ОБЗОР СИСТЕМ МОРФОЛОГИЧЕСКОГО АНАЛИЗА РУССКИХ ТЕКСТОВ

Джекежанов А.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Петрова Н.Е. – к.филол.н.,доцент

В данной статье рассматриваются наиболее популярные морфологические процессоры русского языка, которые доступны для открытого использования. Обращается внимание на использование данных проектов в соревнованиях по морфологическому анализу русского языка, описывается, как они функционируют в практических приложениях.

С развитием компьютерных технологий и ростом потребности в обработке естественного языка, системы морфологического анализа становятся неотъемлемой частью многих языковых приложений. Русский язык, будучи одним из самых распространенных в мире, привлекает внимание разработчиков со всего мира, что приводит к появлению разнообразных инструментов для его морфологического анализа. В данной статье мы рассмотрим несколько из наиболее популярных морфологических процессоров русского языка, доступных для открытого использования, и проанализируем их основные особенности и возможности.

Проект «Автоматическая обработка текста» (АОТ) [1] включает в себя практически все этапы автоматического анализа текстов на естественном языке, в том числе и морфологический. Проект разрабатывался при поддержке группы лингвистов из РГГУ, основным разработчиком выступал А. Сокирко. Морфологический процессор был выложен в открытый доступ в 2004 году. Лексической основой служит словарь А. А. Зализняка [2], который включает более 161 тыс. лемм. Морфологический процессор АОТ предоставляет все функции полного морфологического анализа, включая нахождение леммы и морфологических характеристик словоформы, а также синтез словоформ. Его структура представлена в виде нескольких таблиц: лемм, флексий, приставок, морфологических характеристик. Таблица лемм содержит список псевдооснов слов со ссылками на таблицы флексий и приставок. В таблице флексий каждому из окончаний слов соответствует ссылка на соответствующие морфологические характеристики. Морфологический разбор слова по словарю состоит в поиске максимально совпадающей псевдоосновы в таблице лемм, поиск соответствующих приставки и окончания, а затем нахождение по таблице окончаний соответствующих морфологических характеристик. Синтез осуществляется похожим образом: после нахождения псевдоосновы по морфологическим характеристикам определяется вся парадигма и возвращается пользователю.

Для физического (бинарного) представления словаря используется структура конечного автомата [3]. Стоит отметить, что хранение морфологических характеристик сильно увеличивает число состояний автомата и, как следствие, время поиска в нем, поэтому в АОТ характеристики находятся в отдельной таблице, а сам автомат хранит ссылки на них. Итоговый размер словаря составляет около 9 МБ, что является небольшим значением для современных компьютеров.

Если словоформа не была найдена в словаре, то в этом случае в АОТ работает морфологическое предсказание. Первым шагом предсказания является попытка найти существующую словоформу языка, которая имела бы максимально общее окончание со входным

словом. Если при этом длина левой (неузнанной) части слова не превышает определенного размера (5 символов), а длина общего окончания со словарной словоформой не меньше 4 символов, тогда слово предсказывается по найденной правой части, т. е. берутся морфологические характеристики найденной словоформы. Если же такой подход не сработал, то ищется наиболее длинное совпадающее окончание. В настоящий момент морфопроцессор проекта AOT является полностью открытым и распространяется под лицензией LGPL. Однако проект не поддерживается и не имеет удобных средств для пополнения словаря.

Система TreeTagger [4] позиционируется как система для определения частей речи слов с возможностью настройки на любой естественный язык при наличии словаря и размеченного корпуса. Она была разработана в 1996 году в университете Штутгарта Хельмутом Шмидтом и на данный момент доступна для множества языков, включая русский.

Процессор TreeTagger позволяет определять части речи слов и другие морфологические характеристики, а также их нормальную форму. Основной упор в данном процессоре сделан на разрешение морфологической омонимии и предсказание характеристик неизвестных слов. TreeTagger базируется на словарной морфологии и использует словарь английского языка из проекта Penn TreeBank, содержащий более 2 млн. словоформ. Объем русского словаря неизвестен, однако по объему бинарного файла можно судить о сопоставимости размера словаря с английской версией. В процессе анализа используются 2 вида словарей: словоформ и суффиксов (имеются в виду флексии). Структуры данных, используемые для словарей, похожи на те, что используются в проекте AOT, и также являются вариациями минимальных конечных автоматов. Автомат суффиксов (флексий) строится из всех флексий слов длиной до пяти символов. При этом каждому из суффиксов приписывается соответствующая флективная часть речи на основе взвешенной энтропии Шенона. При этом узлы, имеющие значение энтропии меньше определенного порога, удаляются из автомата. Для снятия частеречной омонимии в TreeTagger используются решающие деревья [5] для частей речи, обученные на размеченном корпусе.

В узлах такого дерева находятся предикаты с ответом «да» или «нет» для двух предшествующих слов. При этом в листьях хранятся значения вероятностей для возможных ответов. Построение дерева происходит рекурсивно, с помощью модифицированного алгоритма ID3. На каждом шаге для двух предыдущих слов проверяются предикаты на равенство всем возможным частям речи, при этом для определения предиката, наилучшим образом разбивающего пространство признаков, используется правило максимизации энтропии Шенона. Для определения части речи входного слова достаточно, используя информацию о предыдущих словах, пройти по дереву от корня до листьев и выбрать наиболее вероятное значение. В настоящий момент TreeTagger распространяется в виде бинарного файла, код самого процессора является закрытым. Проект поддерживается, для него создаются новые словари под различные языки. Синтез словоформ в TreeTagger отсутствует. Это мы считаем существенным недостатком данного продукта.

Pymorphy2 [6] – морфологический процессор с открытым исходным кодом, предоставляет все функции полного морфологического анализа и синтеза словоформ. Он базируется на словарной морфологии и использует словарные данные проекта OpenCorpora [7]. Словарь содержит около 250 тыс. лемм, а также является полностью открытым и регулярно пополняемым. Словарь, как и в проекте AOT, логически представляет собой структуру из трёх таблиц, однако словарные данные хранятся в едином автомате. Для бинарного представления используется автомат с оптимизацией по памяти [8], что позволяет иметь в нем не более чем 2 32 различных связей, однако для задачи морфологического анализа данное ограничение не является существенным.

Итоговый размер данного словаря составляет около 7 МБ. В процессе морфологического синтеза, по исходной словоформе и тегам выполняется поиск нормальной формы слова, а затем перебор всех возможных пар ⟨окончание, теги⟩ в найденной лексеме, пока не будет найдена пара с заданными морфологическими тегами. После этого от нормальной формы отсекается её окончание, а найденное окончание приписывается к полученной псевдооснове. Для анализа неизвестных слов в Pymorphy2 используются несколько методов, которые применяются последовательно. Изначально от слова отсекается префикс из набора известных префиксов и если остаток слова был найден в словаре, то отсеченный префикс приписывается к результатам разбора. Если этот метод не сработал, то аналогичные действия выполняются для префикса слова длиной от 1 до 5, даже если такой префикс является неизвестным. Затем, в случае неудачи, словоформа разбирается по окончанию. Для этого используется дополнительный автомат всех окончаний, встречающихся в словаре с имеющимися разборами. В процессе построения из автомата удаляются редкие окончания и разборы. Метод анализа по окончанию аналогичен тому, что используется в процессоре AOT.

Разрешение морфоанализа построено на основе корпусной статистики. Если слово имеет несколько вариантов разбора, то среди всех выбирается наиболее вероятный. Вероятности определяются по следующей формуле: $P(w|t) = Fr(w,\ t) + 1\ Fr(w) + |R(w)|$. В приведенной формуле, $Fr(w)$ — количество раз, которое словоформа w встретилась в корпусе, а $Fr(w,\ t)$ — количество раз, которое эта словоформа встретилось с тегом t. $|R(w)|$ — число разборов, полученных от анализатора для словоформы w. В настоящее время Pymorphy2 поддерживается, при этом происходит постоянное пополнение корпуса OpenCorpora, что улучшает характеристики точности и полноты морфологического разбора.

Система Snowball разработана Мартином Портером и опубликована в 1980 году. Snowball использует систему суффиксов и окончаний для предсказания части речи и грамматических параметров. Так как одно и то же окончание может принадлежать разным частям речи или различным парадигмам, его оказывается недостаточно для точного предсказания. Применение суффиксов позволяет повысить точность обработки текста.

Система реализовывается на языке программирования в виде большого количества условных операторов, анализирующих самый длинный постфикс и его контекст. По окончании анализа слову приписывается часть речи и набор параметров, а найденное окончание (или псевдоокончание) отрезается. В итоге, помимо параметров, система возвращает стем. Система реализована на многих языках программирования и распространяется в исходных кодах, что позволяет легко встраивать ее в новые системы. Она не требует никакого словаря, однако расширение и уточнение правил выделения окончаний может оказаться нетривиальной задачей. Точность работы данного стеммера находится на уровне около 80%. Заметим, что использование методов машинного обучения, применённых к размеченному корпусу, позволяет получить гораздо лучшие результаты.

MyStem – морфологический анализатор, разработанный компанией Яндекс [9]. Первая версия была создана в 90-х годах, однако не имела большой популярности и не находилась в открытом доступе. Стоит отметить, что первая версия предполагала использование словаря небольшого размера, опираясь в основном на методы бессловарной морфологии, в то время как текущие реализации базируются на классическом подходе словарной морфологии.

В настоящий момент MyStem версии 3.0 предоставляет все функции полного морфологического анализа, однако нет функции синтеза. Данная версия является наиболее стабильной и доступной для скачивания в бинарном виде. Морфоанализатор MyStem базируется на словаре НКРЯ [10], который содержит более 200 тыс. лемм. Исходные коды MyStem являются закрытыми, поэтому характеристики использованной структуры данных не известны, однако размер полученного бинарного словаря более 20 МБ. MyStem производит разрешение морфологической омонимии и делает разбор несловарных словоформ. Для решения этой задачи используются различные методы машинного обучения. В зависимости от входных данных MyStem снимает омонимию двумя способами: с учетом контекста и без учета контекста. Снятие омонимии без учета контекста происходит благодаря обучению наивного баесовского классификатора на размеченном корпусе со снятой омонимией. Частоты встречаемости факторизируются и отдельно настраиваются для окончаний морфологических парадигм, основ парадигм и самих парадигм. Вероятность принадлежности неизвестного слова $word$, имеющего основу $stem$ и окончание $flex$, к парадигме $para$ рассчитывается по формуле: $P(para|word) = P(word|para) \cdot P(para)\ P(word) = = P(stem|para) \cdot P(flex|para) \cdot P(para)\ P(word)$. При этом предполагается, что $stem$ и $flex$ являются независимыми случайными величинами. Контекстное снятие омонимии является подключаемым и использует технологию MatrixNet. Основной идеей является ранжирование разборов на основе ближайших к разбираемому слов (контекстов). В настоящее время MyStem поддерживается и используется в ряде проектов, таких как НКРЯ. Также он доступен в виде динамической библиотеки для некоммерческих приложений и позволяет подключать собственные словари через опции командной строки или интерфейса библиотеки. В этом случае стандартный словарь полностью заменяется пользовательским.

Таким образом, из рассмотренных нами морфологических анализаторов мы хотим отметить Pymorphy2 как наиболее эффективный. Во-первых, он предоставляет все функции полного морфологического анализа, включая синтез словоформ, что делает его полноценным инструментом для работы с текстами на естественных языках. Во-вторых, он базируется на словаре OpenCorpora, который содержит более 200 тыс. лемм, что обеспечивает обширное покрытие слов и грамматических форм. Кроме того, Pymorphy2 имеет открытый исходный код, что позволяет разработчикам легко расширять его функциональность или встраивать его в свои проекты. В свою очередь, другие анализаторы, такие как MyStem и Snowball, также обладают своими преимуществами и характеристиками, например, MyStem использует различные методы машинного обучения для разрешения омонимии, а Snowball работает на классическом подходе словарной морфологии. Однако мы считаем, что Pymorphy2 обеспечивает баланс между

функциональностью, производительностью и доступностью, что делает его наиболее привлекательным выбором для широкого спектра приложений, требующих морфологического анализа текста.

Список использованных источников:

1. АОТ [Электронный ресурс]. – Режим доступа: http://aot.ru/docs/rusmorph.html. – Дата доступа: 31.03.2024.

2. Зализняк, А.А. Грамматический словарь русского языка / А.А. Зализняк – М.: Русский язык, 1980. – 882 с.

3. Fredkin, E. Trie memory Communications of the ACM / E. Fredkin – 1960. – Т. 3. – №. 9. – P. 490-499.

4. Schmid, H. Probabilistic part-of-speech tagging using decision trees. In.: Proceedings of the international conference on new methods in language processing / H. Schmid. – 1994. – P. 44-49.

5. Quinlan, J. R. Induction of decision trees. Machine learning / J. Quinlan. – 1986. – Т. 1. – №. 1. – С. 81-106.

6.Pymorphy2 [Электронный ресурс]. – Режим доступа: https://pymorphy2.readthedocs.io/en/latest/.– Дата доступа: 31.03.2024.

7.Открытый корпус OpenCorpora [Электронный ресурс]. – Режим доступа: http://opencorpora. org/.– Дата доступа: 31.03.2024.

8.Dawgdic [Электронный ресурс]. – Режим доступа: https://code.google.com/archive/p/ dawgdic.– Дата доступа: 31.03.2024.

9. Морфологический анализатор Mystem 3.0 [Электронный ресурс]. – Режим доступа:https://events.yandex.ru/events/yac/2014?openTalkVideo=571-51. – Дата доступа: 31.03.2024.

10. Национальный корпус русского языка [Электронный ресурс].– Режим доступа: http://ruscorpora.ru/. – Дата доступа: 31.03.2024.