предложений), созданного С. Шаровым.

2. **Получение графа:**

   Была выбрана модель синтаксической сети.

   Набор текстов был разбит на предложение и токенизирован С. Шаровым. Затем для каждого токена при помощи статистического морфологического анализатора TreeTagger[2] (модель для русского языка создана С. Шаровым[3]) получена наиболее вероятная лемма и морфологическая информация. После этого был проведен синтаксический анализ при помощи статистического анализатора MaltParser[4] (модель для русского языка создана С. Шаровым[5]). MaltParser проводит синтаксический анализ в формализме грамматики зависимостей: предложение представляется в виде дерева, в узлах которого — токены. На основе полученных для каждого предложения синтаксических деревьев был построен граф.

   Вершина графа — лемма + морфологическая информация. Какая именно морфологическая информация рассматривается, зависит от части речи леммы: для существительных рассматривается часть речи+число+падеж, для остальных частей речи — только сама часть речи. Так, для предложений «Вася видит стол» и «Вася видел стол» получим одинаковые наборы вершин: «Вася_N_n_s», «видеть_V», «стол_N_a_s». Для предложения «Стол стоит» получим набор вершин «стол_N_n_s», «стоять_V».

   В некоторых случаях лемматизатор мог определить часть речи, но не лемму. Тогда в качестве леммы выбиралась сама словоформа.

   Ребро между вершинами означает синтаксическую связь в одном из графов для предложений. Граф ориентированный (ребро направлено от главной вершины к зависимой) с кратными ребрами (кратность ребра = сколько раз такая связь встретилась в предложениях).

МОДУЛЬ 1

3. **Исследование свойств:**

   a. Количество узлов: 10889

      Количество ребер: 50033

[1] http://corpus.leeds.ac.uk/mocky/i-ru-sample.txt.gz

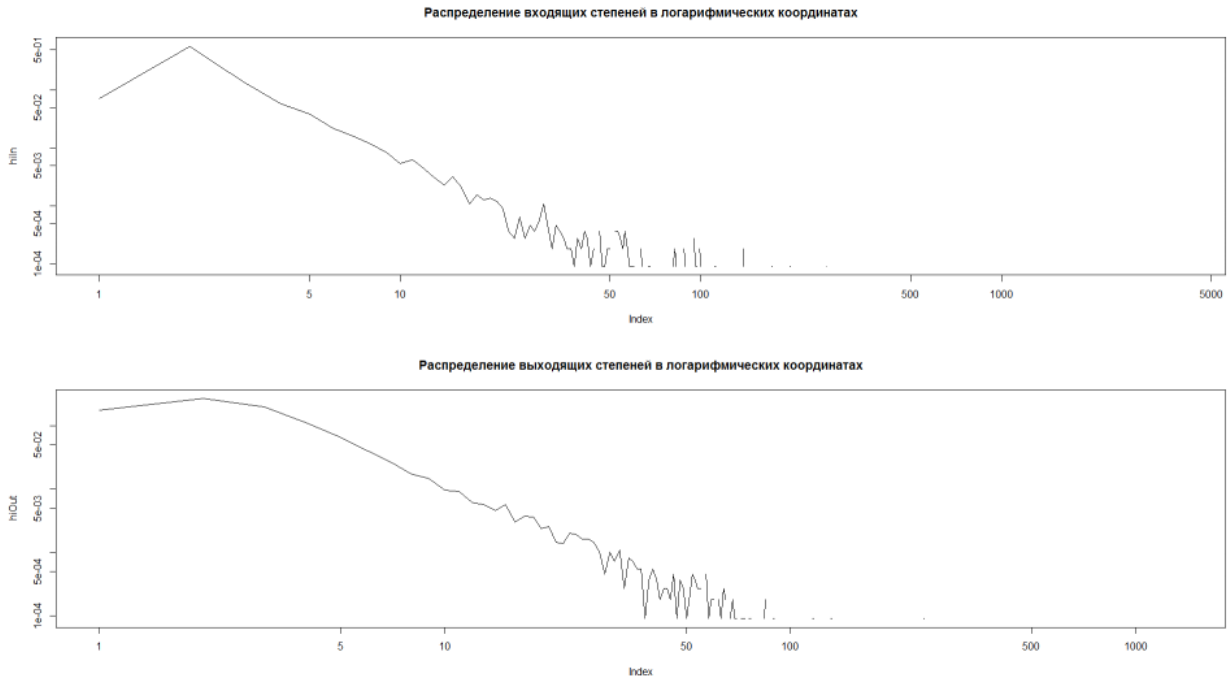
[2] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

[3] http://corpus.leeds.ac.uk/mocky/russian.par.gz

[4] http://maltparser.org/

[5] http://corpus.leeds.ac.uk/mocky/rus-test.mco

b. Распределение степеней вершин



Средняя степень узла: 4,59

c. При предположении, что распределение степенное:

   i. Входящие: α=1,52, p-value=0,99

   ii. Выходящие: α= 1,50 , p-value= 0,99

   Таким образом, нулевая гипотеза о том, что данные получены не степенным распределением, отвергается.

d. Граф имеет 5169 сильных компонент связности:

   i. 5720 узлов (в дальнейшем работа велась с этой компонентой)

   ii. Остальные — по одному узлу

e. Диаметр наибольшей компоненты связности: 26

   Средняя длина пути: 4,18

f. Коэффициент ассортативности: -0,13

g. Кластерный коэффициент: 0,02

   Примеры «треугольников»:

   i. глаголы, разными способами присоединяющие зависимое предложение

      ЗНАТЬ➔ЧТО, ЧТО➔БЫТЬ («Я знаю, что у него есть дом»)

      ЗНАТЬ➔БЫТЬ («Я знаю, у него есть дом»)

   ii. глаголы с несколькими аргументами

БЫТЬ➔В, В➔ВРЕМЯ («Это было в то время…»

БЫТЬ➔ВРЕМЯ («Было время…»)

iii. субстантивация

ДЛЯ➔БЕРЕМЕННЫХ, БЕРЕМЕННЫХ➔ЖЕНЩИН

ДЛЯ➔БЕРЕМЕННЫХ

iv. + ошибки синтаксического разбора или принятые в модели условности (например, зависимость идет от знака препинания)

h. Top-10 узлов по значениям:

Top-10 лемм для разных метриках представлены в таблицах ниже. Расчет делался отдельно без учета стоп-слов и с учетом стоп-слов. Стоп-слово: предлог, союз, частица, местоимение, знак препинания.

Таблица 1 Топ-10 лемм по степени узла

	Degree centrality (in) — все слова	Degree centrality (in) — без стоп-слов	Degree centrality (out)	Degree centrality (out) — без стоп-слов
1.	И_C	@CARD@_M	БЫТЬ_V	БЫТЬ_V
2.	В_S	БЫТЬ_V	И_C	МОЧЬ_V
3.	НЕ_Q	РЕЗЮМЕ_N_A_S	В_S	@CARD@_M
4.	НА_S	ГОСУДАРСТВЕННЫЙ_A	НА_S	СКАЗАТЬ_V
5.	Я_P	МОЧЬ_V	МОЧЬ_V	ЗНАТЬ_V
6.	@CARD@_M	РЕЗЮМЕ_N_N_S	ЧТО_C	РЕЗЮМЕ_N_A_S
7.	С_S	ОЧЕНЬ_R	С_S	ДОЛЖЕН_A
8.	ЧТО_C	ЕЩЕ_R	А_C	ЯВЛЯТЬСЯ_V
9.	_-	ОТБОР_N_G_S	У_S	ХОТЕТЬ_V
10	БЫТЬ_V	РАБОТА_N_G_S	@CARD@_M	МОЖНО_R

Таблица 2 Топ-10 лемм по метрикам closeness и betweenness

	Closeness centrality — все слова	Closeness centrality — без стоп-слов	Betweenness centrality — все слова	Betweenness centrality — без стоп-слов
1.	И_C	БЫТЬ_V	И	БЫТЬ_V

2.	БЫТЬ_V	МОЧЬ_V	В	@CARD@_M
3.	МОЧЬ_V	СКАЗАТЬ_V	НА	МОЧЬ_V
4.	В_S	МОЖНО_R	ЧТО	РЕЗЮМЕ_N_A_S
5.	ЧТО_C	ПРОЙТИ_V	ОНО	ДОЛЖЕН_A
6.	НО_C	ЗНАТЬ_V	НЕ	ЯВЛЯТЬСЯ_V
7.	СКАЗАТЬ_V	ГОВОРИТЬ_V	А	СТАТЬ_V
8.	А_C	НАЙТИ_V	ВЕСЬ	КАНДИДАТ_N_G_P
9.	КАК_C	СОСТОЯТЬ_V	СВОЙ	СКАЗАТЬ_V
10.	НА_S	ХОТЕТЬ_V	С	ОТБОР_N_G_S

Таблица 3 Топ-10 узлов по Page Rank

	Page Rank — все слова	Page Rank — без стоп-слов
1.	|_-	КОНТРАКТ_N_G_S
2.	КОНТРАКТ_N_G_S	КАНДИДАТ_N_G_P
3.	КАНДИДАТ_N_G_P	РАСПИСАНИЕ_N_A_S
4.	РАСПИСАНИЕ_N_A_S	ПОДПИСАНИЕ_N_N_S
5.	О_-	АНАЛИЗ_N_N_S
6.	ПОДПИСАНИЕ_N_N_S	@CARD@_M
7.	А_-	НАЙТИ_V
8.	И_C	ОБЛАДАТЬ_V
9.	;_-	ПОДХОДЯЩИЙ_A
10.	АНАЛИЗ_N_N_S	НОВЫЙ_A

4. **Сравнение со случайным графом Erdős–Rényi (и с характеристиками синтаксического графа из статьи Sole, Murtra** Language Networks: Their Structure, Function and Evolution[6])

	граф	случайный	характеристики синтаксического

[6] http://leonidzhukov.ru/hse/2013/lingnetworks/papers/LanguageNetworks.pdf

	(весь текст)	граф	графа [Sole, Murtra]
Средняя степень узла	4,59	9,19	5—10
Коэффициент кластеризации C	0,02	0,0008	
C/C_{rand}	~10		$\sim 10^3$
Средняя длина пути	4,29	6,26	3,5

МОДУЛЬ 2

1. Core structure

Таблица 4 Топ-10 узлов по значению максимального core (входящие ребра)

	in-core — все слова	значение core	in-core — без стоп-слов	значение core
1.	@card@_M	77	@card@_M	77
2.	-_-	77	быть_V	39
3.	._S	77	случай_N_l_s	35
4.	|_-	75	служба_N_a_s	29
5.	у_S	39	государственный_A	28
6.	что_C	39	база_N_l_s	26
7.	он_P	39	мочь_V	25
8.	в_S	39	данные_N_g_p	25
9.	то_P	39	база_N_a_s	25
10.	быть_V	39	помощь_N_i_s	23

Таблица 5 Топ-10 узлов по значению максимального core (выходящие ребра)

	out-core — все слова	значение core	out-core — без стоп-слов	значение core
1.	@card@_M	77	@card@_M	77
2.	|_-	75	быть_V	38
3.	что_C	38	мочь_V	38
4.	в_S	38	сказать_V	29

5.	но_C	38	знать_V	24
6.	то_P	38	должен_A	24
7.	быть_V	38	можно_R	23
8.	и_C	38	говорить_V	23
9.	мочь_V	38	делать_V	20
10.	о_S	34	пройти_V	19

2. Сетевые сообщества

метод	**модулярность**
walktrap.community	0,23
leading.eigenvector.community (рассматривались ребра без учета направления)	0,22

Методом walktrap.community получили 3406 сообществ. Среди них есть два больших сообщества (2558 и 1955 элементов), 5 сообществ с количеством элементов, меньшим 1955 и большим 100.

Примеры небольших сообществ:

- дитя_N_i_p, разрешение_N_g_s, отношение_N_g_p, **между_S**, страна_N_i_p, мяч_N_i_s, отскочившим_V, взаимоотношение_N_l_p, пол_N_i_p, вызвать_V, начало_N_l_s, перспектива_N_a_s, неудовлетворением_N_i_s, родитель_N_i_p, строка_N_a_p, конфликт_N_g_p, данные_N_i_p, конфликт_N_g_s, недоверие_N_a_s, пауза_N_n_s

- право_N_g_s, принцип_N_n_s, право_N_g_p, доступ_N_g_s, профессионализм_N_g_s, единство_N_g_s, лишение_N_l_s, признание_N_n_s

- около_S, профессионально_R, работник_N_n_p, отдел_N_g_p, число_N_n_s, nan_A, клерк_N_n_p, рабочий_N_n_p, чернорабочий_N_n_p, квалификационные_A, вес_N_l_s, инвестировала_V, 2000рублей_M, килобайт_N_g_p, укладываться_V

- импорт_N_g_s, обеспечиваться_V, выделение_N_g_s, выделение_N_n_s, коррекция_N_n_s, название_N_g_s, компания_N_g_p, период_N_g_s, коррекция_N_a_s, подсчет_N_a_s, автоматический_A

- входить_V, почта_N_d_s, письмо_N_a_p, почта_N_g_s, фильтрацию_N_a_s, посылать_V, уведомляющие_V, почта_N_n_s, разбиении_N_l_s, папка_N_g_p, Распознанные_A, просмотр_N_n_s, текст_N_d_s, новизна_N_l_s, адрес_N_a_p, sales@keystaff_N_n_s, электронный_A,

На рисунке ниже представлен фрагмент дендрограммы, построенной для графа из 4 предложений по результату метода edge betweenness (подписи — транслитерированные леммы)

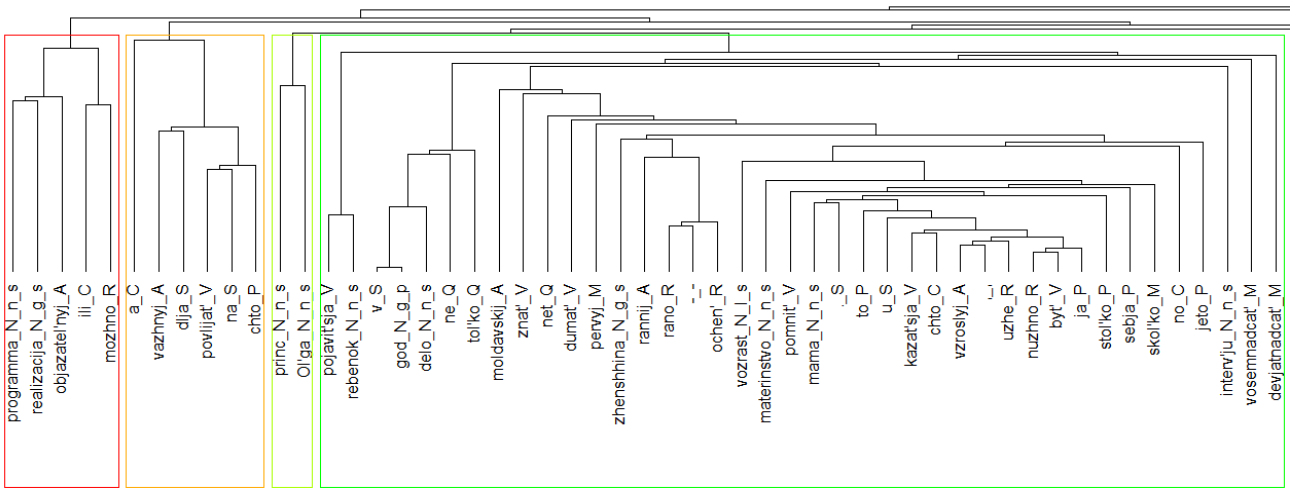


Рисунок 1 Фрагмент дендрограммы для 4 предложений

3. Мотивы:

Количества диад и триад показаны в таблицах

Таблица 6 Диады

тип диады	кол-во диад в синтаксическом графе	кол-во диад в случайном графе Erdos-Renyi
A⇔B	**764**	10
A➔B	48505	50013
A B	59230447	59229693

Таблица 7 Триады

тип триады	кол-во триад в синтаксическом графе	кол-во триад в случайном графе Erdos-Renyi	доля от общего числа в синтаксическом графе, %	доля от общего числа в случайном графе, %
A,B,C	214728779719([7])	214581948653([6])	99,815	99,747
A->B, C	334711157	543572422	0,156	0,253
A<->B, C	52107864	108715	**0,024**	0,000

[7] igraph возвращает для первого типа триад Nan. Это дефект, который пока не исправлен в официальных версиях (см. http://lists.gnu.org/archive/html/igraph-help/2013-05/msg00077.html). Поэтому данное число я рассчитала сама: (общее число троек) – (сумма всех остальных). Общее число троек = (число_вершин)*(число_вершин-1)*(число_вершин – 2)/6 = 215126089364

A<-B->C	1119319	115200	**0,001**	0,000
A->B<-C	7035037	114645	**0,003**	0,000
A->B->C	1794372	229458	**0,001**	0,000
A<->B<-C	237689	89	0,000	0,000
A<->B->C	244158	66	0,000	0,000
A->B<-C, A->C	26818	85	0,000	0,000
A<-B<-C, A->C	1689	31	0,000	0,000
A<->B<->C	22411	0	0,000	0,000
A<-B->C, A<->C	2237	0	0,000	0,000
A->B<-C, A<->C	2300	0	0,000	0,000
A->B->C, A<->C	3091	0	0,000	0,000
A->B<->C, A<->C	1378	0	0,000	0,000
A<->B<->C, A<->C	125	0	0,000	0,000

Примеры взаимных связей

- БЫТЬ➔И➔БЫТЬ («Был➔и➔будет»)
- ГОД➔ТЯЖЕЛЫЙ➔ГОД («Тяжелый год», «год тяжелых решений»)
- ИМЕТЬСЯ➔РЕЗЮМЕ➔ИМЕТЬСЯ («Имеющиеся резюме», «Резюме имеются»)

4. Структурное сходство

jaccard.similarity: сходство = 1		
Петербург_N_l_s	коллектив_N_l_s	знаки препинания, предлог «в»
акцент_N_i_s	трубка_N_i_s	знаки препинания, предлог «с»
осечься_V	лелеять_V	знаки препинания, союз «с»
0,7<сходство <1		

шепнуть_V	отступить_V	«и», «на» , «я»
0,5<сходство <=0,7		
ноябрь_N_n_s	март_N_n_s	@card@_M,декабрь_N_n_s,январь_N_n_s
правило_N_i_p	критерий_N_i_p	основной_A,отбор_N_g_s,являться_V
интервью_N_a_s	собеседование_N_a_s	кадровый_A,проводить_V
уточнить_V	ляпнуть_V	я_P, убить_V
0,4<сходство<=0,5		
рука_N_a_p	окно_N_a_s	в_S,за_S,на_S
раз_N_g_s_	час_N_g_s	@card@_M,два_M,три_M,четыре_M

На рисунке ниже приведена матрица сходства для графа из 10 предложений:

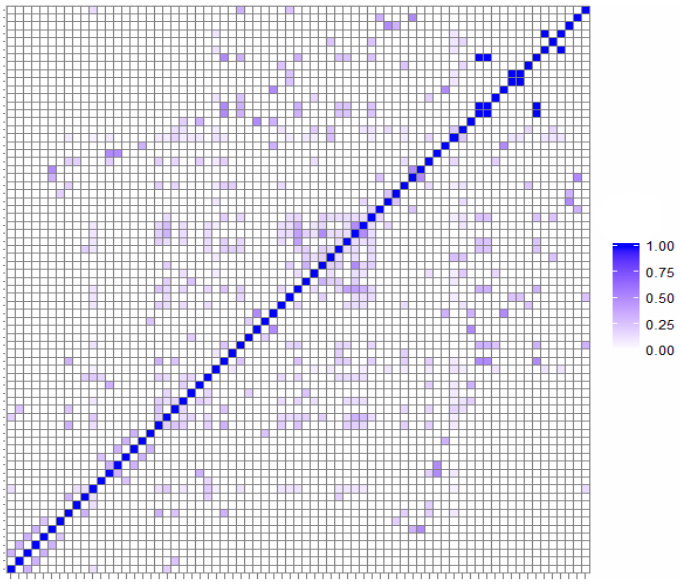


Рисунок 2 Матрица сходства для графа из 10 предложений

5. Визуализация

На рисунках представлены граф из четырех предложений и его фрагмент

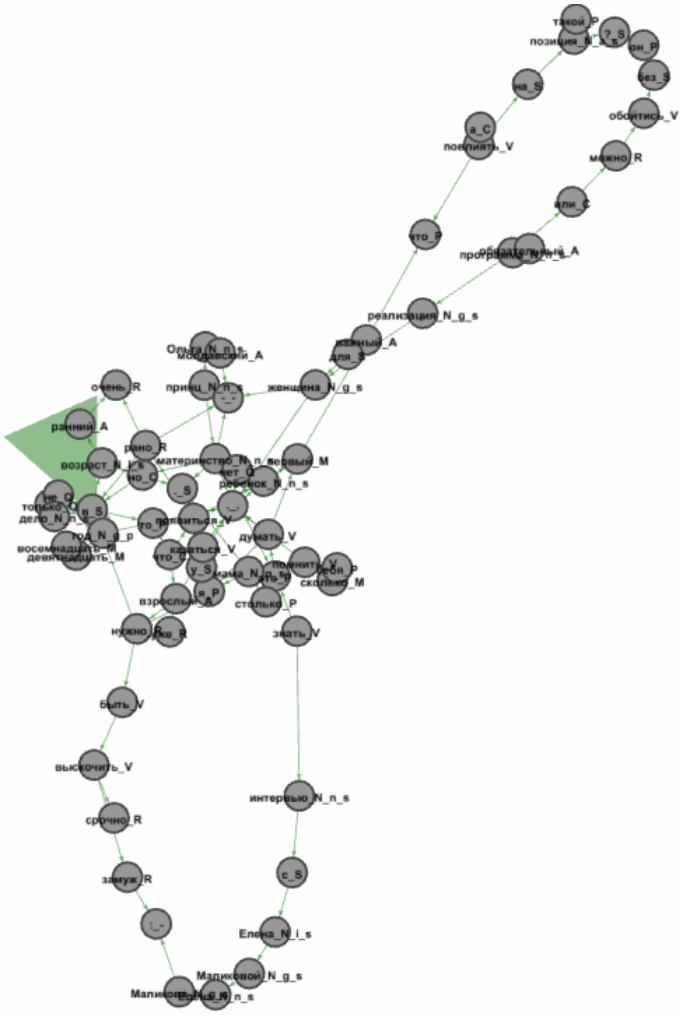


Рисунок 3 Граф (4 предложения)

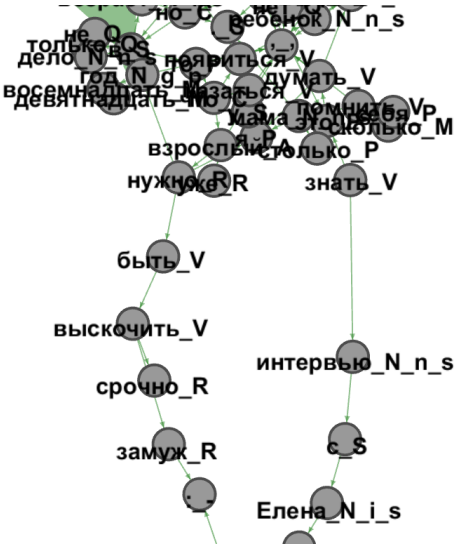


Рисунок 4 Фрагмент графа

ВЫВОДЫ

- Для синтаксического графа: closeness — ключевые предикаты, betweenness — ключевые имена
- Методами поиска сообществ можно получить группы семантически связанных слов
- Методами поиска структурного сходства можно получить синонимы/представителей одного таксономического класса. Однако для получения глаголов, имеющих сходную модель управления, нужен, видимо, больший граф