УДК 004.855

ОБУЧЕНИЕ С НУЛЯ ГРАММАТИКИ СВЯЗЕЙ РУССКОГО ЯЗЫКА

С. В. Протасов[1]

В статье рассматривается вероятностная модель языка, основанная на грамматике связей и самообучающийся алгоритм, позволяющий устанавливать связи между словами в предложении. В процессе построения модели используются самые минимальные знания о грамматике. Предварительные результаты применения алгоритма для небольшого корпуса предложений показали хорошие результаты.

1 Введение

В настоящий момент в некоторых практических приложениях пользуются популярностью n-грамные модели грамматики [Bahl et al., 1983], [Jelinek, 1997]. К n-грамным грамматикам относятся и триграмные модели, занимающие сильные позиции в статистическом моделировании языка [Brown et al., 1992]. Однако в триграмной модели каждое слово зависит только лишь от двух предыдущих, и не может учитывать дальние связи в предложении. Если лингвистический формализм будет иметь дальние связи, то он потенциально должен иметь лучшие характеристики при моделировании естественного языка. В данной работе изучаются *вероятностные грамматики связей*, относительно новый контекстно-свободный формализм (относительно грамматик непосредственно составляющих [Chomsky, 1957] и грамматик зависимостей [Mel'chuk, 1979]), которые впервые были предложены в работе [Sleator et al., 1991], а применимость для русского языка была показана в работе [Протасов, 2005]. Формализм грамматики связей содержит n-грам модели как подкласс и одновременно допускает наличие дальних связей [Lafferty et al., 1992].

В данной работе рассмотрена концепция грамматики связей, её вероятностная модель и обучающий алгоритм. На базе алгоритма была создана программа, которая при тестировании на реальных русскоязычных текстах показала значительное снижении *кросс-энтропии*[2], что подтверждает принципиальную возможность существования *автоматизированных* технологий создания *грамматики связей* русского языка. Предполагается вывод *правил* грамматики и оценка вероятности срабатывания выведенных правил грамматики при отсутствии подробной грамматической теории. Исследование возможности создания *грамматики связей* с помощью только лишь анализа неразмеченного корпуса предложений есть главная цель данной работы.

Попытки статистического обучения контекстно-свободных грамматик уже совершались [Lari et al., 1990] [Jelinek et al., 1992] [Yuret, 1998] [Collins, 1999], однако *кросс-энтропия* моделей языка либо не указывалась, либо была хуже триграмных моделей [Brown et al., 1992]. В большинстве подобных работ для обучения используются предварительно размеченные тексты, специфичные для данной модели языка, что сильно осложняет возможность сравнения самих моделей языка с другими формализмами. Мы же попытались получить численное значение (кросс-энтропии), которое можно сравнивать с другими моделями. Так как *грамматика связей* имеет дальние связи, контекстную свободу и эффективный алгоритм разбора, то мы надеемся на получение преимущества над триграмными моделями.

Далее в работе будет показан процесс создания вероятностной модели языка, основанной на *грамматике связей*. После короткого описания концепции *грамматики связей* мы рассмотрим алгоритм разбора и обучения. После чего будут обсуждены вопросы, касающиеся сглаживания параметров, лингвистических ограничений и оценки качества модели.

[1]Россия, Москва 117303, ул. Керченская, д. 1«А», корп. 1, МФТИ, svp@mtu.ru

[2]кросс-энтропия - (нестрого) среднее число бит, необходимых для кодирования каждого слова с помощью модели грамматики языка

2 Предварительная информация

2.1 Грамматика связей

Лучшим способом объяснить основы *грамматики связей* является демонстрация *связки*. *Связка* - один из вариантов разбора, разрешенный *грамматикой связей*. На рисунке 1 показан пример связки слов, разрешенный *грамматикой связей*, разработанной вручную [Протасов, 2005].

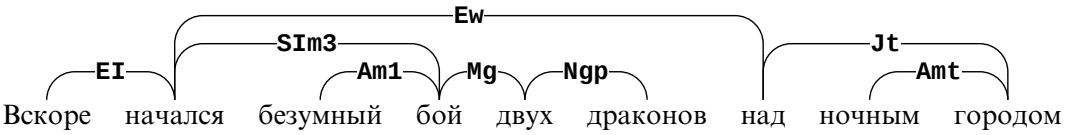

Рис. 1: *Связка слов.*

Использование слова w определяется теми способами, которыми его можно соединить в предложении. К примеру, слово *"двух"* соединено со словом *"бой"* и со словом *"драконов"*. Таким образом, использование слова *"двух"* можно описать коннектором Ngp с правой стороны и коннектором Mg с левой стороны. Эти требования по связям можно записать в виде $((Mg),(Ngp))$. Соответственно использование слова *"бой"* можно записать в виде $((SIm3),(Mg))$. Подтипы коннекторов позволяют делать обобщающие или, наоборот, специализированные соединения. К примеру, коннектор $Mg-$ (родительный падеж) может соединяться с коннектором общего типа $M+$, но не может соединяться с коннектором $Mt-$ (творительный падеж), который присутствует в *валентностях* некоторых глаголов.

Все возможные использования для каждого слова записываются в *словаре* в форме списка формул с операцией *ИЛИ* над элементарными слагаемыми требований по связям, которые далее мы будем называть *дизъюнктивными слагаемыми* или просто *дизъюнктами* и обозначать каждое такое слагаемое как:

$$d = ((l_m, l_{m-1}, ..., l_1), (r_1, r_2, ..., r_n)),$$

где l_i - список левых коннекторов и соответственно r_j - правых коннекторов. Таким образом *дизъюнкт* - это два связанных упорядоченных списка коннекторов. У каждого слова W с выбранным *дизъюнктом* d должны быть соединены все левые и правые коннекторы. Мы можем выбирать только подходящий *дизъюнкт*, у нас нет права оставлять пустым хотя бы один коннектор r_j и l_i . Каждый коннектор l_i соединяется с некоторым коннектором $\tilde{r}_i$, от находящегося слева от него слова L_i, а r_j - с некоторым коннектором $\tilde{l}_j$ слова W_i, справа от него. На рисунке 1 к примеру левый коннектор Mg в слагаемом $((Mg)(Ngp))$ для слова "два" соединен с правым коннектором Mg слагаемого $((SIm3),(Mg))$ слова "бой". Списки левых и правых коннекторов строго упорядочены и требуют, чтобы слова, с которыми соединяются коннекторы $l_m, l_{m-1}, ...,$ удалялись от слова W с уменьшением i, а слова, с которыми соединяются коннекторы $r_1, r_2, ..., r_n$, приближались с ростом i. То есть коннекторы, стоящие по бокам формулы *дизъюнктивного слагаемого*, требуют близкие слова, а коннекторы, стоящие в центре формулы, могут образовывать дальние связи.

Далее обозначим первый левый коннектор дизъюнкта d как $left[d] = l_1$, а первый правый как $right[d] = r_1$. Причем нам не запрещено иметь пустые списки: $left[d] = nil$ $right[d] = nil$ или $(()())$. Напомним также, что дизъюнкт d - это два связанных упорядоченных списка коннекторов.

Разбор или *связка* предложения составляется путем выбора *дизъюнктов* для всех слов. Все коннекторы во всех выбранных *дизъюнктах* должны соединяться друг с другом не более одного раза, то есть на каждый коннектор по одной связи. Соединенные коннекторы образуют связи и граф, где узлы - слова, а дуги - связи с названиями коннекторов. Граф расположен выше линейно расположенных слов. Дуги графа не пересекаются (это свойство называется планарностью[3]). Если подобрать *дизъюнкты* невозможно, тогда разбираемое предложение не принадлежит языку.

В работах [Протасов, 2005] [Sleator et al., 1993] [Sleator et al., 1991] *грамматика связей* разъяснена более подробно. У неё есть некоторое сходство с грамматикой зависимостей [Mel'chuk, 1979] [Mel'cuk, 1988], однако *грамматика связей* может иметь циклы и в ней отсутствует *корневое слово*. В работе [Протасов, 2005] можно найти подробное сравнение с грамматикой зависимостей, довольно популярной у нас в стране. Следует отметить, что демонстрационный рисунок 1 соответствует грамматике связей, разработанной вручную с использованием школьных правил грамматики русского языка. В дальнейшем мы предполагаем отсутствие априорных знаний о грамматических связях.

[3] планарность - слабая проективность в формализме грамматики зависимостей

2.2 Алгоритм анализа

Алгоритм обучения *грамматики связей* во многом опирается на алгоритм разбора. Далее он будет очень кратко рассмотрен. Более подробно алгоритм разбора представлен в работе [Sleator et al., 1991]. Алгоритм представляет собой рекурсивный разбор предложения сверху вниз с кэшированием промежуточных результатов. На каждом этапе перебора подсчитывается число связок между левым словом L и правым словом R, которые (связки) должны удовлетворить коннекторам l и r, направленным соответственно вправо и влево. Алгоритм перебирает все *дизъюнктивные слагаемые* d у всех слов W, где $L < W < R$ (L, W, R - порядковые номера слов в предложении), и ищет те, которые могут соединяться с l и r. Далее есть три варианта развития событий (рисунок 2):

- коннектор $left[d]$ связан с l, $right[d] = nil$ или свободен;
- коннектор $left[d] = nil$ или свободен, а $right[d]$ связан с r;
- $left[d]$ связан с l, и $right[d]$ связан с r

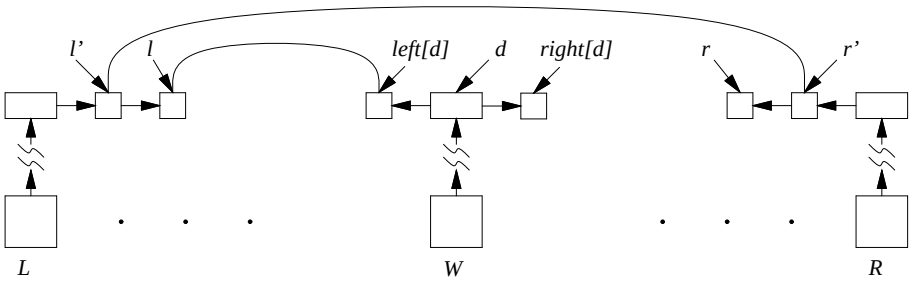

Рис. 2: *Рекурсивный алгоритм разбора*

После того как связи со словом W определены - задача решается рекурсивно для интервалов $L..W$ и $W..R$. Далее мы будем обозначать слова в предложении $S = W_0, W_1, ..., W_{N-1}$ с помощью индекса $0, 1, ..., N-1$.

Пусть $c(L, R, l, r)$ - число способов конструирования *подсвязок* между словом L и словом R с использованием коннекторов l и r.

Задача подсчета $c(L, R, l, r)$ сводится к расчету самой себя через рекурсию - путем перебора всех промежуточных слов W между L и R. В работе [Lafferty et al., 1992] выводятся рекурсивные формулы для счетчиков $c(L, R, l, r)$, позволяющие рассчитывать все параметры без полного перебора вариантов за время обычного разбора. Позже эти счетчики используются для оценки параметров вероятностной модели языка.

2.3 Вероятностная модель

Если всему языку грамматики[4] назначить полную вероятность, равную 1, тогда каждое предложение языка будет иметь свою вероятность появления, и мы сможем оценивать, какова вероятность того, что данная последовательность слов и связей между ними принадлежит языку. Подобным свойством обладает также и обычная вероятностная модель контекстно-свободной грамматики непосредственных составляющих, заданная параметрами $P_A(A \to BC)$ и $P_A(A \to w)$.

Когда основным оператором контекстно-свободной грамматики непосредственных составляющих является *оператор замены* нетерминальных символов, в *грамматике связей* основным оператором является *оператор связывания*. Связь зависит от двух коннекторов - от левого коннектора l и от правого коннектора r.

Это прямые аналоги нетерминального символа A, который заменяется в грамматике непосредственных составляющих. Формирование связи для заданных l и r происходит следующим образом. Сначала выбирается слово W, далее у слова W выбирается *дизъюнкт* d из списка дизъюнктивных слагаемых для этого слова. Наконец, выбирается *ориентация*, которая определяет один из трех способов соединения: $d \to l$, $(d \to l)\&(d \to r)$, $d \to r$.

[4] весь язык - это множество (возможно бесконечное) всех предложений языка, удовлетворяющий правилам грамматики языка

Также мы можем принять во внимание слова L и R, с которыми связаны коннекторы l и r. Все это приводит к набору параметров Pr вероятностной модели языка:

$$Pr(W, d, O|L, R, l, r) \tag{1}$$

Здесь O параметр вероятности, обозначающий ориентацию связи, который может принимать значения $\leftarrow, \rightarrow, \leftrightarrow$ в случаях d соединен с l, d соединен с r, или d соединен и с l и с r. Эту вероятность можно разложить на компоненты

$$\begin{aligned} &Pr(W, d, O|L, R, l, r) = \\ &\quad Pr(W|L,R,l,r)Pr(d|W,L,R,l,r)Pr(O|d,W,L,R,l,r) \end{aligned} \tag{2}$$

Так как мы формируем условные вероятности для набора событий, которые должны на естественном языке встречаться достаточно часто, то у нас могут быть проблемы с робастностью параметров Pr и с адекватностью оценки. Поэтому мы используем приближение для вероятности связи:

$$\begin{aligned} &Pr(W, d, O|L, R, l, r) \approx \\ &\quad Pr(W|L, R, l, r)Pr(d|W, l, r)Pr(O|d, l, r) \end{aligned} \tag{3}$$

В дополнение нам понадобится соединяющая вероятность $Pr(W_0, d_0)$ для самого первого слова и его дизъюнктивного слагаемого.

Вероятность связки равна произведению вероятностей всех связей, из которых связка формируется. Поэтому связку $\mathcal{L}$ можно представить в виде набора связей $\mathcal{L}\{(W, d, O, L, R, l, r)\}$ вместе с начальным дизъюнктивным слагаемым d_0, а потом подсчитать вероятность этой связки:

$$\begin{aligned} &Pr(S, \mathcal{L}) = \\ &\quad Pr(W_0, d_0) \prod Pr(W, d, O|L, R, l, r) \end{aligned} \tag{4}$$

где произведение берется по всем связям в связке $\mathcal{L}$ предложения S. Эта вероятность должна совпадать с вероятностью генерации связки $\mathcal{L}$ для предложения S. Тогда *кросс-энтропия* корпуса предложений $S_1, S_2, \ldots$ по отношению к равномерному распределению предложений языка будет определена как

$$H = -y^{-1} \sum_i log \sum_{\mathcal{L}} Pr(S_i, \mathcal{L}) \tag{5}$$

при некоторой нормализующей константе y. Далее будет описан алгоритм, который ищет локальный минимум этой энтропии. Найденные параметры позволят нам сконструировать грамматику, анализирующую новые предложения.

3 Обучающий алгоритм

Целью алгоритма обучения грамматики является нахождение максимально-правдоподобных оценок параметров *вероятностной грамматики связей*. Алгоритм работает в духе Inside-Outside [Jelinek et al., 1992] [Lari et al., 1990], который по сути является частным случаем более общего алгоритма EM (Expectation Maximization) [Baum, 1972]. В алгоритме рассчитываются две вероятности: внешняя вероятность $Pr_{\mathcal{I}}$ и внутренняя вероятность $Pr_{\mathcal{O}}$. Внутренняя вероятность $Pr_{\mathcal{I}(L,R,l,r)}$ - это вероятность того, что слова между L и R могут быть связаны таким образом, что удовлетворяются требования по связям коннекторов l и r. Внешняя вероятность $Pr_{\mathcal{O}(L,R,l,r)}$ - это вероятность того, что слова вне слов L и R связаны таким образом, что удовлетворены требования по связям вне коннекторов l и r. Имея данные вероятности, получаем вероятность того, что предложение $W_0, ..., W_{N-1}$ генерируется нашей грамматикой:

$$Pr(S) = \sum_{d_0 \in \mathcal{D}(W_0)} Pr(W_0, d_0) Pr_{\mathcal{I}}(0, N, right[d_0], nil) \quad (6)$$

Внутренние и внешние вероятности вычисляются рекурсивно через формулы из работы [Lafferty et al., 1992]. Там же можно найти счетчики для параметров $Pr(d|W,l,r)$ и $Pr(0|d,l,r)$. Из-за громоздкости формул здесь они не будут приводиться.

Алгоритм для расчета оценок параметров Pr основан на рекурсивном алгоритме разбора *грамматики связей* [Sleator et al., 1991]. Алгоритм использует кэширование[5]. Каждая итерация поиска минимума *кросс-энтропии* корпуса состоит из трех стадий. На первой стадии рассчитываются внутренние вероятности тем же способом, что и считается число связок при обычном разборе. На второй стадии рассчитываются внешние вероятности. Наконец, третий этап обновляет счетчики параметров модели согласно реккурентным формулам.

Общая сложность обучающего алгоритма составляет $O(nN^3D^3)$ на каждый шаг обучения, где N - среднее число слов в предложении, D - верний предел числа дизъюнктивных слагаемых, n - число предложений в корпусе. D в свою очередь зависит от максимально разрешенного числа коннекторов с каждой стороны m и от числа типов связей. Если тип связи всего один, то $D = m^2 - 1$. Таким образом, отсутствие экпоненциального роста требуемых ресурсов позволяет надеяться на легкую масштабируемость и применимость алгоритма к крупным текстам.

3.1 Сглаживание

Сглаживание значительно улучшает качество обучения вероятностной модели языка. Языковые явления могут быть довольно редки и чтобы обучающий алгоритм сходился, необходимы сглаживающие оценки параметров Pr. В качестве первой такой оценки можно предложить следующее приближение:

$$\widetilde{Pr}(W|L,R,l,r) = \gamma^{-1}\delta_{l,r}(W)\Big[\lambda Pr(W|L,R) + (1-\lambda)Pr(W|L,R,l,r)\Big] \quad (7)$$

где $\delta_{l,r}(W)$ равна единице, когда W имеет дизъюнктивное слагаемое, которое может соединяться либо с l, либо с r, и нулю в остальных случаях; λ подбирается через *удаление интерполяции* [Bahl et al., 1983] [Manning and Schutze, 1999]; а γ - нормализующая константа. Этот метод сглаживания довольно привлекателен тем, что $Pr(W|L,R)$ могут быть получены из *неразмеченного* текста. Фактически для каждого предложения S нужно рассмотреть $\binom{|S|}{3}$ троек слов, которые потенциально могут связываться друг с другом. Если предположить, что число слов в русскоязычных предложениях подчиняется распределению Пуассона со средним значением в 25 слов, то тогда нам нужно будет в среднем рассмотреть около 2600 троек на каждое предложение, что примерно в 100 раз больше, чем обычных триграм. Мы можем рассматривать вероятность $Pr(W|L,R)$ как *априорную* вероятность того, что тройка (L,W,R) согласно названию данному в работе [Lafferty et al., 1992] формирует *грамматическую триграму*.

После получения первых оценок параметров нашей модели языка можно получить *апостериорную* вероятность *грамматических триграм* через формулу:

$$\widetilde{Pr}(W|L,R) = \sum_{l,r} \widetilde{Pr}(W|L,R,l,r) Pr(l,r|L,R) \quad (8)$$

где вероятности $Pr(l,r|L,R)$ рассчитываются через соединяющую вероятность $Pr(L,R,l,r)$, которая оценивается через счетчики:

$$Count(L,R,l,r) = Pr_{\mathcal{O}}(L,R,l,r) Pr_{\mathcal{I}}(L,R,l,r) \quad (9)$$

Мы можем повторять эту процедуру на каждом шаге обучения модели. Дальнейшие техники сглаживания подробно рассмотрены в работах [Bahl et al., 1983], [Manning and Schutze, 1999].

[5] кэширование - мемоизация, запоминание промежуточных результатов в памяти

3.2 Ограничения на поиск

Эксперимент заключается в выводе набора параметров вероятностей *грамматики связей* путем анализа корпуса предложений русского языка. Предполагается, что *априори* о грамматике языка ничего неизвестно. Предложим самые простые и сильные ограничения.

Единственный тип связи. Для уменьшения пространства поиска был введен единственный тип связи. На рисунке 3 показан пример связки с единственным типом связи N. Так как число дизъюнктивных слагаемых растет по экспоненте от числа типов связей, то обучение без этого ограничения крайне затруднительно.

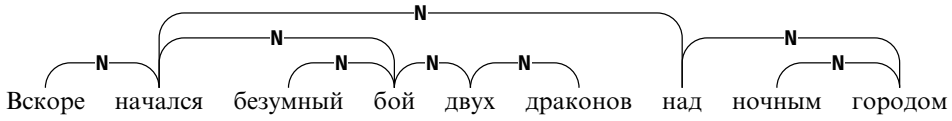

Рис. 3: *Единственный тип связи N.*

Небольшое число дизъюнктивных слагаемых. В грамматике языка каждое слово содержит некоторый набор дизъюнктивных слагаемых. В алгоритме обучения требуется ввести ограничение на число дизъюнктивных слагаемых. В противном случае пространство поиска будет очень большим и результаты будут плохими. При изучении *грамматики связей*, разработанной вручную, было обнаружено, что многие слова имеют очень мало способов для соединения с другими словами. К примеру, предлоги всегда имеют только два *дизъюнкта*: $(N)(N)$ и $()(N,N)$. Однако этот факт не использовался в нашей модели языка.

Единственный класс слова. В алгоритме не используются знания о частях речи. Предполагается, что набор дизъюнктивных слагаемых после процедуры обучения и подскажет нам грамматический класс слова.

3.3 Процедура обучения

Процедура обучения состоит из 3-х этапов:

1. Инициализация вероятностных параметров
2. Обучение на неразмеченном корпусе предложений.
3. Отсев дизъюнктивных слагаемых.

Инициализация вероятностных параметров. Так как алгоритм обучения гарантирует только нахождение локального минимума кросс-энтропии, то задача выбора начальных параметров Pr очень важна для качества обучения. В процедуре инициализации мы предполагаем, что все события равновероятны. То есть в процессе генерации предложения вероятность выбора слова равна вероятности выбора любого другого слова, а вероятность выбора дизъюнкта равна вероятности выбора любого другого дизъюнкта.

Обучение на неразмеченном корпусе предложений. Для обучения важно подобрать корпус. Желательно максимальное значение отношения числа предложений к числу слов, чтобы для каждого слова было достаточно примеров его использования. Для этих целей был создан корпус, где число предложений примерно в 10 раз больше числа используемых слов. Корпус создавался автоматически, путем фильтрации большого корпуса русского языка и поиска только тех предложений, все слова в которых принадлежали заданному набору из 355 высокочастотных слов. Если бы мы попытались составить корпус из более простых слов средней частотности, то мы бы получили меньшее число предложений, что для наших целей обучения не совсем подходит.

Отсев дизъюнктивных слагаемых. В конце обучения происходит отсев дизъюнктивных слагаемых с низкими вероятностями. После процедуры удаления общие вероятности пересчитываются, и число вариантов разбора уменьшается. Для увеличения скорости работы алгоритма процедуру удаления дизъюнктивных слагаемых можно повторять на каждом шаге обучения.

4 Эксперимент

Для целей эксперимента был составлен русскоязычный корпус из 2780 предложений русского языка длины от 3 до 8 слов, содержащий 355 разных слов. Максимальное число коннекторов с

каждой стороны было ограничено двумя, а число дизъюнктивных слагаемых восьмью. Порог отсева дизъюнктивных слагаемых был установлен на 0.001. Корпус был разбит на обучающую часть и на тестирующую часть тремя различными способами согласно таблице 1.

Корпус	Объем	Словарь
corpora1.train	2680	355
corpora1.test	100	159
corpora2.train	2580	355
corpora2.test	200	177
corpora3.train	2480	355
corpora3.test	300	185

Таблица 1: *Параметры корпусов*

Так как предложения для тестовой части корпуса выбирались случайно, число различных слов в словаре тестового корпуса имеет некоторую вариацию.

4.1 Результаты эксперимента

В общем случае алгоритм разбора может допускать несколько разрешенных связок. В качестве результата выбиралась связка, имеющая наибольшую вероятность. При анализе результатов разбора после обучения возникла проблема методологического характера. Не существует однозначного критерия правильного разбора. Алгоритм находит связи между словами, которые с одной стороны противоречат школьным правилам грамматики, с другой стороны, интуитивно приемлемы. В тестовом корпусе существуют предложения, по которым у разных носителей русского языка разные мнения о правильном разборе.

Разборы оценивались на принадлежность к трем классам: правильные, приемлемые и неправильные. На рисунке 4 приведен пример *связки* правильного разбора, а на рисунке 5 - приемлемого, на рисунке 6 - неправильного.

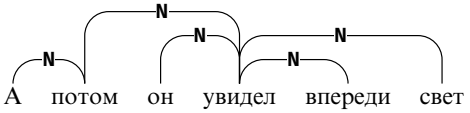

Рис. 4: *Пример правильного разбора*

Рис. 5: *Пример приемлемого разбора*

В таблице 2 приведены проценты приемлемых и правильных разборов для предложений из тестовых корпусов. Читатель может самостоятельно оценить адекватность полученных коэффициентов [6].

4.2 Перплексивность

Предсказательная сила модели языка часто оценивается через *перплексивность* на основе тестового корпуса. Перплексивность определяется как $perplexity = 2^H$, где H - кросс-энтропия, и по сути является более удобным (растянутым) аналогом кросс-энтропии. Для одного и того же набора данных, меньшая *перплексивность* означает лучшую модель языка. Одна и та же модель, исследующая разные языки, характеризует сложность самого языка. Впервые оценивать модели языка через *перплексивность* было предложено в работе [Bahl et al., 1977]. *Перплексивность* можно понимать как некий *коэффициент неопределенности* модели языка.

Перплексивность на наших тестовых корпусах была подсчитана для нашей *грамматики связей* и для *биграмной* модели [Brown et al., 1992]. Небольшой размер корпуса не позволяет сделать

[6] См. корпус разборов http://sz.ru/parser/corpora1-test.txt

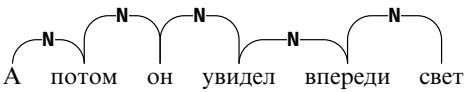

Рис. 6: *Пример неправильного разбора/Пример биграм модели*

Корпус	% правильных	% приемлемых
corpora1.test	23	65
corpora2.test	28	50.5
corpora3.test	19.3	59

Таблица 2: *Проценты приемлемых и правильных разборов.*

Корпус	Биграм	Грамматика связей
corpora1	63.1	43.5
corpora2	57.3	46.8
corpora3	57.1	45.8

Таблица 3: *Перплексивность моделей.*

сравнение с триграмной моделью языка [Brown et al., 1992], поэтому для сравнения мы использовали биграмную модель, где каждое слово зависит только от предыдущего. Биграмная модель допускает только единственный последовательный вид связки, часто неправильный (рисунок 6), таким образом биграм модель не может на равных участвовать в соревновании "правильных разборов", но несмотря на это она обладает неплохой предсказательной силой и перплексивностью.

Результаты сравнения модели биграм и модели грамматики связей показаны в таблице 3.

Вариация в оценках перплексивности в зависимости от тестового корпуса позволяет судить о достигнутой точности оценки. Из таблицы видно, что на всех трех случайных выборках показатели грамматики связей оказались лучше биграм модели.

5 Заключение

В данной работе был рассмотрен метод получения *вероятностной грамматики связи* только лишь на основе анализа корпуса языка. Оригинальная модель языка [Lafferty et al., 1992] была упрощена до небольшого числа *дизъюнктов* и одного типа коннекторов. При тестировании использовался небольшой корпус неразмеченных русскоязычных предложений из ограниченного списка слов. Результаты показали хороший процент разбора и стабильное превосходство над биграмной моделью языка.

Так как в эксперименте использовались высокочастотные слова, которые довольно сложны из-за большого количества потенциальных связей, то при добавлении в обучающий корпус предложений, состоящих из более простых слов, следует ожидать увеличение коэффициента приемлемых разборов.

Следующим шагом в развитии статистического обучения грамматики является ввод вспомогательных данных (учет частей речи) и обучение на более крупных текстах. Обучение может происходить поэтапно, от простого к сложному, опираясь на результаты предыдущего шага, постепенно добавляя дополнительные слова, *дизъюнктивные слагаемые* и более длинные предложения.

Список литературы

[Протасов, 2005] Протасов С. В. Автогенерация семантических словарей с использованием грамматики связей для русского языка. //Процессы и методы обработки информации. М., 2005.

[Bahl et al., 1977] Bahl L. R., Baker J. K., Jelinek F., Mercer R. L. Perplexity – A measure of the difficulty of speech recognition tasks. //J. Acoust. Soc. Amer., vol. 62. 1977.

[Bahl et al., 1983] Bahl L. R., Jelinek F., Mercer R. L. A maximum likelihood approach to continuous speech recognition. //IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5(2):179–190. 1983.

[Baum, 1972] Baum L. E. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process. //Inequalities, 627(3):1-8. 1972.

[Brown et al., 1992] Brown P. F. Stepen A. L. An estimate of an upper bound for the entropy of English. //Computational Linguistics. 1992.

[Chomsky, 1957] Chomsky N. Syntactic Structures. //Mouton, The Hague, 1957.

[Collins, 1999] Collins M. Head-Driven Statistical Models for Natural Language Parsing. //PhD Dissertation, University of Pennsylvania. 1999.

[Jelinek et al., 1992] Jelinek F. Lafferty J. Mercer R. Basic methods of probabilistic context-free grammars. //In Speech Recognition and Understanding. 1992.

[Jelinek, 1997] Jelinek F. Statistical Methods for Speech Recognition. //MIT Press. 1997.

[Lafferty et al., 1992] Lafferty J. Sleator D. Temperley D. Grammatical Trigrams: A Probabilistic Model of Link Grammar. //Proceedings of the AAAI Conference on Probabilistic Approaches to Natural Language. 1992.

[Lari et al., 1990] Lari K. Young S. J. The estimation of stochastic context-free grammars using the inside-outside algorithm. //Computer Speech and Language. 1990.

[Manning and Schutze, 1999] Manning C., Schutze H. Foundations of Statistical Natual Language Processing. //Cambridge, MA: MIT Press, 1999.

[Mel'chuk, 1979] Mel'cuk I.A. Studies in dependency syntax. //Karoma Publishers, Ann Arbor, 1979.

[Mel'cuk, 1988] Mel'cuk I.A. Dendency Syntax: Theory and Practice. //State University of New York Press. 1988.

[Sleator et al., 1991] Sleator D. Temperley D. Parsing English with a Link Grammar. //Carnegie Mellon University Computer Science technical report CMU-CS-91-196. 1991.

[Sleator et al., 1993] Sleator D. Temperley D. Parsing English with a Link Grammar. //Third International Workshop on Parsing Technologies. 1993.

[Yuret, 1998] Yuret D. Discovery of Linguistic Relations Using Lexical Attraction. //PhD thesis, MIT. 1998.