Домашнее задание № 3
Синтаксический анализ, коллокации
Отчет

Задание Е:

Составить (с использованием любого модуля морфоанализа) программу, выполняющую извлечение словосочетаний определенного вида из заданного русскоязычного текста. Выделение словосочетаний может базироваться на локальных высоковероятных синтаксических связях (см. слайд 50 Лекции № 7). Программа выводит все словосочетания заданного вида/ов, встречающиеся в обрабатываемом тексте. Рассмотреть несколько (2-5) грамматических образцов словосочетаний.

Протестировать программу на нескольких небольших текстах разных жанров.

Отчет: Описание грамматических образцов извлекаемых словосочетаний и стратегии (алгоритма) их выделения; составленная и примененная программа с комментариями, результаты ее тестирования.

1 Введение

Отчет содержит описание работы по созданию программы выполняющей извлечение словосочетаний определенного вида из текста.

Для токенизации текста и извлечения слов (игнорируя пунктуацию) использовалась система анализа текста Mystem в составе pymystem3, созданном компанией Яндекс. Для морфологического анализа токенов использовался pymorphy2, основы которого были изучены мной для первого домашнего задания.

2 Описание создания программы

Программный файл сохранен в формате .ipynp и выполняется в Jupyter Notebook на Python3.

Первая функция **read_txt(text)** выполняет загрузку текста из файла .txt, токенизацию и морфологический анализ каждого токена. Текст загружается из файла, разделяется на токены с помощью Mystem. Mystem может проводить морфологический анализ выделенных токенов, но его формат (формат тэга, приписываемого каждому токену) не позволяет выполнить поставленную задачу, выделяя из него часть речи, число, падеж, и так далее. Поэтому в качестве морфологического анализатора используется pymorphy2, в удобной форме приписывающий тэг каждому токену. С помощью анализатора присваиваются теги каждому слову. В случае возможной омонимии слову присваивается наиболее вероятный (на основе статистики) тег.

Следующие функции выполняют извлечение словосочетаний определенного вида.

1. **find_AN(tokens)** принимает список токенов в формате pymorphy, возвращает список словосочетаний формата прилагательное + существительное и существительное + прилагательное. Словосочетание считается согласованным, если у прилагательного и существительного совпадают род, число и падеж. Числительные и местоименные прилагательные так же учитываются как прилагательные, имена собственные считаются существительными. Из-за неверного разрешения омонимии анализатором pymorphy2 функция может возвращать не все словосочетания, которые находит в тексте человек.

[1]

Функция находит словосочетание, если между согласующимися словами не более двух других слов в прямом порядке ('идеальный политик и популист'), или слова идут подряд в обратном порядке ('ситуация другая'). В случае, если с существительным согласуются несколько прилагательных, стоящих перед ним, функция возвращает несколько словосочетаний — одно с первым прилагательным, одно со вторым, и т.д.

Описание алгоритма: Функция по порядку перебирает токены, проверяя часть речи. Как только встречается прилагательное, ищет вокруг существительное, к которому оно относится, и записывает найденное словосочетание в список. Искать сначала прилагательные эффективнее, чем искать существительные, так как не для всех существительных (которых много) есть прилагательные, но для всех прилагательных (которых мало) обычно есть существительные.

2. **find_VN_tran(tokens)** принимает список токенов в формате pymorphy, возвращает список словосочетаний формата переходный глагол+NP(прилагательные+ существительное в винительном падеже)

   Именная группа (прилагательные+существительное) формируется на основе функции find_AN(tokens). Функция находит словосочетание, если между глаголом и существительным не более трех слов. Так как функция не поддерживает разделение на предложения, в некоторых (редких) случаях полученные сочетания являются составленными из слов двух соседних предложений. Так как это случается довольно редко, я решила не бороться с этой проблемой.

   Описание алгоритма: сначала с использованием первой функции находятся и записываются в словарь именные группы. Затем по порядку перебираются токены в поисках переходного глагола. Для каждого глагола в трех словах после него ищется существительное в винительном падеже. Если оно найдено, проверяется, есть ли для него в словаре именная группа, и если она есть, то возвращается словосочетания глагола и группы, если группы нет — глагола и существительного. Функция возвращает список найденных словосочетаний.

3. **find_AV(tokens)** принимает список токенов в формате pymorphy, возвращает список словосочетаний из наречий глаголов в каком-либо порядке (глагольная группа). Так как к глаголу может относиться несколько наречий, то функция возвращает все возможные варианты разбора для близко стоящих глаголов и наречий.

   Описание алгоритма: функция перебирает по порядку токены в поисках наречия. Для каждого найденного наречия в трех словах вокруг ищется глагол, к которому оно будет относиться. Далее в словарь найденное наречие записывается в группу глагола, к которому оно относится. Таким образом, функция находит все группы, в которых есть наречия, объединяя в одно словосочетание глагол и все относящиеся к нему наречия. Возвращается список найденных словосочетаний.

[1] Например, в исследуемом тексте в одном из первых предложений встречается конструкция 'единственной региональной столицей', но функция возвращает только одно сочетание 'единственной столицей'. Это произошло потому, что слово 'региональной' (неверно) определено как прилагательное предложного падежа, в отличие от двух других других, у которых падеж предложный.

Работа программы была проверена не трех текстах разных жанров. Результаты (выделенные словосочетания) для этих текстов приведены ниже.

Репортаж

В качестве исследуемого новостного текста используется статья сайта ”Медуза“ о мэре города Якутска (ссылка на статью). Длина статьи около 5000 слов.

Первый абзац текста:

«На выборах 9 сентября 2018 года Якутск стал единственной региональной столицей, где мэром был избран представитель оппозиции. 48-летняя Сардана Авксентьева обошла единоросса Александра Саввинова, поддержанного губернатором, и стала первой женщиной-градоначальником в истории города. После избрания Авксентьева развернула бурную деятельность, направленную на исполнение «народного заказа»: принялась сокращать расходы на городской аппарат; увольнять сомнительных чиновников и подрядчиков; распродавать записанные на мэрию дорогие машины. В результате Авксентьева стала общероссийской знаменитостью — а успехами якутской мэрии в интернете даже заинтересовались в администрации президента. Спецкор «Медузы» Таисия Бекбулатова отправилась в Якутск, чтобы понять, как Авксентьевой удалось победить и какое будущее ее ждет. »

Примеры найденных словосочетаний вида ADJF+NOUN и NOUN+ADJF:

- ’единственной столицей’,
- ’бурную деятельность’,
- ’городской аппарат’,
- ’якутской мэрии’,
- ’какое будущее’,
- ’якутский бизнесмен’,
- ’телефонного разговора’,
- ’фёдоров который’,
- ’свою кандидатуру’,
- ’этой подготовки’

Примеры найденных словосочетаний вида VERB+NP:

- ’развернула бурную деятельность’,
- ’занял место’,
- ’возглавил якутию’,
- ’разыграл национальную карту’,
- ’делала упор’,
- ’подчёркивает проигрыш’,
- ’одобрил пенсионную реформу’,
- ’поддержала инициативу’,
- ’устанавливают большую палатку’,
- ’покинул пост’,
- ’поменял отношение’,
- ’дал интервью’

Примеры найденных словосочетаний вида VERB+ADVB и ADVB+VERB с возможностью нескольких наречий:

- ’действительно возглавил’,
- ’было по-другому’,
- ’сейчас говорит’,
- ’активно тоже встречался’,
- ’часто приходили’,
- ’ездили вечером вдвоём’,
- ’исправно судился’,
- ’смотрелась однозначно там’,
- ’было хорошо’,
- ’было уже тогда понятно’,
- ’сидят спокойно редко очень’,
- ’потом идут’,
- ’прекрасно знает’,
- ’сперва позвонил’,

Научно-популярная книжка о диетологии

Следующий жанр — научпоп. В качестве текста взят ознакомительный отрывок из научно-популярной книги по доказательной диетологии Елены Мотовой “Мой лучший друг желу-

док". Длина отрывка: около 1000 слов.

Абзац текста:

«Мы все отличаемся друг от друга способностью воспринимать вкус. Если высунуть язык и внимательно проинспектировать его, можно заметить возвышающиеся над поверхностью грибовидные структуры —сосочки. Они содержат рецепторы, которые, соединяясь со вкусовыми молекулами, дают мозгу представление о том, что мы едим. Количество и чувствительность рецепторов у всех разные. Это проверяли в нейробиологических исследованиях: как по реакции на эталонное химическое вещество горького вкуса — пропилтиоурацил, так и подсчитывая количество вкусовых сосочков. Можете протестировать себя и близких. Покрасьте кончик языка синим пищевым красителем и сосчитайте под лупой все вкусовые сосочки, как это показано на рисунке. Если их оказалось тридцать и больше, вы лучше чувствуете оттенки вкуса, чем 75% окружающих вас людей. Если сосочков меньше пятнадцати, вам сложнее различать вкусы. »

Примеры найденных словосочетаний вида ADJF+NOUN и NOUN+ADJF:

- 'научный журналист',
- 'калифорнийском университете',
- 'доказательной диетологии',
- 'доказательной лекции',
- 'новый формат',
- 'эту книгу',
- 'одном флаконе',
- 'доказательная диетология',
- 'пищеварительная система',
- 'пищевое поведение',
- 'красивой подаче',
- 'меньшем количестве'

Примеры найденных словосочетаний вида VERB+NP:

- 'ведёт блог',
- 'чувствуем вкус',
- 'регулируют вес',
- 'написала эту книгу',
- 'дают энергию',
- 'содержит лактозу',
- 'предложите овощи',
- 'отвергает такую еду'

Примеры найденных словосочетаний вида VERB+ADVB и ADVB+VERB с возможностью нескольких наречий:

- 'постоянно учится',
- 'хорошо готовит',
- 'почему толстеют',
- 'останется осторожно',
- 'буквально заставляет',
- 'улавливает буквально',
- 'гораздо уловила',
- 'развивалась эволюционно',
- 'получим достаточно',
- 'сообщает насколько',
- 'поэтому отказываются',
- 'рождаются редко',
- 'попробуйте сегодня',
- 'увлекайтесь избыточно',
- 'много пытайтесь'

Статья на Хабре

В качестве третьего примера текста взята статья из блога разработчиков Яндекса на habr.com, Как мы распределяем заказы между водителями в Яндекс.Такси. Это довольно специализированный текст о разработке и алгоритмах. Длина отрывка: около 1000 слов.

Абзац текста:

«Итак, трекер подготовлен, скоринг считается и в Tracker'е (жадное назначение), и в новом сервисе (DriverDispatcher'е), алгоритм решения задачи о назначениях отлажен и корректно работает. Появился вопрос, как интегрировать это всё в конечный автомат обработки заказа. Мы добавили отправку и удаление метаинформации о заказе в DriverDispatcher при переходе заказа из состояния в состояние. И это уже почти

работало. Почти — потому что итерации поиска исполнителя на заказ не контролировались извне. Мы могли просто заменить поход в трекер за водителем на поход в наш сервис и отдавать водителя, когда он найден, а до этого просто отдавать 404. Но это плохо, потому что нужно предлагать заказ водителю сразу, как только мы нашли заказ, и даже несколько секунд задержки тут играют роль: водитель может просто повернуть не в ту сторону, и заказ станет неактуален. Для этого мы сделали возможность вызвать процесс поиска исполнителя, не влияя на запланированные задачи. Так мы сохранили логику поиска (с перезапросами) и добавили возможность вызвать его вне планировщика.»

Примеры найденных словосочетаний вида ADJF+NOUN и NOUN+ADJF:

- 'холостого пробега',
- 'подходящего водителя',
- 'общая архитектура',
- 'конечным автоматом',
- 'жадный подход',
- 'жадный подход',
- 'таком подходе',
- 'таком этапе',
- 'локальном геоиндексе',
- 'прямому радиусу',
- 'этой информации',
- 'лучший вариант',
- 'эта логика',
- 'такое назначение'

Примеры найденных словосочетаний вида VERB+NP:

- 'нажимает кнопку',
- 'ранжировали водителей',
- 'используем множество',
- 'забили пул',
- 'добавили отправку',
- 'нашли заказ',
- 'играют роль',
- 'сделали возможность',
- 'сохранили логику',
- 'добавили возможность',
- 'нажмёт кнопку',
- 'требует множество'

Примеры найденных словосочетаний вида VERB+ADVB и ADVB+VERB с возможностью нескольких наречий:

- 'извне могли',
- 'контролировались извне',
- 'тут играют',
- 'происходит сразу',
- 'компенсируется более',
- 'далее последует',
- 'сегодня расскажу',
- 'выбираем наиболее',
- 'поэтому поступает',
- 'сначала определяет',
- 'затем уточняется',
- 'ранжировали уже',
- 'могут вообще',
- 'годится максимально',
- 'лежит хорошо',
- 'пришлось немного',
- 'сразу договорились',
- 'индивидуально попадали',
- 'поэтому добавили',
- 'обрабатывались параллельно',

3 Результаты выполненной работы

Создана программа, извлекающая словосочетания следующих типов: NOUN+ADJF, VERB+ADV, VERB+NP. Такие модели для словосочетаний выбраны с тем, чтобы поработать со всеми основными частями речи, а так же это одни из самых частых связей. Для обрабатываемого текста программа выводит все словосочетания заданного типа. Программа протестирована

на трех тестах разных жанров: журналистский репортаж Медузы, научно-популярная книга о питании, научный текст по алгоритмам с Хабра. Найдены словосочетания из текста.

Решенные проблемы: выделение словосочетаний из трех и больше слов (глагол+именная группа)

Нерешенная проблема: не все словосочетания выделяются из-за неверного разрешения омонимии (из-за неверного определения падежей не все словосочетания прилагательных и существительных возможно выделить). Возможное решение: использование более качественного морфологического анализатора. Так как тема этого задания именно синтаксический анализ, и функции опираются на готовый морфоанализатор, то при другом (более качественном) анализаторе они будут работать таким же точно образом, и выделять больше словосочетаний.