

Объектные модели представления окружающего мира

Немного фактов из истории

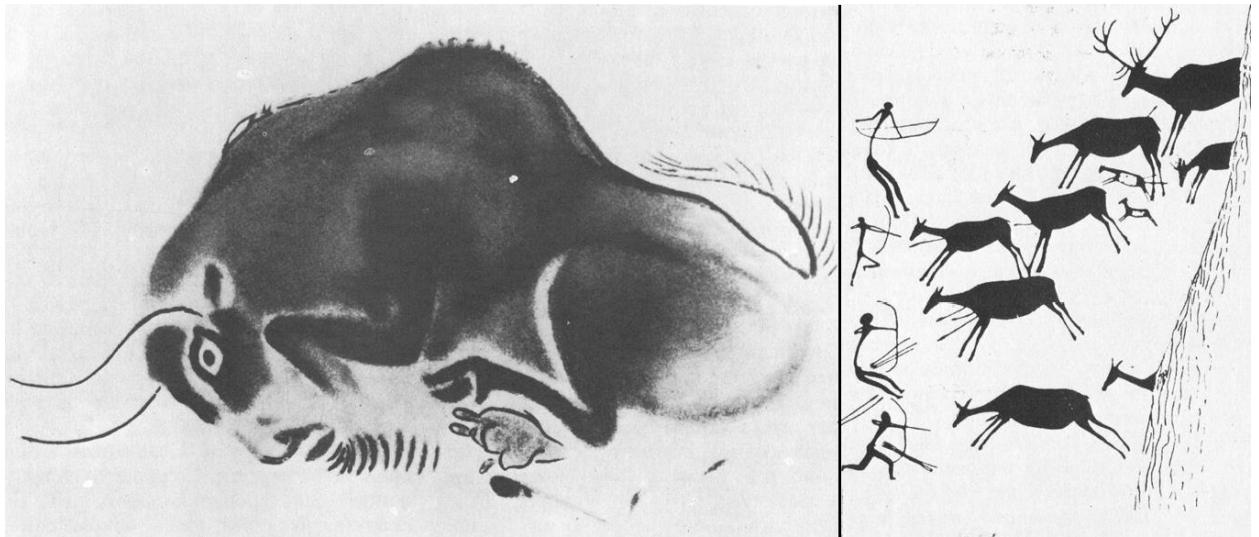


Рис.1 Восприятие человеком процесса охоты в палеолите(слева) и в неолите (справа)

Мышление человека на ранних этапах его развития было конкретно-образным. Но постепенно мышление стало приобретать новые черты. Объекты начали объединяться в группы на основе схожих свойств и качеств самих объектов. Это ускорило процесс добывания новых знаний, поскольку появились новые пути передачи знания.

В настоящее время

В настоящее время появляется все больше возможностей для взаимодействий типа объект-объект.

Технологические достижения последних десятилетий позволили расширить возможности реализации объектного подхода, поскольку появилась возможность хранить и передавать немислимые для человека объемы накопленных знаний, а также оперативно представлять знания в необходимых формах.

Модель Вещь-свойство-отношение в философии

Понятие «вещь» появляется достаточно рано в человеческой культуре, когда человек начинает отделять самого себя, свое сознание, от окружающей действительности. Можно сказать: вещь – это отдельный предмет, обладающий относительной независимостью и устойчивостью существования.

В истории философии понятие вещи было впервые четко сформулировано Аристотелем, который говорил о том, что вещь – это то, что обладает признаками и самостоятельно существует в пространстве и времени, но само не может быть ничьим признаком. И. Кант вводит понятие «вещь-в-себе», означающее, что мы познаем лишь те характеристики вещи, которые нам доступны в явлении. Таким образом, сущность вещи познаваема лишь относительно, через понимание нами ее свойств, каждое из которых связано с сущностью вещи как таковой.

Качество есть такая определенность вещи, утрачивая которую вещь перестает существовать, переходя из бытия в небытие. Количественные же характеристики вещи могут до известной степени изменяться, но сама вещь при этом сохраняет свою качественную определенность. Причем любая вещь взаимосвязана с совокупностью других вещей, т. е. является элементом более крупной системы, в той или иной мере приобретаая и так называемые системные качества.

В мире человеческой культуры мы сталкиваемся с особым рядом вещей, которые несут идеально-информационное содержание. Такая вещь, созданная человеком и включенная в мир человеческого общения и совместной деятельности, носит название символа или знака в самом широком смысле.

Связь – это взаимообусловленность существования явлений, разделенных пространственными или временными характеристиками. Познание вещи есть познание свойств самой этой вещи, обусловленных системой связей, в которые она объективно включена и которые нами исследуются в настоящий момент времени.

Связи могут быть внутренними и внешними. Внутренняя связь – это структура предмета, т. е. совокупность его внутренних связей. Она обеспечивает его целостность и устойчивость, т. е. качество. Но поскольку предмет или вещь не находятся в вакууме, то они испытывают влияние со стороны других предметов или вещей. Соответственно внутренняя структура зависит от внешних воздействий и может изменяться под их прямым влиянием.

Само определение вещи претерпело значительную эволюцию. Оно приобретало все более абстрактное содержание: от понятия вещи как внешнего тела или предмета через аристотелевское понятие самостоятельности существования – до разделения его на семантическое (или символическое), онтологическое и гносеологическое определение. В онтологическом плане вещь – это любой носитель признаков. В гносеологическом – любой объект мысли. В семантическом – нечто, что

может быть обозначено или названо, т. е. имеет идеально-информационное измерение.

Рассматривая гносеологическую модель “Вещь-свойство-отношение” важно отметить, что не любой объект мысли может реально существовать в окружающем нас мире.

Примеров этому имеется множество: “Ковер-самолет”, “Скатерть - самобранка”, “Машина времени”. Такие объекты могут выражать идеалы, быть предметом веры, являться мечтами, фантазиями или галлюцинациями.

Следует заметить, что такие объекты обычно имеет аналогии в мире вещей. Например, “Ковер-самолет” это ковер, который умеет летать. Также стоит отметить, что в окружающем мире существуют только конкретные объекты. Например, существует конкретный кот Наседкина Игоря, а кота “вообще” не существует. Т.е. не существует кота, который будет наиболее полно раскрывать сущность понятия “кот” и будет лишен индивидуальных черт.

Искусственный интеллект является продуктом науки. Во всех научных парадигмах, которые были раньше и в той, которая господствует сегодня, наука изучает те явления, которые либо относятся к реальным объектам окружающего мира, либо их можно отнести к таковым. Исключение составляют некоторые гуманитарные науки, такие как философия.

Для реализации этой парадигмы необходимо провести границу, разделяющую объекты мысли на те, что реально существуют в окружающем нас мире (т.е. реальные) и на те, что не могут существовать в окружающем нас мире (т.е. идеальные). Тут уже приходится ввести новое понятие концепт.

Концепт- слово в конкретном значении. Например Геракл(Геркулес) – был и является мифическим героем Древней Греции, а цемент “Геркулес” продается в строительных магазинах в Греции.

Всем вещам присущи некоторые общие признаки, которые отражаются в таких категориях, как «качество и количество», «сущность и явление», «общее и единичное» и т. д. И здесь важно понимать, как среди признаков вещи различить свойство и отношение. По Аристотелю, свойством называется отдельный признак, который принадлежит одному носителю. Отношением называется отдельный признак, который принадлежит нескольким носителям. Таким образом, отношение как бы связывает все вещи по какому-то типу отношения, а свойство, напротив, их обособляет, выделяя из других вещей.

Итак, природные вещи представляют собой материальные образования, включенные в относительно устойчивые системы движения. Вещи, благодаря

такому движению, воздействуют друг на друга. Взаимодействие вещей порождает у них ряд новых свойств и одновременно выявляет относительную самостоятельность вещей. Точно так же и символы культуры как идеально-материальные образования всегда включены в систему человеческих отношений, проявляя только в этом процессе свои имманентные качества и только в нем обретая новые системные свойства.

Модель VSO для представления текста

Модель VSO основывается на гносеологической модели “Вещь-свойство-отношение”

Вещь обычно представляет собой объект окружающего мира. Обычно в предложении соответствующие объекты выражаются следующими частями речи: существительные, местоимения, имена собственные, аббревиатуры. Это выполняется когда в предложении есть глаголы.

Свойства представляют собой характеристики конкретного объекта-вещи или действия-отношения. Обычно в предложении они выражаются следующими частями речи: прилагательные, причастия (являются свойствами объектов), наречия (являются свойствами действия).

Отношения представляют собой взаимосвязи между объектами-вещами. В предложении соответствующие объекты выражаются глаголами и деепричастиями. Например, в предложении: «Я иду в кино» слово «иду» является подчинительной связью между объектами-вещами.

Рассматриваемая модель модели VSO.

Модель может быть представлена в виде связного ориентированного графа, являющегося деревом. Таким образом, последовательность предложений можно представлять либо в виде леса из деревьев, либо в виде конкретного дерева.

Основной вопрос заключается в связи между соседними деревьями.

Например, для предложения “Каждый охотник желает знать, где сидит фазан”, можно представить следующим образом (рис.2)

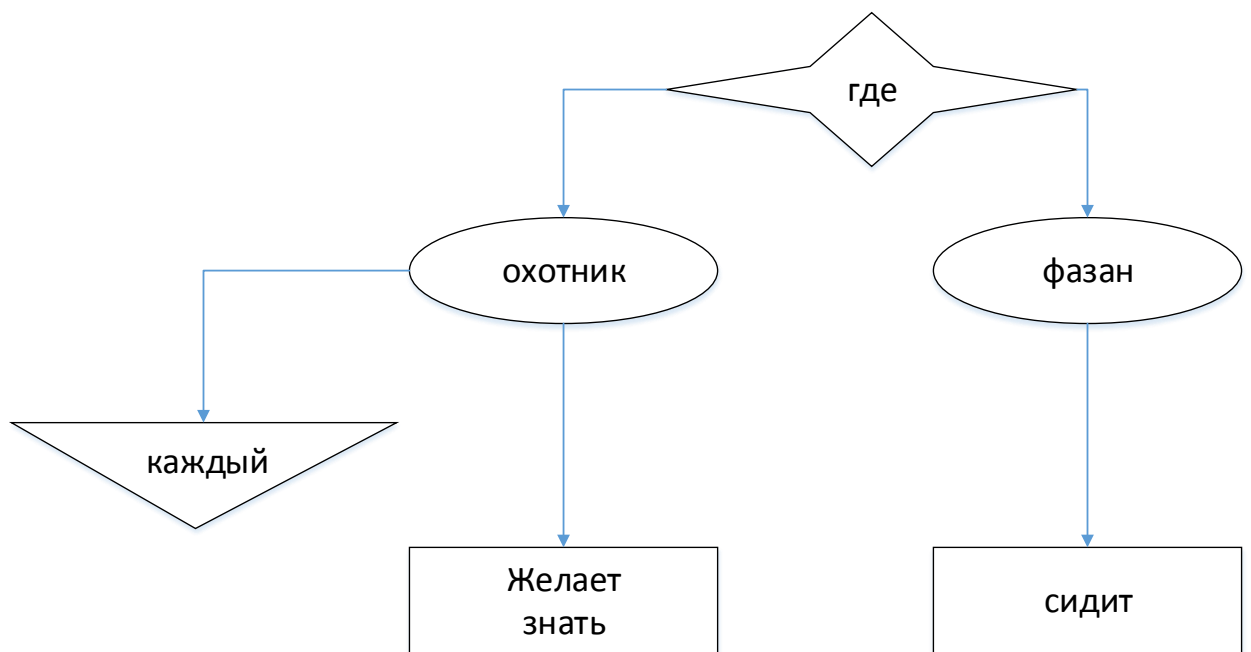


Рис.2

Это же предложение можно представить в виде:

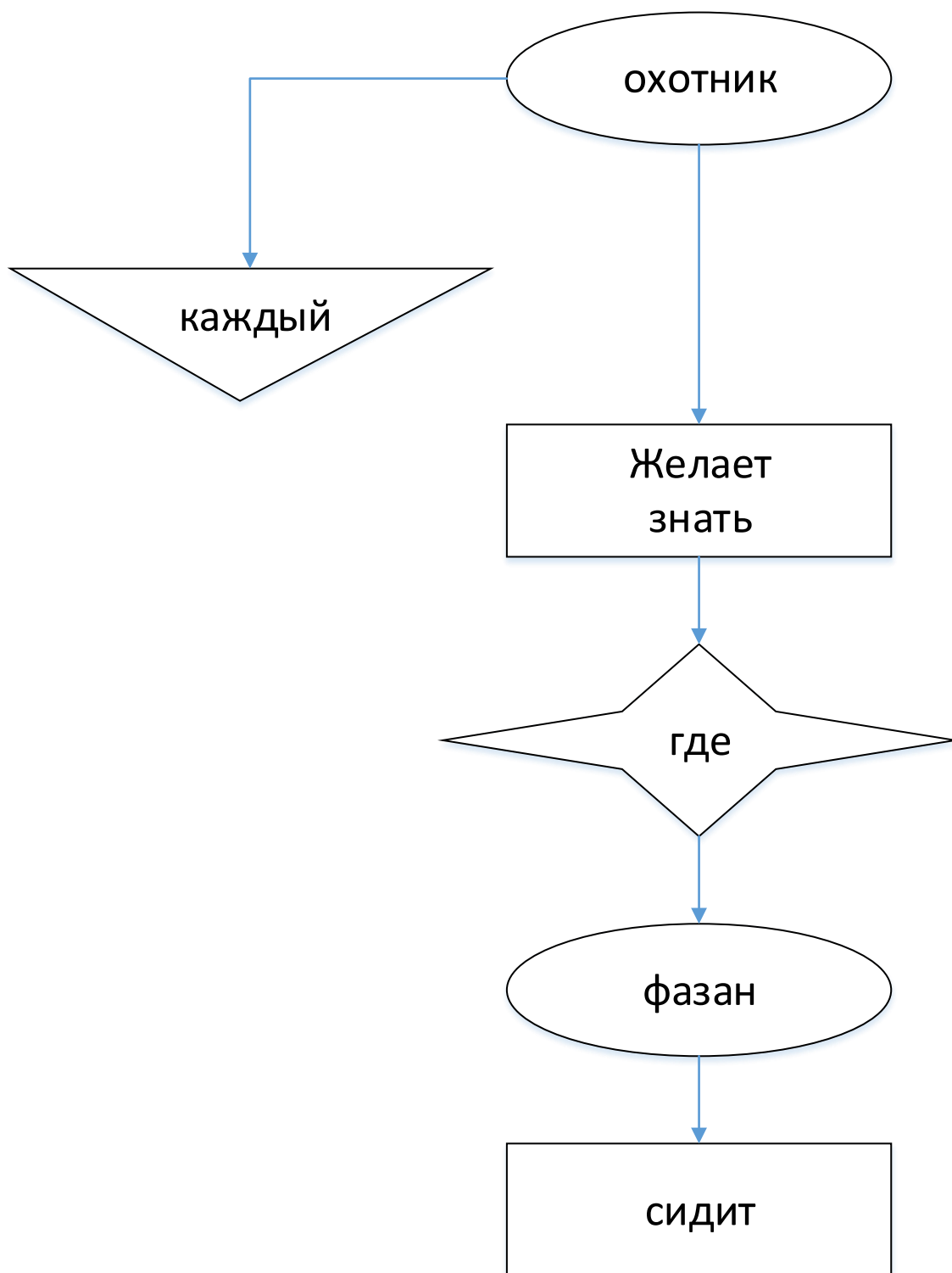


Рис.3

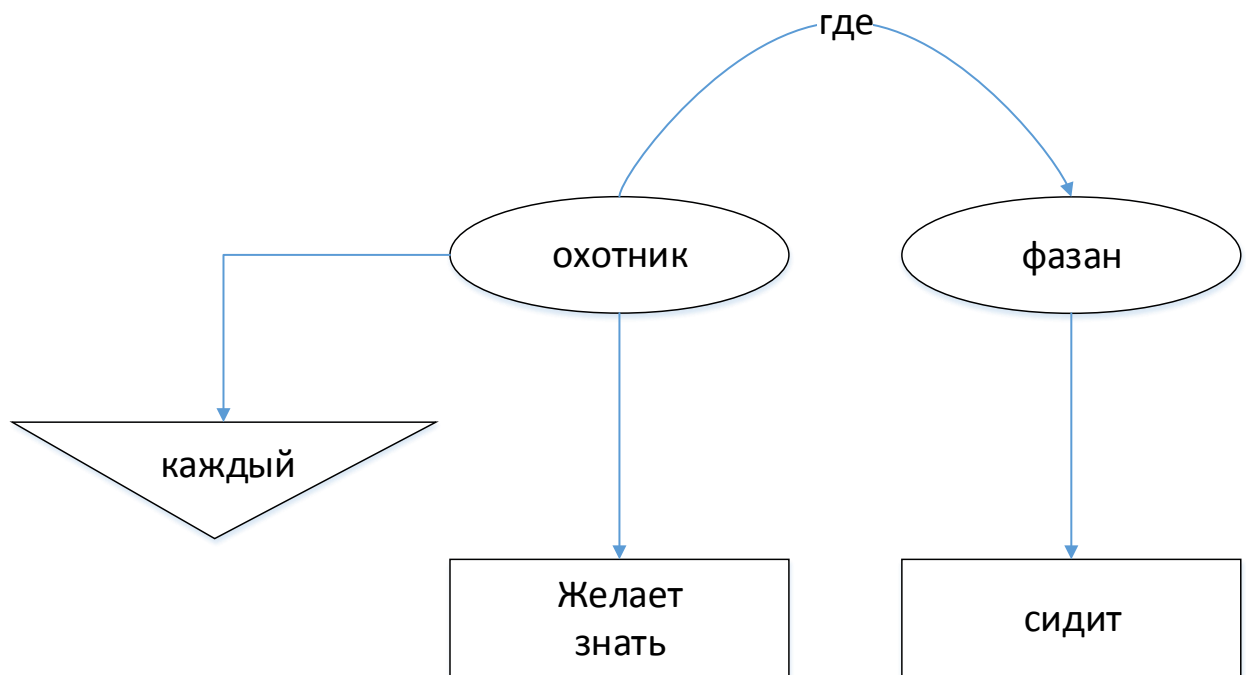


Рис.4

Можно по-разному рассуждать, пытаюсь определить, какая из моделей верна. Но для определенности будем использовать первую модель.

Пример: “отгорел закат, и полная луна облила лес зеленоватым, призрачным серебром”

[[', '], ['отгорел', 'verb'], ['закат', 'noun'], [',и', ''], ['полная', 'noun'], ['луна', 'noun'], ['облила', 'verb'], ['лес', 'noun'], ['зеленоватым', 'adj'], ['призрачным', 'adj'], ['серебром', 'noun']]

3

[['закат', 'отгорел']]

['0 закат obj', '1 отгорел act']

7

[['луна', 'облила'], ['полная', 'облила']]

['0 луна obj', '0 полная obj', '1 облила act', '2, лес obj', '2, серебром obj', '2, призрачным prop', '2, зеленоватым prop']

Проблема реального и идеального.

Особенности объектной модели.

Как было отмечено ранее, в модели VSO используются 3 вида объектов: вещь (объект), свойство (характеристика объекта) и отношение (действие, который один объект совершает над другим объектом, возможно на самом собой).

Рассмотрим каждый из этих видов объектов более подробно.

Модели, построенные на основе объектов

Модели представляют собой иерархическое дерево, в котором каждому объекту поставлен в отношение другой объект, который наилучшим образом его обобщает.

Примерами такого обобщения служат толковые словари.

Кокос- орех на пальме

Следующие виды отношений:

1)элемент-система

сборка-разборка

2)фрагмент-сущность

3) член-множество

4) кусок-целое

5) компонент-смесь

Отношение целое-часть

Бывает случаи, когда использование такой модели приводит к возникновению логических противоречий. Например, имеется предложение:

“Художник может быть один, а человек – никак”.

Если объекты “художник” и “человек” связаны отношением включения, то возникает следующий вопрос. Пусть понятие “художник” обобщает понятие “человек”, то наличие отношения художник-может быть-один означает, что некоторые люди могут быть одни, но в последующей части предложения это категорически отрицается.

Отсюда можно сделать вывод, что одно из понятий (“художник” или “человек”) употреблены в значении, которое отсутствует в модели.

Проблема “синего банана”.

Так сложилось, что подавляющее число людей представляют банан как фрукт желтого цвета. Действительно, желтый банан продается в магазинах, на рынках и в многих других местах.

При обучении сети на корпусе текстов в настоящее время используется следующий подход: собирается статистика слов, которые наиболее часто встречаются рядом. Предполагается таким способом узнать свойства объектов реального мира.

Как говорилось ранее, при слове “банан” возникает образ фрукта желтого цвета. Поэтому вместо “желтый банан” в текстах употребляется слово “банан” без указания цвета. Однако, когда обозначается синий банан, в тексте встречается словосочетание “синий банан”.

Описание объектов и их основных свойств есть в словарях на естественном языке. Например, в словаре Ожегова: “Высокое тропическое растение с большими листьями, а также его удлинённый и слегка изогнутый сладкий мучнистый плод, растущий в соплодии.”. Но у этого определения имеется ряд недостатков:

Во-первых, в нем не указан самый распространенный цвет банана, а также все его возможные цвета в реальном мире.

Во-вторых, в определении фигурируют 2 объекта: растение банан, и сам фрукт банан.

Однако, даже если известны все возможные цвета банана в реальном мире и среди них выделен желтый цвет, как наиболее распространенный, этого недостаточно для глубокого обучения, но вполне достаточно для поверхностного.

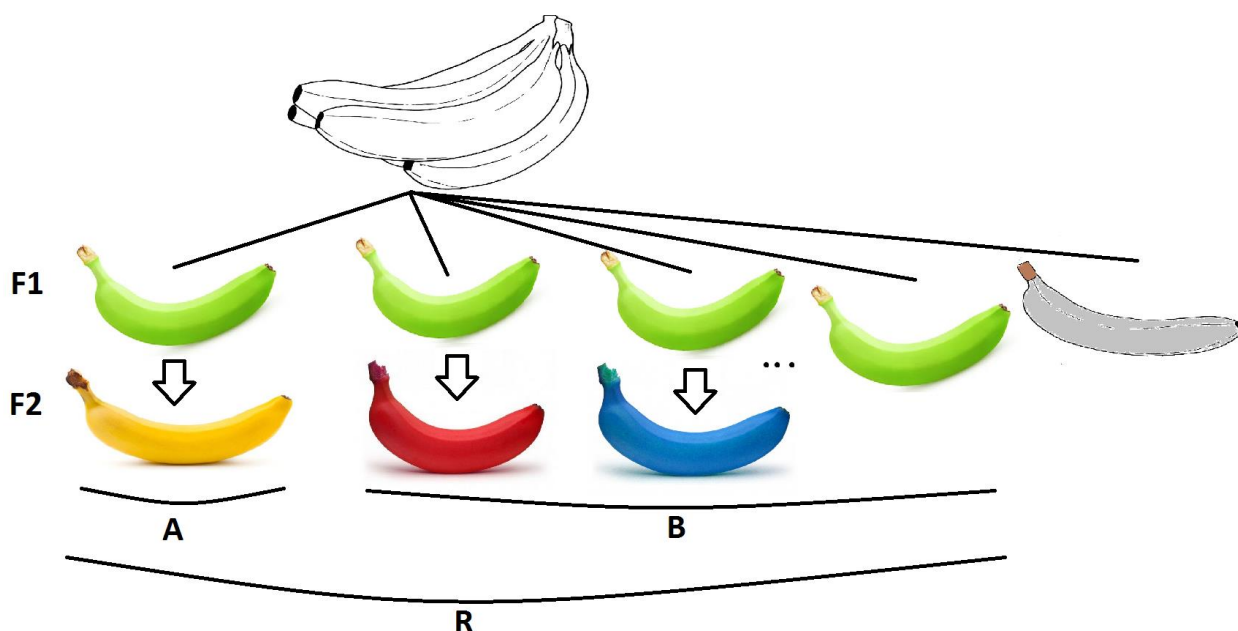


Рис.1 Свойство “цвет” у банана

На рисунке 1 бананы поделены на группы по свойству “цвет”.

В группе R находятся бананы, которые реально выращивают.

В группе A находится желтый банан. Его используют в еду ежедневно.

В группе B находятся все реально выращиваемые экзотические бананы. Их подают в ресторанах, они используются на свадьбах, юбилеях и т.д. Также они используются блогерами для привлечения аудитории.

Также стоит отметить, что банан имеет 2 фазы: “спелую” и “неспелую”. В неспелой фазе банан зеленый. Однако, существует сорт зеленых бананов, которые зеленые вне зависимости от спелости.

Игра на арфе пальцами ног.

В реальном мире существуют нормальные отношения между объектами. Например, арфу держат одной рукой, а другой рукой на ней играют. Возьмем следующий пример из осетинского эпоса:

“Нарт Батрадз рожден железнотелым, и потому он приходит к божественному кузнецу Курдалагону с просьбой закалить его [11:269]. Тот кладет героя в горнило и закаляет в течение нескольких месяцев, однако Батрадз жалуется, что ему холодно и от скуки просит арфу (фандыр) поиграть. Затем он играет на ней *пальцами ног*, змеи засыпают, их пережигают на уголь ...”

И из ирландского:

“Связанного Гуннара живьем опускают в ров со змеями, вместе с конунгом кладут и его арфу, и вождь играет на ней *пальцами ног* – ведь его руки связаны. Погибает он от укуса змеи в печень.”

Задача заключается в определении вероятности того, что такое отношение возможно. Действительно, инструкция по игре на арфе есть, например, на сайте WikiHow (<https://ru.wikihow.com/играть-на-арфе>).

Рассмотрим более подробно инструкцию.

Она состоит из 5 частей:

- Типы арф
- Покупка арфы
- Сидение и положение рук
- Настройка арфы
- Обучение

Сами разделы уже можно разделить по степени важности.

Во-первых, близким по значению к слову “игра” является слово “обучение”. Поэтому часть “Обучение” будет важна. Кроме того, как отмечалось ранее, объект “рука” и объект “нога” являются “братьями”, так как отцовским объектом для них является объект “конечность” (рис. 2)

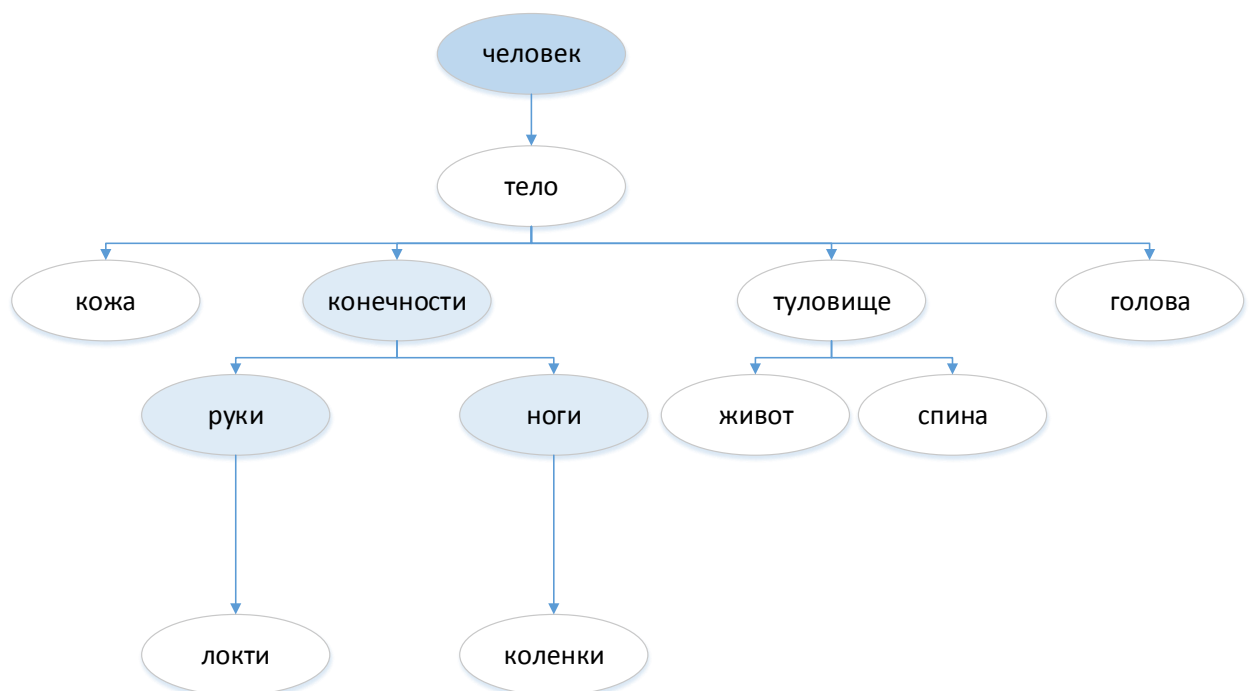


Рис.2 Упрощенная объектная модель “человек”

Так как объекты “рука”, “конечность”, “коленка” являются наиболее близкими к объекту “нога”, они с большей вероятностью будут иметь похожие свойства и назначение, что и объект “нога”.

Поэтому пункт “Положение рук” также важен.

В подпункте “Сидение за арфой” есть “ловушка”

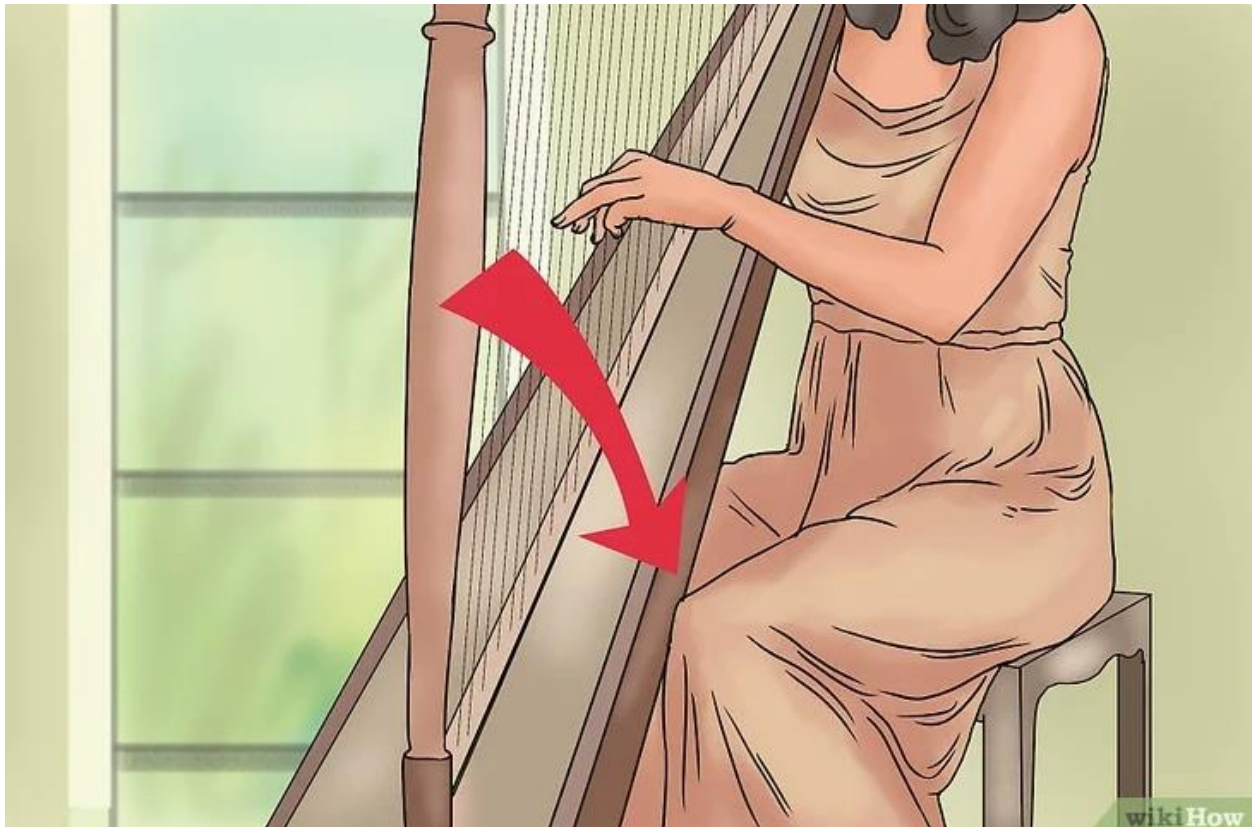


Рис. 3 Положение тела и арфы

Рисунок 3 является иллюстрацией к фразе: “Корпус арфы поставьте между ваших ног, наклоните его и положите на правое плечо”.

Теперь рассмотрим предложения, в которых встречаются слова “рука” и “нога”:

Корпус арфы поставьте между ваших **ног**, наклоните его и положите на правое плечо.

Вы должны сидеть так, чтобы ваши **руки** находились под углом чуть меньше 90 градусов к вашему телу, были параллельны полу и располагались посередине струн. В этом случае ваши **ноги** должны находиться на полу.

Положение **рук** – наиболее спорный вопрос среди арфистов. Нет общепринятого положения **рук**, которое подходило бы всем арфистам. Есть некоторые сходства в технике, например, частое расслабление **рук**, но они подчиняются здравому смыслу и помогают предотвратить травмы.

Теперь вытяните ваш указательный палец на правой **руке**.

Палец может быть как у руки, так и у ноги, поэтому найдем предложения отдельно:

Начните с игры двумя **пальцами**, затем перейдите к игре тремя пальцами, и только потом учитесь играть четырьмя **пальцами**.

Большинство преподавателей также делают упор на том, что вы должны прижимать пальцы к ладони после проигрывания ноты.

На большинстве арф (педальных и рычажных) играют мягкой частью пальцев (подушечками) – большого пальца и трех средних (т.к. мизинец слишком мал).

Теперь пометим каждый подпункт скалярной величиной, которая будет обозначать вероятность нахождения подпункта в контексте для каждого слова в отдельности.

Проблема глубины

Определение части речи слов в предложении

Пусть каждое слово в предложении соответствует одному из следующих значений.

'S': 'сущ.',
'A': 'прил.',
'NUM': 'числ.',
'A-NUM': 'числ.-прил.',
'V': 'глагол.',
'ADV': 'нареч.',
'PRAEDIC': 'предикатив',

'PARENTH': 'вводное',
'S-PRO': 'местоим. сущ.',
'A-PRO': 'местоим. прил.',
'ADV-PRO': 'местоим. нареч.',
'PRAEDIC-PRO': 'местоим. предик.',
'PR': 'предлог',
'CONJ': 'союз',
'PART': 'частица',
'INTJ': 'межд.'

В итоге получается, что каждое слово в русском языке в определенном контексте(т.е. концепт) должно принадлежать одному из 16 классов выше.

Метод определения части речи слова с помощью рекуррентной нейронной сети LSTM

Идея:

Русский язык обладает следующей особенностью: слова и их морфологические признаки согласуются между собой. Например, у подлежащего и сказуемого согласуется время, число и род (в прошедшем времени). У существительного и зависимого от него прилагательного – род, число и падеж и так далее.

Пусть имеется слово в предложении.

- 1) Возьмем суффикс и окончание самого слова в предложении.
- 2) поставим этому слову в соответствие вектор #размерности 6 по таблице.
- 3) Возьмем 4 следующих слова, если исходное слово находится ближе к началу предложения.
- 4) Для этих слов определим исходные данные
- 5) На основании этих данных поставим в соответствие каждому слову вектор размерности 6

В пункте 1 мы сначала пытаемся найти окончание слова по имеющимся у нас шаблонам. Если произошло несколько совпадений, возьмем самое длинное окончание. Далее удаляем это окончание из слова и тем же способом ищем самый длинный суффикс в оставшейся части слова.

В пункте 2 мы сначала поставим найденному окончанию слова число в диапазоне от 0 до 4. Трехкратно повторенное число и будет первыми 3-мя компонентами вектора. По таблице 1 и найденному суффиксу слова - число в

диапазоне от 0 до 4 по таблице 2. Трехкратно повторенное число и будет 4-й, 5-й и 6-й компонентами вектора.

Таблица 1 Соответствие окончаний слов цифрам

0	1	2	3	4
свойство	Свойство или объект	объект	Объект или действие	действие
еми	ах	е	у	ат
ех	ая	её	и	ёт
ие	ей	ию	им	ёте
ий	ем	ми	а	ёшь
оё	ею	ов	ом	ит
ую	ого	ам		ите
ёх	ое			ишь
емя	ас			м
мя	ами			ут
ум	ими			шь
умя	их			ст
ух	о			
	ой			
	ому			
	ою			

Таблица 2 соответствия суффиксов слов цифрам

0	1	2	3	4
свойство	Свойство или объект	объект	Объект или действие	действие
ал	ан	ани	ть	я
ел	ян	яни	л	ка
аст	ств	анин	а	ева
ат	ик	янин	е	ова
ев	ин	ач	и	ыва
ов	ист	ени	ть	ива
еват	тель	ет		нича
оват	к	еств		ну
ен	ни	есть		ствова
енн	либо	ец		ся
онн	нибудь	изм		ти

енск	то	изн		
инск		ник		
ив		итель		
лив		их		
чив		ниц		
ит		лк		
овит		льник		
н		льщик		
шн		льщиц		
тельн		от		
уч		ость		
юч		ун		
яч		ур		
чат		ура		
б		чик		
ене		чиц		
ск				
ов				
иан				
ий				
ейш				
айш				
жды				
о				
учи				
ючи				

В пункте 3 мы возьмем 4 следующих за словом слова.

Все последующие за словом слова известны нам по таблицам синтаксического корпуса русского языка. Проблема омонимии в данном случае не мешает, поскольку слова-омонимы, если они относятся к разным частям речи, представлены во всех вариантах и разбор этих слов представлен во всех соответствующих таблицах. В таблице 3 показано, что произвольное слово можно преобразовать в 6-мерный числовой вектор, каждой компонентой которой служит число от 0 до 2.

В таблице 3 имеются фиксированные цифры, обозначенные цифрами. На месте символа “х” может быть число от 0 до 2.

Таблица 3. Используемый способ представления слова в виде 6-мерного
вектора

предполагаемый тип сущности	Часть речи	признаки				№
"свойство"	Прилагательные	пол	падеж	число	время	1
"свойство"	Наречия	степень	полярность	-	-	2
"свойство"	числительные	пол	падеж	одушевленность	-	3
"вещь"	Местоимения	одушевленность	пол	число	падеж	4
"вещь"	Местоимение	лицо	пол	число	падеж	5
"вещь"	Существительные	одушевленность	пол	число	падеж	6
"вещь"	Имена	одушевленность	пол	число	падеж	7
"вещь"	Аббревиатуры	-	-	-	-	8
"отношение"	Глаголы	лицо	число	пол	время	9
"отношение"	Глаголы	лицо	число	пол	время	10
"отношение"	Союз подчинительный	-	-	-	-	11
"отношение"	Соединительные союзы	-	-	-	-	12
"отношение"	частица (PART)	-	-	-	-	13
"отношение"	междометия	-	-	-	-	14
"отношение"	Знаки препинания	-	-	-	-	15
"отношение"	ADP	-	-	-	-	16
условное обозначение признака						№
0	0	x	x	x	x	1
0	1	x	x	0	0	2
0	2	x	x	x	0	3
1	0	x	x	x	x	4
1	0	x	x	x	x	5
1	1	x	x	x	x	6
1	1	x	x	x	x	7
1	2	0	0	0	0	8
2	0	x	x	x	x	9
2	0	x	x	x	x	10
2	2	0	0	0	0	11
2	2	0	0	0	0	12
2	2	1	0	0	0	13
2	2	1	0	0	0	14
2	2	2	0	0	0	15
2	2	2	0	0	0	16

После преобразований для каждого из слов у нас получилось 5 векторов размерности 6.

На вход нейронной сети подадим вектор, полученный в пункте 2 и 5 векторов, полученных в предыдущем пункте.

На выходе нейронной сети мы должны получить вектор размерности 6.

В этом векторе первая координата означает предполагаемый тип сущности в модели VSO. Вторая координата – предполагаемую часть речи слова. Остальные координаты выходного вектора не представляют интереса для распознавания части речи слова.

Зная часть речи слова, можно в дальнейшем найти его признаки (такие как род, число, падеж) исходя уже из правил русского языка, а не полагаясь на качество обучения нейронной сети.

Описание нейронной сети:

Выбранная архитектура нейронной сети состоит из следующих слоев:

Слой LSML

Слой Dense

Слой LSTM

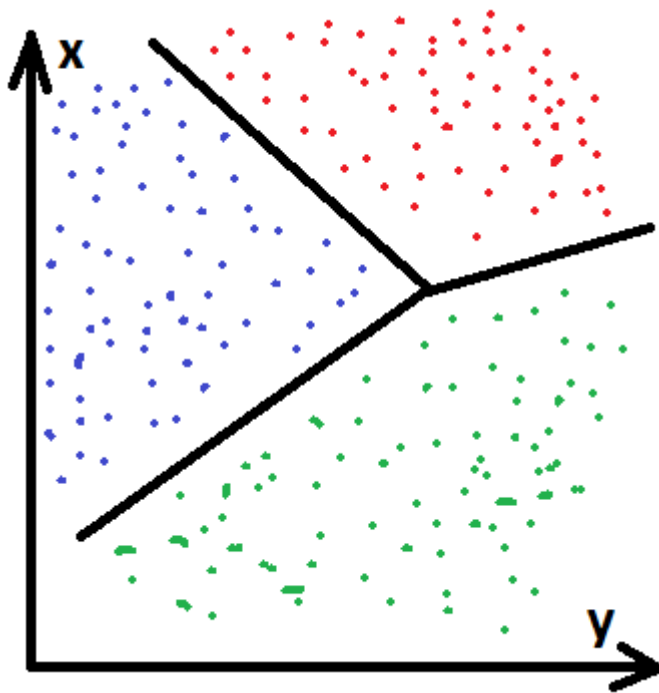
Входные данные слоя представляют собой матрицу размерности 5x6.

Выходные данные слоя представляют собой матрицу 1x100.

Линейные классификаторы

Когда перед нами стоит задача разделить множество объектов на m классов с непересекающимися элементами, мы можем представить каждый объект точкой в n -мерном пространстве, далее разместить все точки в n -мерном пространстве и провести не более, чем m гиперплоскостей (размерности $n-1$). На рисунке 1 показан случай 2-мерного пространства и 3 классов.

Рис.1 Линейное разделение объектов в 2-мерном пространстве на 3 класса.



Суть метода опорных векторов:

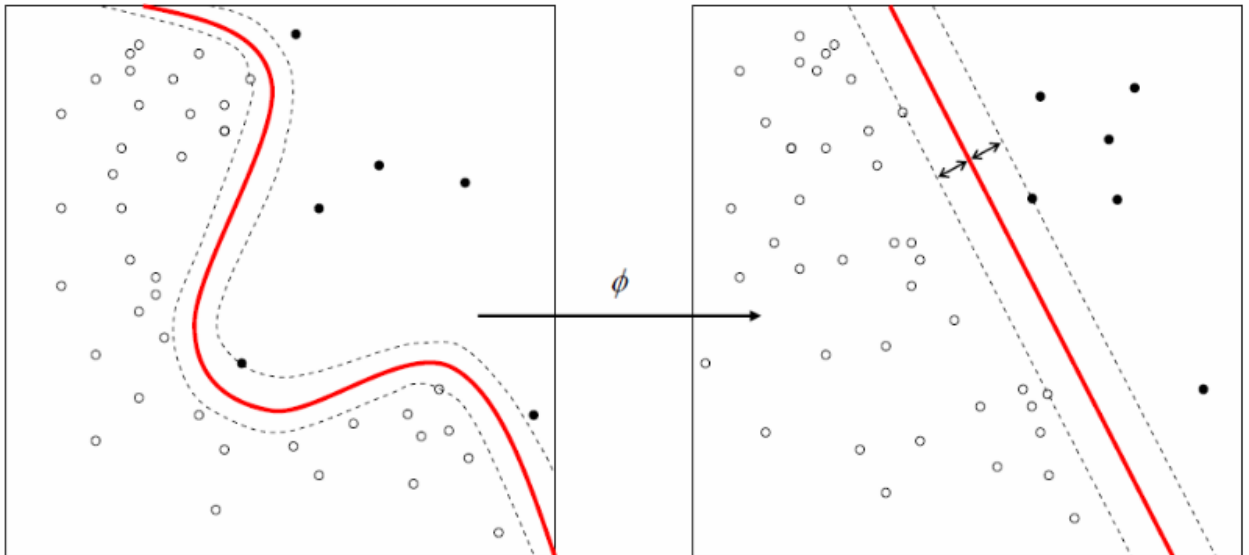
Метод опорных векторов нужен, когда разделение объектов на классы линейными классификаторами дает большой процент ошибок.

Пусть имеется набор входных векторов (x_1, x_2, \dots, x_m) и набор выходных векторов (y_1, y_2, \dots, y_m) ,

y_i принадлежит $\{1, 2, \dots, N\}$

причем каждому x_i соответствует y_i . Необходимо подобрать такую функцию аппроксимации, которая разделит все объекты таким образом, что каждый x_i попадет в свой класс, соответствующий y_i .

Не ограничивая общности рассуждений, положим, что $N=2$. Тогда задача сводится к задаче бинарной классификации.



На рисунке выше показан случай, показывающий, что линейные классификаторы не всегда эффективны для задачи бинарной классификации. Идея заключается в увеличении размерности пространства поиска.

Описание метода опорных векторов:

Значения целевой переменной:

$$t_1 \dots t_N \quad t_j \in \{-1, 1\}$$

Функция принятия решений:

$$y(x) = w^T \Phi(x) + b$$

$\Phi(x)$ – функция перехода из исходной системы координат в спрямленную.

Разделяющая плоскость:

$$y_i(x_i)t_i > 0$$

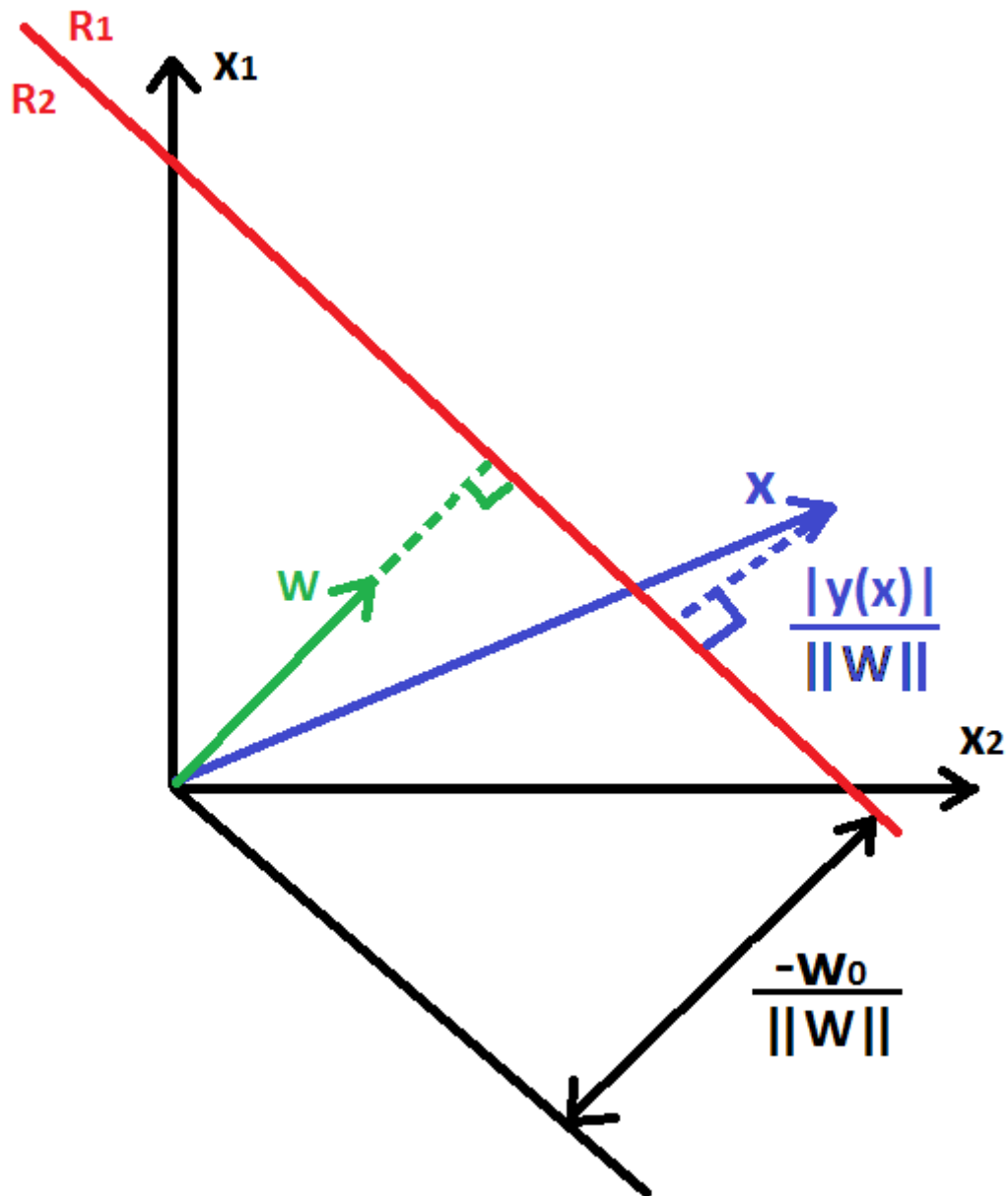
$$\text{Или } \text{sign}(y_i(x_i)) = \text{sign}(t_i)$$

Это означает, что правильный ответ и полученное нейронной сетью значение лежат в одной полуплоскости относительно разделяющей плоскости

Случай линейной разделимости:

В этом случае разделяющая плоскость является гиперплоскостью по отношению к размерности исходного пространства ($\dim(f(x))=N-1$)

Рис.х



На рисунке х видно, что для любой точки, лежащей на оси x

$$d_j = \frac{|y_j(x)|}{\|w\|}, j \in 1:M$$

Таким образом:

Т.к. $y_i(x_i)t_i$ и $t_j \in \{-1,1\}$:

$$d_j = \frac{|y_j(x)|}{\|W\|} = \frac{t_j y_j(x)}{\|W\|} = \frac{t_j (w^T \Phi(x) + b)}{\|W\|}$$

Оптимальная расчетная плоскость находится из максимизации минимального расстояния от точки на оси x до разделительной плоскости.

$$\operatorname{argmax}_{w,b} [\min_j d_j] = \operatorname{argmax}_{w,b} \left[\frac{1}{\|W\|} \min_j t_j (w^T \Phi(x) + b) \right]$$

$$d_j = \frac{t_j (w^T \Phi(x) + b)}{\|W\|}$$

Пусть $t_j (w^T \Phi(x) + b) = \text{const}$. Тогда разделим дробь для d_j на const и

$$t_j (w^T \Phi(x) + b) = 1$$

Тогда задача сводится к оптимизации системы:

$$\operatorname{Max} \left(\frac{1}{\|W\|} \right) = \operatorname{min} (\|W\|^2)$$

$$\begin{cases} \|\mathbf{w}\|^2 \rightarrow \min \\ c_i (\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1, \quad 1 \leq i \leq n. \end{cases}$$

Заменяя $\|W\|^2$ на $\frac{1}{2} \|W\|^2$ получим:

$$\begin{cases} \frac{1}{2} \|W\|^2 \rightarrow \min \\ c_i (w x_i - b) \geq 1 \end{cases}$$

Метод множителей Лагранжа:

- Составим функцию Лагранжа в виде **линейной комбинации** функции f и функций φ_i , взятых с коэффициентами, называемыми **множителями Лагранжа** — λ_i :

$$L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i \varphi_i(x),$$

где $\lambda = (\lambda_1, \dots, \lambda_m)$.

- Составим систему из $n + m$ уравнений, приравняв к нулю **частные производные** функции Лагранжа $L(x, \lambda)$ по x_j и λ_i .
- Если полученная система имеет решение относительно параметров x'_j и λ'_i , тогда точка x' может быть условным экстремумом, то есть решением исходной задачи. Заметим, что это условие носит необходимый, но не достаточный характер.

В нашем случае получим:

$$L(w,b,a) = \frac{1}{2} \|W\|^2 - \sum_{j=1}^N a_j (d_j - 1) = \frac{1}{2} \|W\|^2 - \sum_{j=1}^N a_j (t_j (w^T \Phi(x) + b) - 1)$$

Полученное уравнение нужно продифференцировать по w, b и каждому a_j . Дифференцировать по a_j не имеет смысла, потому что это наши ограничения.

$$W = \sum_{j=1}^N a_j t_j \Phi(x_j),$$

$$0 = \sum_{j=1}^N a_j t_j$$

Теорема Куна-Таккера (достаточное условие экстремума):

Необходимые условия минимума функции [\[править | править код \]](#)

Если $\hat{x} \in \arg \min f$ при наложенных ограничениях — решение задачи, то найдётся вектор множителей Лагранжа $\lambda \in \mathbb{R}^m$ такой, что для [функции Лагранжа](#)

$$L(x) = f(x) + \sum_{i=1}^m \lambda_i g_i(x) \text{ выполняются условия:}$$

- **стационарности:** $\min_x L(x) = L(\hat{x})$;
- **дополняющей нежёсткости:** $\lambda_i g_i(\hat{x}) = 0, \quad i = 1 \dots m$;
- **неотрицательности:** $\lambda_i \geq 0, \quad i = 1 \dots m$.

Двойственная задача поиска седловой точки функции Лагранжа:

Седловая точка

Определение:

Пара векторов x^*, y^* называется седловой точкой функции Лагранжа $L(x, y)$, если:

$$L(x^*, y) \leq L(x^*, y^*) \leq L(x, y^*)$$

По теореме Куна-Таккера:

По [теореме Куна — Таккера](#) эта задача эквивалентна двойственной задаче поиска [седловой точки функции Лагранжа](#)

$$\begin{cases} \mathbf{L}(\mathbf{w}, \mathbf{b}; \lambda) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^n \lambda_i (c_i ((\mathbf{w} \cdot \mathbf{x}_i) - b) - 1) \rightarrow \min_{w,b} \max_{\lambda} \\ \lambda_i \geq 0, \quad 1 \leq i \leq n \end{cases} \quad (2)$$

где $\lambda = (\lambda_1, \dots, \lambda_n)$ — вектор двойственных переменных.

Сопряженная задача:

$$\tilde{L}(a) = \sum_{j=1}^N a_j t_j \Phi(x_j)^T \Phi(x_j) \rightarrow \max_a$$

При условиях:

$$a_j \geq 0 \quad \forall j \in 1 \dots N$$

$$\sum_{j=1}^N a_j t_j = 0$$

Применение решения:

$$Y(x) = W^T \Phi(x) + b = \sum_{j=1}^N a_j t_j \Phi(x_j)^T \Phi(x_j) + b = \sum_{j=1}^N a_j t_j k(x_j, x) + b$$

{Условия Куна-Таккера}

$$a_j > 0$$

$$t_j y(x_j) - 1 \geq 0$$

$$a_j (t_j y(x_j) - 1) = 0$$

Из последнего выражения следует:

$$\begin{cases} a_j = 0 \\ t_j y(x_j) - 1 \geq 0 \end{cases} \quad \begin{cases} a_j > 0 \\ t_j y(x_j) - 1 = 0 \end{cases}$$

$$b = \frac{1}{N_s} \sum_{i \in S} (t_i - \sum_{j \in S} a_j t_j k(x_i, x_j))$$

Неразделимый случай:

Переменные $\varepsilon_j \geq 0$

$$\varepsilon_j = \begin{cases} 0, & \text{если } y(x_j) t_j \geq 1 \\ |t_j - y(x_j)|, & \text{иначе} \end{cases}$$

Задача оптимизации:

$$C \sum_{j=1}^N \varepsilon_j + \frac{1}{2} \|w\|^2 \rightarrow \min_{w,b}$$

Тут возможны 2 варианта:

Можно либо стремиться все $\varepsilon_j = 0$, либо максимального расстояние до разделяющей поверхности.

Сопряженная задача:

$$\tilde{L}(a) = \sum_{j=1}^N a_j - \frac{1}{2} \sum_{j=1}^N a_i a_j t_i t_j \Phi(x_i)^T \Phi(x_j) \rightarrow \max_a$$

При условиях:

$$0 \leq a_j \leq C \quad \forall j \in 1 \dots N$$

$$\sum_{j=1}^N a_j t_j = 0$$

При этом:

$a_j = 0$ – правильно проклассифицированные объекты

$a_j = C$ – опорные векторы внутри отступа

$0 < a_j < C$ – опорные отступы на границе

Классификация

ФПР:

$$y(x) = \sum_{j=1}^N a_j t_j k(x_j, x) + b$$

Константа b:

$$b = \frac{1}{N_\mu} \sum_{i \in M} (t_i - \sum_{j \in S} a_j t_j k(x_i, x_j))$$

Функции ядра

Вычисление функции перехода происходит столько раз, какова размерность пространства, в которое осуществляется переход. Однако при вычислении функции k можно существенно сократить число операций, избежав вычисления $\Phi(x)$:

$$k(x_i, x_j) = \Phi(x_i)^T \Phi(x_j)$$

Теорема Мерсера

Функция $k(x,z)$ является ядром \Leftrightarrow она:

- Симметрична $k(x,z) = k(z,x)$
- Неотрицательно определена $\int_{x \in X} \int_{z \in X} k(x,z)g(x)g(z)dx dz \geq 0 \quad \forall g(x): X \rightarrow R$

Стандартные функции ядра:

Линейное ядро	$k(x,z) = x^T z$
Полиномиальное ядро	$k(x,z) = (x^T z + r)^d$
Radial Basis Function	$k(x,z) = e^{-\gamma x-z ^2}$
Sigmoid	$k(x,z) = \tanh(\gamma x^T z + r)$

53:02

Преимущества SVM:

Не линейная разделяющая поверхность

Глобальная оптимизация

Разреженное решение

Хорошая обобщающая способность

Недостатки SVM:

Не поддерживает $p(C_k | x)$

Чувствительность к выбросам

Нет алгоритма выбора ядра

Медленное обучение

Реализация методов на языке python

Подготовка обучающей выборки

Для подготовки обучающей выборки был взят размеченный корпус русского языка ru-syntagrus, состоящий из 48000 предложений русского языка. В каждом предложении для каждого слова даны его признаки. На рисунке 4 приведен фрагмент этого файла:

Рис.4 фрагмент файла ru-syntagrus

```
# sent_id = 2003Anketa.xml_1
# text = Анкета.
1  Анкета  анкета  NOUN      _      Animacy=Inan | Case=Nom | Gender=F
2  .      .      PUNCT    _      punct  1:punct _

# sent_id = 2003Anketa.xml_2
# text = Начальник областного управления связи Семен Еремеевич
1  Начальник  начальник  NOUN      _      Animacy=Anim | Case=Nom |
2  областного  областной  ADJ       _      Case=Gen | Degree=Pos | Gender
3  управления  управление  NOUN      _      Animacy=Inan | Case=Gen |
4  связи  связь  NOUN      _      Animacy=Inan | Case=Gen | Gender=F
5  Семен  семен  PROPN     _      Animacy=Anim | Case=Nom | Gender=M
6  Еремеевич  еремеевич  PROPN     _      Animacy=Anim | Case=Nom |
7  был  быть  AUX       Aspect=Imp | Gender=Masc | Mood=Ind | Number
```

На этом этапе каждому слову в предложении ставится в соответствие 6-мерный вектор по таблице 2.

Рис.4. Пример предложения, записанного в файл

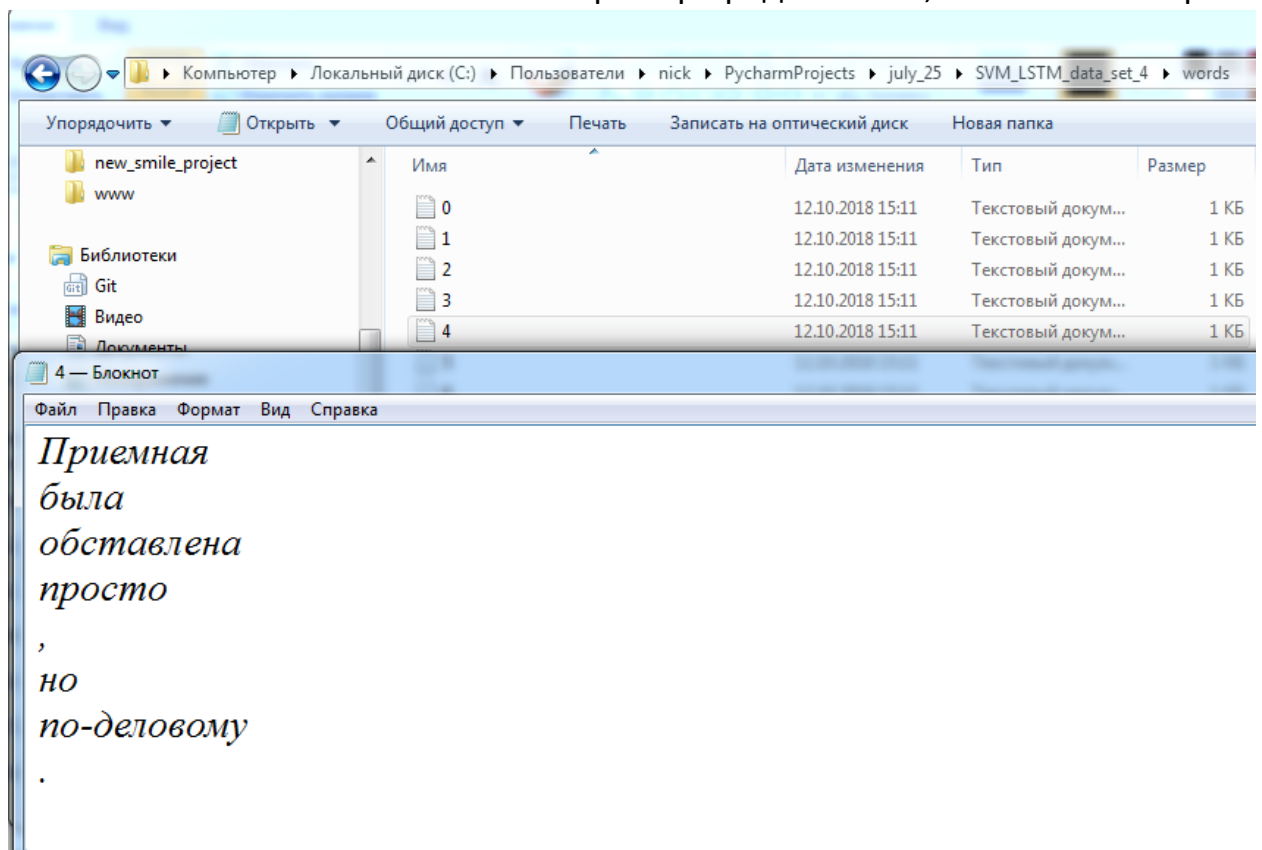
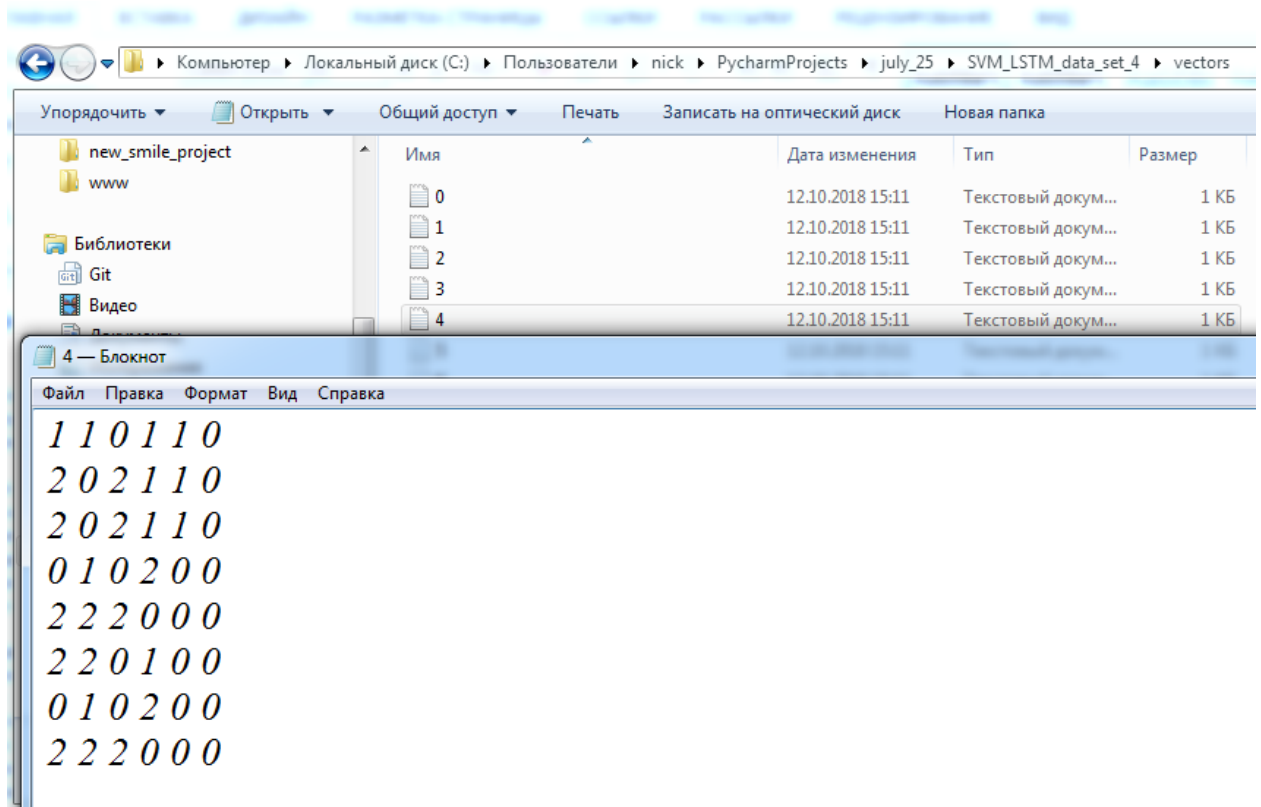


Рис.5. То же предложение, записанное в виде последовательности векторов



Итак, у нас получилось 48000 текстовых файлов, каждый из которых содержит последовательность векторов.

На следующем шаге мы проходимся по каждому из полученных текстовых файлов. Для каждой последовательности векторов проходим от первого вектора и до 2-го с конца. На каждом шаге цикла составляем последовательность из предыдущего слова, самого вектора слова и 3 последующих слова. Недостающие слова заменяем специальным вектором: "2 2 2 2 2 2". Далее записываем в текстовый файл само слово и последовательность.

Полученная структура имеет следующий вид (рисунок 3)

1	была
2	1 1 0 1 1 0
3	2 0 2 1 1 0
4	2 0 2 1 1 0
5	0 1 0 2 0 0
6	2 2 2 0 0 0
7	

Для удобства загрузки обучающей выборки в нейронную сеть все полученные текстовые файлы группируются по первому вектору. Итоговая файловая структура представлена на рисунке 6.

Рис.6. Итоговая файловая структура

1	СТОЛ
2	0 0 2 0 1 2 1)
3	1 1 0 2 1 0 2)
4	2 2 2 1 0 0 3)
5	0 0 2 2 0 2 4)
6	1 1 0 1 0 2 5)
7	

Загрузка обучающей выборки перед обучением

Для обучения нейронной сети была взята тестовая обучающая размером в 30000 последовательностей. Из каждой последовательности берется вектор №5, затем №4, затем №3,

затем само слово переводим в вектор (шаг 2). Затем берем вектор №1 и записывается последовательность векторов в двумерный массив. Вектор № 2 представляет собой результирующий вектор. (номера векторов отмечены на рисунке 6).

Обучение нейронной сети

Для обучения нейронной сети подадим обучающую выборку на 1-й слой LSTM. Обучим сеть на 20 эпохах, затем выполним последующий пункт проверки результатов. Повторим этап загрузки данных, обучения нейронной сети и проверки результатов еще 10 раз, каждый раз дообучая модель нейронной сети. Это необходимо для решения 2-х проблем: во-первых, недообучения, во-вторых, переобучения.

Проверка результатов

Осуществим проверку на 1000 тестовых данных, которые были взяты из других файлов, чем обучающая выборка.

В результате получим 1000 векторов размерности 6. Для каждой пары – правильный вектор- результат обучения вычислим модуль разности между первой и второй компонентами этих векторов. Если обе разности не превосходят 0.25, то распознавание прошло успешно. Иначе - была произведена ошибка.

Результаты

Итоговая точность обучения составляет в районе 70%.

Формирование обучающей выборки для сети SVM

На вход сети SVM поступает 3 числа – число, обозначающее суффикс слова, и число, обозначающее окончание слова, полученные ранее на шаге 2. В качестве 3-го числа примем 1-ю компоненту вектора, полученного на предыдущем шаге. В качестве правильного результата возьмем 1-ю компоненту правильного вектора.

Обучение нейронной воли.

Исходный код нейронной сети представлен на рисунке 7.

Рис.7. код нейронной сети SVM

```
def SVC_sklearn(X, y):
    X_train=X[0:6000]
    X_check=X[6000:6999]
    y_train = y[0: 6000]
    y_check = y[6000:6999]
    model=SVC()
    model.fit(X_train,y_train)
    print(model)
    expected=y_check
    predicted=model.predict(X_check)
    print("expected=\n")
    print(expected)
    print("\n")
    print("predicted=\n")
    print(predicted)
    print(metrics.classification_report(expected,predicted))
    print(metrics.confusion_matrix(expected,predicted))
    return list(predicted)
```

В результате обучения получается следующая таблица (таблица 4)

Анализ результатов

Эксперименты показали, что наибольшая точность приходится на центральный класс или класс “вещей”. 2-й по точности был класс “свойств”, а класс “отношений” “пострадал”, так как в него были записаны части речи, не относящиеся ни к “вещам”, ни к “свойствам”, ни к “отношениям”.

Повторное обучение нейронной сети SVM.

Повторно обучим нейронную сеть, но в качестве объекта распознавания возьмем 2-ю компоненту векторов (часть речи).

Анализ результатов для повторного обучения нейронной сети

Эксперименты показали, что наибольшей точностью обладает класс №1 или класс №2, в разных случаях по-разному. Класс №3 имеет наименьшую точность.

Что делать?

Из полученных результатов можно сделать вывод, что классы 1 и 2 обладают приемлемой надежностью, класс 3 в обеих сетях – неприемлемой. Отсюда

возникает потребность в экспертной системе, в случае, хотя бы 1 из нейронных сетей выдала ответ 2.

Экспертная система

Построение семантического дерева предложения

Постановка задачи

Пусть имеется предложение произвольной длины. Необходимо поставить каждому слову из этого предложения в соответствие слово из этого же предложения, от которого оно зависит. Главное слово в предложении не будет зависеть ни от какого другого слова. Данная разметка присутствует в корпусе syntagrus.

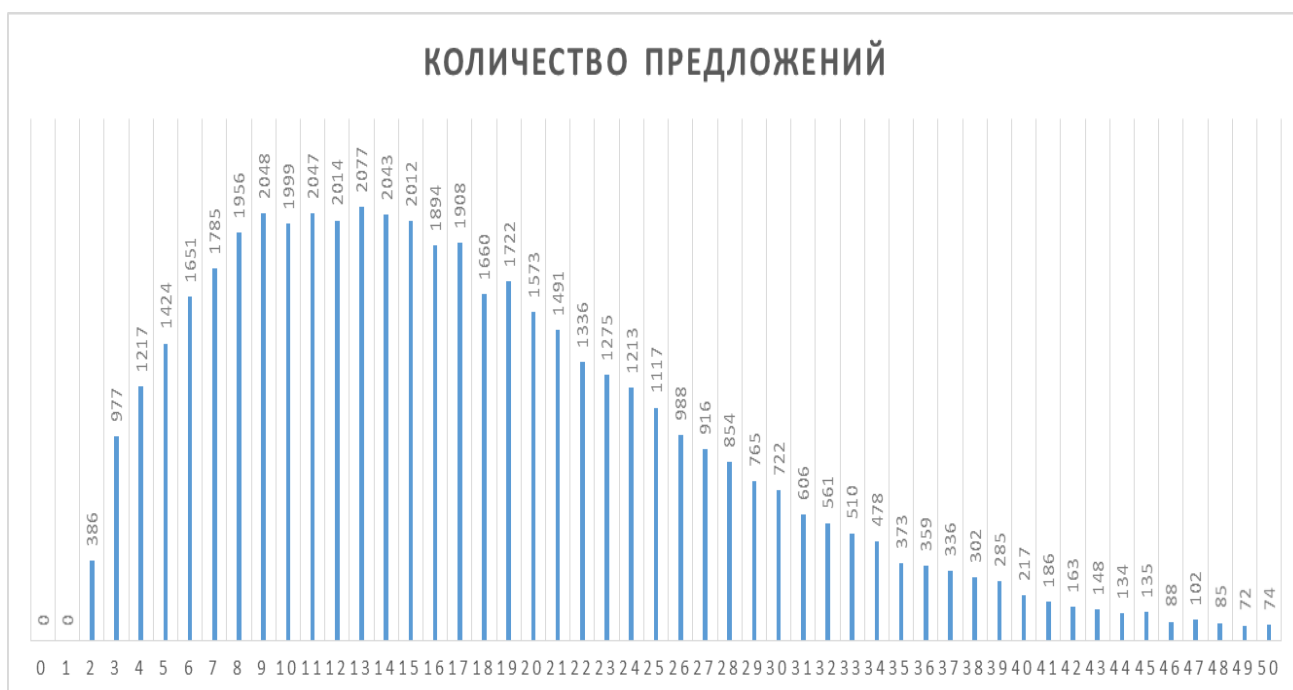
Методы решения задачи

Нейросетевой метод

Основное ограничение нейронных сетей – размерность входных данных должна быть фиксирована.

Проанализируем размер предложений в корпусе syntagrus. На рисунке ниже показан график количества предложений определенной длины. По оси x отложены длины предложений в словах, по оси y – количество предложений с такой длиной в корпусе.

Рис.6. количество предложений



Но при обучении нейронной сети можно немного “схитрить”. Например, для обучения построению дерева на 3-х словных предложениях можно взять не только предложения из 3 слов, но и 3 главных слова из более длинных предложений.

Например, в предложении:

“Кабинет отличался скромностью, присущей Семену Еремеевичу.”

Можно выделить 3 главных слова: “Кабинет”, “отличался”, “скромностью” с соответствующей иерархией слов: 2,0,2.

Тут следует заметить, что в корпусе главный член предложения – сказуемое. В модели же VSO лучше сделать главным членом предложения – подлежащее.

Таким образом, иерархия слов будет следующей: 0,1,2.

Тут можно сделать логичное замечание: достаточно перевернуть исходную иерархию подлежащего и сказуемого, чтобы получилась нужная иерархия.

Тут следует заметить, что подлежащие представляет собой “вещь” в модели VSO. А значит может иметь некоторое количество свойств. В данном случае при смене иерархии эти свойства будут все-равно будут зависеть от подлежащего, поэтому их “цифра” не изменится.

Кстати, тот же метод можно использовать и для предложений любой длины, а не только с длиной 3.

Также следует заметить, что нумерация слов в исходном предложении ведется от начала предложения. Нумерация же слов в полученном предложении из 3 слов будет от 1 до 3. Слова же будут в том же порядке, в котором они были в исходном предложении.

В предложении может быть и более 3 главных слов.

“В приемной его с утра ожидали посетители, - кое-кто с важными делами, а кое-кто и с такими, которые легко можно было решить в нижестоящих инстанциях, не затрудняя Семена Еремеевича.”

Экспертная система

