АВТОМАТИЗАЦИЯ ПРОЦЕССА ТРАНСКРИПЦИИ ДЛЯ ЗАДАЧИ МНОГОЯЗЫКОВОЙ ПЕРЕДАЧИ ИМЕН СОБСТВЕННЫХ*

А.В. Бондаренко
Государственный Научно Исследовательский Институт Авиационных Систем
Россия 125319 Москва ул. Викторенко, 7

В.А. Галактионов
Институт Прикладной Математики им. М.В. Келдыша РАН
Россия, 125047, Москва, Миусская пл., д.4

А.А. Герасименко
Государственный Научно Исследовательский Институт Авиационных Систем
Россия 125319 Москва ул. Викторенко, 7

С.В. Ёлкин
Институт Прикладной Математики им. М.В. Келдыша РАН
Россия, 125047, Москва, Миусская пл., д.4

А.М. Мусатов
Институт Прикладной Математики им. М.В. Келдыша РАН
Россия, 125047, Москва, Миусская пл., д.4

Э.С. Клышинский
Институт Прикладной Математики им. М.В. Келдыша РАН
Россия, 125047, Москва, Миусская пл., д.4

Слёзкина О.Ю.
Российский Государственный Гуманитарный Университет
Россия 125267 Москва Миусская площадь д.6

Введение

Корректная передача звучания иностранных слов на русский язык является важной задачей в области налаживания коммуникации. Благодаря этому можно решить проблему узнаваемости названий организаций, мест, людей. В связи с тенденциями калькирования терминов появляется возможность сделать кальку адекватной. Автоматизация процесса передачи слов позволит решить сразу несколько задач. Во-первых, увеличить производительность труда лиц, часто сталкивающихся с этой проблемой: переводчиков, операторов, сотрудников почты, других лиц, в обязанности которых входит регулярная обработка иностранных имен собственных в написанном виде. Во-вторых, формализовать процесс транскрипции, что позволит перейти на иной уровень оценки качества передачи, составления правил передачи звучания. И, наконец, в-третьих, закрепить некоторые варианты прочтения, до сих пор вызывающие разногласия в среде лингвистов.

* Работа поддержана РФФИ, грант № 03-01-00353

Практическая транскрипция слов (в частности имен собственных) заключается в передаче слова с одного языка на другой, при которой максимально сохраняется *фонетический* (*звуковой)* облик слова[†]. Прежде всего она применяется при оформлении и обработке машиночитаемых документов.

На данный момент существует ряд систем (транслитераторы, транскрипторы и машинные переводчики), которые отражают три основных метода перевода фамильно-именных групп [1].

- П*еревод*, при котором некоторому часто встречающемуся имени ставится в соответствие его эквивалент, устоявшийся в языке, на который осуществляется перевод, в данный период времени.
- Т*ранскрипция* (точнее *практическая транскрипция*), метод, в котором имени собственному одного языка ставится в соответствие слово другого языка, наиболее точно отражающее его звучание в родном языке.
- Т*ранслитерация* – побуквенная передача имен собственных, записанных с помощью одной графической системы, средствами другой графической системы[‡].

В дальнейшем исходным языком будем называть язык с которого, а языком перевода – на который осуществляется передача имени.

На основе проведенного исследования существующих методов и программных реализаций [2] были сделаны два вывода:

- во-первых, проблема адекватной машинной передачи имен собственных с одного языка на другой не имеет удовлетворительного практического решения;

- во-вторых, для решения этой проблемы следует использовать метод практической транскрипции, так как она дает наиболее приемлемые результаты.

Также выбранный метод более удобен для машинной передачи имен собственных с одного языка на другой, так как два других имеют значительные минусы, а именно: метод перевода возможен лишь при создании чрезвычайно большой и неограниченно пополняемой базы имен, а метод транслитерации не позволяет ориентироваться на фонетический облик конкретного языка.

Основные проблемы

[†] Когда же один и тот же звук можно передать различными буквами/буквосочетаниями, выбирается тот вариант, который максимально отображает *графическую* форму слова.

[‡] Базируясь на каком-либо алфавите, транслитерация допускает условное употребление букв, введение дополнительных и диакритических знаков.

При создании экспериментальной системы машинной транскрипции мы столкнулись с целым рядом проблем.

1. В некоторых странах существует несколько национальных систем транскрипции и транслитерации с национального языка на латиницу, которые зачастую конкурируют. Примером могут служить пиньин и система Уэйда в китайском; ромадзи, кунрэй ("официальная") и система Хёпберна в японском; ГОСТ 16876-71, ISO 9, Библиотеки конгресса Соединенных Штатов, АН СССР, Yellow pages в русском и т.д.

2. В других странах системы транскрипции на кириллицу либо еще совсем не разработаны, либо разработаны, но вызывают много вопросов (т.е. даны лишь основные соответствия, а правильная передача многих буквосочетаний остается неясной). Примером могут служить арабский и турецкий языки.

Сложности и разночтения встречаются даже в английском языке, являющемся сейчас одним из наиболее употребимых мировых языков. В нем правила практической транскрипции, существующие на настоящий момент, опираются на фонетическую транскрипцию, однако историческое развитие английской орфографии привело к ее значительному расхождению с произношением. Из-за этого часто оказывается невозможным определить, какой из возможных вариантов чтения слова оказывается правильным. Так, например, английские сочетания **ou, ow** могут соответствовать дифтонгу [ou] и тогда передаются через **оу:** *Barrow ['bærou] → Бэрроу, Boulder ['bouldэ] → Боулдер*, но также могут выражать (обычно в ударном слоге) и дифтонг [au], в этом случае они передаются соответственно через **ау**: *Founder → Фаундер*. Однако нельзя с уверенностью сказать, когда они читаются так, а когда иначе. В подобных случаях при транскрипции возникают два потенциально возможных варианта имени, выбрать одно из которых не представляется возможным.

3. Помимо сложностей с определением фонетического образа слова, возникает целый ряд затруднений с обозначением звуков, отсутствующих в данном языке. Это приводит к тому, что при создании систем транскрипции приходится ставить в приблизительное соответствие звукам одного языка звуки другого. В результате, зачастую, теряется важная фонетическая информация, такая как длительность, палатализованность, высота тона и др. Разные звуки обозначаются одинаково: звуки [t], [θ] и [ð] передаются буквой «т», [u] и [w] в большинстве случаев передаются русской буквой «у» и др. Из-за этого возникают различные варианты транскрипции: до сих пор, например, не решен окончательно вопрос

о том, передавать ли звук [æ][§] как **а, е** или **э** - у каждого из этих вариантов находятся свои плюсы и минусы [3].

4. Еще одна проблема возникает вследствие отсутствия взаимнооднозначного соответствия при транскрипции слова с исходного языка на язык перевода и обратно. То есть, если слово языка транскрибировать в другой язык, а затем транскрибировать его обратно, то полученное слово в значительном количестве случаев будет отличаться от исходного. Этот же результат возникает в связи с потерей части фонетической информации в ходе самой транскрипции. Данная проблема связана с отсутствием в языке перевода определенных звуков, входящих в фонемный состав языка оригинала.

В соответствии с международными требованиями машиночитаемые документы оформляются латинскими буквами, в связи с чем при транскрипции берется не само слово языка-оригинала со всеми его специфическими буквами и диакритиками, а оно же, но записанное латиницей, что также приводит к потере информации. Так, например, распространенная корейская фамилия Choi (Цой) на русский язык транскрибируется как Чхве. При последующей транскрипции в международных документах указывается написание Chhwe. В результате связь между двумя написаниями имени отсутствует. Китайская фамилия Zhongzhou (Чжунчжоу) после транскрипции на русский язык и последующей записи в международных документах (например, в водительском удостоверении) будет иметь вид Chzhunchzhou.

5. Другая проблема возникает при транскрипции с одного языка (например, английского) имен собственных, исконно принадлежащих другому языку. То есть, если попытаться протранскрибировать «с английского» имя и фамилию мексиканца, например, *Jose Enrique Martinez* (*Хосе Энрике Мартинес*), то по правилам английского языка получится *Джоус (или Джоуз) Энрайк Мартинез*.

6. И последней из проблем, о которых хотелось бы упомянуть в данной статье, является «борьба» между правилами транскрипции, принятыми в настоящее время, и исторической традицией при переводе иностранных имен.[**]

Разработка метода практической транскрипции позволит формализовать и, возможно, закрепить некоторые приемы и правила транскрипции, что, в конечном итоге, приведет к

[§] Буква **а** в закрытом ударном слоге произносится как нечто среднее между русскими буквами **а** и **э**.

[**] В этом списке были перечислены лишь основные проблемы, встающие при осуществлении практической транскрипции при оформлении машиночитаемых документов. Такие же частные случаи как проблемы неблагозвучности и встречающийся в основном в художественной литературе перевод имен по смыслу, здесь не рассматриваются.

решению некоторых указанных проблем. В связи с этим проработка проблемы создания формального метода практической транскрипции и его программная реализация являются чрезвычайно актуальными.

Создание единой фонетической таблицы

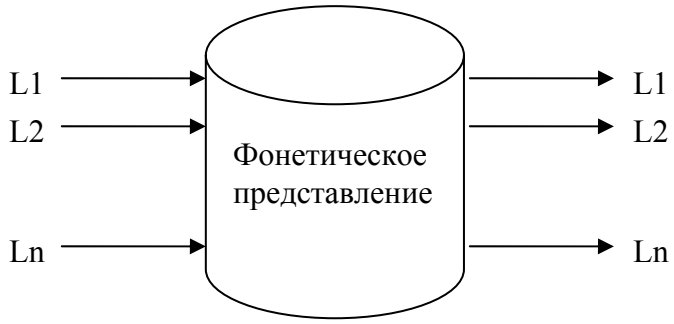
Традиционно практическая транскрипция осуществляется при помощи отдельного алгоритма (и набора правил) транскрипции с каждого языка на каждый язык. Создание подобных правил обязывает лингвиста знать оба языка (как исходный, так и перевода), либо же требует совместной работы двух лингвистов. Поскольку на начальном этапе внедрение этих правил в код программы требует совместной работы как лингвистов, так и программистов, такой подход может являться затруднительным, особенно при большом количестве языков. В связи с этим было принято решение разработать экспериментальную систему машинной транскрипции, работа которой основывается на единой фонетической таблице, что является фактором, в корне отличающим систему от ей подобных. Создание единой фонетической таблицы для языков, между которыми производится транскрипция, позволило намного сократить количество правил транскрипции, работу по их написанию, не ухудшив при этом качество транскрипции.

В существующих системах перевод осуществляется напрямую с исходного языка на язык перевода (что, как отмечалось, требует написания правил транскрипции для каждой такой пары языков).

	L1	L2	…	Ln
L1	-	+		+
L2	+	-		+
…			-	
Ln	+			-

В результате необходимо составить n*(n-1) баз транскрипции, что в ряде случаев может быть невозможно, например в связи с отсутствием соответствующих лингвистов.

Использование единой фонетической таблицы позволяет писать для каждого языка лишь правила с исходного языка в некоторое фонетическое представление (ФП) и обратно. Под фонетическим представлением будем понимать фонетический облик слова, записанный в терминах некоторой фонетической таблицы.



В этом случае для работы необходимо всего 2n баз правил.

Транскрипция при этом осуществляется в два этапа: на первом этапе фамильно-именная группа переводится с языка транскрипции в промежуточное фонетическое представление в соответствии с таблицей, о которой подробнее речь пойдет ниже, а на втором этапе из ФП – в написание на языке перевода. Сама транскрипция осуществляется за счет работы программного «движка», который остается неизменным при присоединении к нему баз данных, содержащих правила транскрипции различных языков. В связи с этим совместная работа программистов и лингвистов требуется лишь на начальных этапах – при создании и отладке программного обеспечения, предназначенного для транскрипции.

Важной задачей при таком подходе является само создание фонетической таблицы, то есть отбор звуков таким образом, чтобы в таблице присутствовали все звуки исследуемых языков, и в то же время ни один звук не был представлен двумя символами. Использование уже имеющихся таблиц вызывает объективные трудности. Представляется невозможным просто представить эту таблицу как пресечение множеств звуков разных языков, во-первых, из-за того, что одни и те же звуки в фонетических системах разных языков обозначаются по-разному (или различные звуки – одинаково), а во-вторых, вследствие того, что в каждом конкретном случае приходится принимать решение, должны ли два похожих, но все же различающихся звука обозначаться в ФП двумя разными символами или одним символом (возможно с разными параметрами). В качестве примеров этих трех случаев можно привести:

1. звуки [n] и [ŋ], соответствующие разным символам фонетической таблицы,
2. английское «л» и немецкое «ль», обозначаемые одинаково как «l», но имеющие разные значения параметра «мягкость/твердость»;
3. французское (дорсо-увулярное) «r» японское (или русское) «р» (апико-альвиолярное), которые в ФП обозначаются одним и тем же символом r.

Прежде чем приступать к созданию таблицы нами был тщательно проанализирован материал различных языков. Ориентируясь при транскрибировании в основном на

фонетическую форму слова, необходимо было одновременно учитывать и орфографический момент, с тем чтобы, не препятствуя правильному чтению, по возможности сохранить при передаче слова близость к его графической форме. Так, например, возник вопрос, следует ли английское [θ] (на письме “th”) и испанское [ȼ] (на письме “c”), похожее на него по звучанию, обозначать одним и тем же символом или нет. Тут вступают в противоречие принципы фонетического и графического подобия. В данном конкретном случае вопрос был решен в пользу их различения (передачи испанского «c» в английском буквой «s») из-за того, что в американских диалектах испанского языка эта буква читается как [s], что сближает ее с графическим написанием в английском.

Помимо этого встает вопрос, необходимо ли учитывать традицию передачи имен или же при транскрипции стоит опираться лишь на фонетический облик слова. Многие фамилии и имена были транскрибированы достаточно давно и в отношении строго определенных людей, оставивших свой след в истории. Однако истории известны примеры, когда людей, принадлежащих к одной семье, транскрибировали в разные периоды времени различным образом. Даже транскрипция имени одного человека может сильно изменяться со временем. Поэтому все говорит в пользу того, чтобы имена современных однофамильцев знаменитых исторических личностей передавать по общим правилам. То есть, *Hamlet, Prince of Denmark* — останется *Гамлетом, принцом Датским*, ибо именно в таком виде он давно уже вошел в русскую культуру и всем знаком. Но его современные тезки будут по-русски Хэмлетами, так как русское орфоэпическое «г» — звук взрывной, а не фрикативный (как английское «h»).

Математическое описание машинной транскрипции

Изложим проблему машинной транскрипции с использованием языка математики.

Здесь мы принимаем, что сама буква, а не только обозначаемый ею звук, обладает некоторыми параметрами (например, гласная/согласная, ряд и так далее). Это необходимо для того, чтобы выяснить, какой звук обозначает данный символ в определенном контексте и какой набор параметров будет соответствовать данному звуку. В противном случае подобная операция представляется затруднительной или трудоемкой.

Определим параметр как пару P=<N,V>, где N – имя параметра, а V – его значение. Параметр будет отображать некоторые характеристики буквы, важные для транскрипции, или позволяющие классифицировать буквы по группам. Например: <”ряд“,

”передний“>,<”тип“, ”гласная“>, <”ударение“, ”безударная“>. Два параметра равны, если совпадают их имена и значения.

Также дадим определение буквы удобное для дальнейшего изложения. Буква состоит из графемы, однозначно идентифицирующей данную букву, и набора параметров, либо изначально присущих данной букве, либо отражающих положение буквы в слове. В связи с этим определим букву как пару S=<C,{P}>, где C – фиксированный символ (графема), обозначающий данную букву, а P – набор ее параметров. При этом будем считать, что различные написания одной и той же буквы (например, строчное и прописное или начальное, срединное, конечное и изолированное) имеют одно и то же обозначение, однако могут обладать (в зависимости от особенностей применения) различными значениями определенных параметров. Набор параметров определяется критичностью различения таких написаний при транскрипции и особенностями языка.

Примером буквы может служить пара <’A’,{<“тип”, “гласн”>, <”написание”, ”прописн”>, <”ряд”, “задний”>}>, где ‘A’ – графема, идентифицирующая данную букву, а множество, заключенное в фигурные скобки – множество параметров данной буквы. Здесь и в дальнейшем выделим с помощью апострофов графемы, относящиеся к символам некоторого языка. Служебные графемы, предназначенные для обеспечения процесса транскрипции, будут обозначаться несколькими символами и не будут заключаться в апострофы.

Определим следующие операторы сравнения букв.

Оператор = производит сравнение как графем букв, так и их наборов параметров. Две буквы S_1 и S_2 равны в смысле оператора = (S_1=S_2), если их графемы совпадают и множество параметров S_2 является подмножеством параметров S_1.

Оператор ≈ производит сравнение только наборов параметров букв. Две буквы S_1 и S_2 равны в смысле оператора ≈ (S_1≈S_2), если множество параметров S_2 является подмножеством параметров S_1. То есть можно сказать, что если S_1≈S_2 и C_1=C_2, то S_1=S_2.

В целом транскрипция будет состоять из двух частей – перевода с языка оригинала на язык-посредник (промежуточную фонетическую таблицу) и перевода с языка-посредника на язык транскрипции. Плюсом такого подхода является сокращение количества наборов правил транскрипции в случае работы со многими языками. Как это было показано выше, при отсутствии языка-посредника приходилось бы создавать базы для транскрипции с каждого языка на все остальные, что составило бы N_L*(N_L-1) баз, где N_L – количество языков, с которыми производится работа. При транскрипции через язык-посредник это

количество составит лишь 2* N_L, так как потребуются базы лишь для транскрипции на язык-посредник и с него.

Однако подобный подход налагает дополнительные требования на язык-посредник. Алфавит языка-посредника должен содержать звуки всех языков, с которых производится транскрипция. Кроме алфавита для языка-посредника должен определяться набор параметров, которыми могут обладать буквы этого языка. Для того, чтобы корректно произвести транскрипцию, правила транскрипции с языка-посредника должны охватывать все буквы алфавита этого языка, что несколько увеличивает объем правил. Одновременно с этим за счет проведения дополнительных работ скорость транскрипции падает.

Также имеется необходимость определить алфавит каждого языка с тем, чтобы сопоставить любому символу, встречающемуся в данном языке, букву из этого алфавита (графему и набор параметров).

В целом, процесс транскрипции разобъем на пять этапов:

1. преобразование написания слова на исходном языке во внутреннее представление;
2. выделение слогов, расстановка переносов и ударений;
3. перевод внутреннего представления слова в ФП;
4. перевод ФП слова во внутреннее представление слова на языке перевода;
5. преобразование внутреннего представления слова на языке перевода в написание слова на языке перевода.

Под промежуточным представлением здесь понимается термин из программирования, означающий формат записи внутренней информации программы в памяти.

Опишем каждый из этих этапов подробнее

1. Преобразование написания слова на языке оригинала во внутреннее представление состоит в преобразовании слова языка, записанного как множество символов W={G}, во множество букв W'={S}. Здесь G – символ (знак), а в случае машинной транскрипции - информационный код знака в одной из компьютерных кодировок (ASCII, ANSI или иной другой). Для такого преобразования вводится множество правил, называемых правилами алфавита, сопоставляющих символу (информационному коду знака) G букву S. $\mathfrak{R}_a=\{R_a\}$, где $\mathfrak{R}_a$ – множество правил алфавита, а $R_a=<G,S>$ – правило. При машинной транскрипции все множества правил хранятся в некоторых базах, называемых в дальнейшем базами правил.

Примерами правил алфавита может служить следующее множество.

<’A’,<*’A’,**{<“тип”, “гласн”>, <”написание”, ”прописн”>, <”ряд”, “задний”>}*>>

<’a’,<*’A’,**{<“тип”, “гласн”>, <”написание”, ”строчн”>, <”ряд”, “задний”>}*>>

<’B’,<*’B’,**{<“тип”,“согласн”>,<”написание”,”прописн”>,<”звонкость”,“звонкая”>}*>>

<’b’,<*’B’,**{<“тип”,“согласн”>,<”написание”,”строчн”>,<”звонкость”,“звонкая”>}*>>

Курсивом здесь выделена часть, относящаяся к букве (S), а полужирным шрифтом – параметры буквы.

Для всех графем входного слова последовательно находятся такие правила, что графема, входящая во входное слово W, совпадает с графемой из найденного правила. Внутреннее представление слова W’ получается путем последовательной конкатенации букв, входящих в полученные правила. Кроме того, в начало и конец слова добавляются специальные буквы, обозначающие начало и конец слова. Все графемы, для которых не было найдено соответствия в правилах алфавита, считаются знаками препинания и передаются дальше без изменений с соответствующей пометкой. Перед началом группы знаков препинаний ставится буква конца слова, после нее – начала слова. Подобный подход позволяет вычленить не только знаки препинания, но и символы из других алфавитов, которые не должны транскрибироваться в рамках данного языка.

Таким образом $W \Rightarrow W' = \bigcup_{m=1}^{N_{W'}} S_m$, причем

a) $S_1 = <BEG, \{\}>$,

b) $S_N = <END, \{\}>$, здесь BEG и END – графемы, обозначающие начало и конец слова,

c) $S_m = S$, если $\exists (R_a = <G,S> \in \Re_a : G = G_j)$, здесь j=1..M, где M – общее количество знаков во входном слове, причем j не убывает при увеличении m,

d) $S_m = <G_j, \{\}>$, если не$\exists$ $R_a = <G,S> \in \Re_a : G = G_j$,

e) $S_m = <BEG, \{\}>$, если S_{m-1} получено по правилу d), а S_{m+1} получено по правилу c),

f) $S_m = <END, \{\}>$, если S_{m-1} получено по правилу c), а S_{m+1} получено по правилу d),

Здесь $m \in (1, N_{W'})$, где $N_{W'}$ – общее количество букв в выходном слове (во внутреннем формате).

2. Выделение слогов и расстановка ударений производятся для того, чтобы определить закрытые/открытые слоги и ударные/безударные буквы. Любая буква, находящаяся в конце слога, приобретает дополнительный параметр «буква в слоге» со значением «открытая». Для остальных букв значение этого параметра – «закрытая».

Выделение слогов производится по следующему алгоритму. Для алфавита каждого языка может быть задан набор слогообразующих букв. В качестве части слога, присоединяемой к слогообразующей букве, берется половина букв между двумя слогообразующими. При нечетном количестве букв, средняя передается следующему слогу. Исключение делается для приставок, суффиксов и окончаний, разделение на слоги которых фиксировано. Они присоединяются к остальной части слова как отдельный слог или несколько выделенных фиксированным образом слогов. Написание и деление на слоги таких приставок, суффиксов и окончаний задается отдельной базой правил.

Расстановка ударений, как и выделение слогов, не является обязательной. Их необходимо производить для языков, в которых буквы читаются различным образом в зависимости от того, в какой позиции находится данная буква – в ударной или безударной, в конце слога или нет.

Для расстановки ударений в языках, где оно является критичным, фиксируется номер слога и направление, в котором ведется счет слогов – от начала или от конца слова. В случае, если в слове меньше слогов, чем указанный номер, ударение ставится на последний встретившийся слог.

3. Задачей ***перевода внутреннего представления слова в промежуточное фонетическое написание*** является приведение слов различных языков к единой записи в рамках алфавита фонетической таблицы. На вход данного этапа поступает последовательность букв языка. Выходом этапа является набор фонем, входящих в состав фонетической таблицы.

Под строкой (словом) здесь будем понимать упорядоченное множество букв. Подстрокой слова будет являться подмножество последовательно идущих букв данного слова. Обозначим через W_l^i подстроку слова W длиной l, начинающуюся с буквы в позиции i. В дальнейшем верхний индекс подстроки будет обозначать позицию, с которой начинается данная подстрока в слове, а нижний индекс будет обозначать длину подстроки. Символом * будем обозначать произвольное значение позиции.

Под правилом перевода будем понимать пару $R_t=<W_{l1}^*, \overline{W}_{l2}>$, где W_{l1}^* - строка-образец, а $\overline{W}_{l2}$ - строка-результат. Правило R применимо к подстроке W_{l1}^i, если строка-образец сравнима с W_{l1}^i. Под сравнимостью понимается нахождение равенства букв из W_{l1}^* и W_{l1}^i в одних и тех же позициях строки W_{l1}^* и подстроки W_{l1}^i. При этом здесь две буквы S_1

и S_2 равны, если $S_1=S_2$ или $S_1 \approx S_2$. Подробный алгоритм определения применимости правила к строке приведен ниже.

Под переводом подстроки W_{l1}^i будем понимать функцию $\overline{W}_{l2} = F^t(W_{l1}^i)$, такую, что $\exists R_t = <W_{l1}^*, \overline{W}_{l2}> \in \Re_t$ применимое к W_{l1}^i. Здесь $\Re_t=\{R_t\}$ - база правил перевода.

Задача перевода в промежуточное фонетическое написание в этом случае может быть представлена следующим образом.

Пусть имеем на входе на данный этап некоторое слово $W=<S_1, S_2, ..., S_a>$ и набор правил перевода $\Re_t$. Перевод внутреннего представления в промежуточное фонетическое написание в этом случае будет заключаться в нахождении и применении упорядоченного подмножества правил $\Re = <W_l^*, \overline{W}_{l2}>$, таких что:

1) $i=<i_1, i_2, ..., i_n>$, где n – число правил в подмножстве $\Re$;
2) $l=<l_1, l_2, ..., l_n>$;
3) $\sum_{j=1}^{n} l_j = a$;
4) $i_1=1$;
5) $i_{k+1}=i_k+l_k$ для k<n и $i_n+l_n=a+1$;
6) $\forall k,m \; \exists R_t = <W_k^*, \overline{W}_{k2}> : \exists \; \overline{W}_{k2} = F^t(W_k^m)$.

Здесь множество i – это множество позиций с которых применимы правила, а множество l – множество длин подстрок.

В случае, если такого набора правил не существует, транскрипция считается неуспешной. В этом случае можно попытаться подобрать такой набор правил, что количество разрывов (непереведенных букв) во входном слове минимально.

Результатом перевода будет являться конкатенация результатов последовательного применения правил перевода.

$$\overline{W} = \bigcup_{i,l} F^t(W_l^i)$$

Проверка применимости правила к строке производится следующим образом. Правила могут содержать в себе буквы со специально определенной графемой EMPTY. Сравнение буквы правила и буквы строки производится при помощи оператора ≈, если графема буквы правила равна EMPTY, и при помощи оператора = в противном случае.

В начале перевода внутреннего представления слова в промежуточное фонетическое написание текущая позиция во входной строке устанавливается в 1. Далее, до тех пор пока не будет достигнут конец слова, последовательно применяется следующий алгоритм.

Сохраняем текущую позицию. Далее пытаемся найти все правила, применимые для строки, начинающейся с текущей позиции. Если первые несколько последовательно идущих букв в правиле имеют графему, равную EMPTY, то уменьшаем текущую позицию на количество таких букв. Если текущая позиция меньше 1, то считаем, что правило не применимо, восстанавливаем текущую позицию и переходим к следующему правилу.

Начиная с полученной текущей позиции, последовательно сравниваем буквы строки и правила. Если хотя бы одна буква строки не равна соответствующей букве правила, то считаем, что правило не применимо, восстанавливаем текущую позицию и переходим к следующему правилу. Если сравнение всех букв прошло успешно, то считаем, что правило применимо. В этом случае помещаем сохраненную текущую позицию в множество *i*. Во множество *l* помещаем количество букв в правиле за вычетом последовательно идущих букв в начале и в конце правила, имеющего графему равную EMPTY. В случае, если к одной и той же позиции в слове применимо несколько правил, то для каждого правила на основе существующих заводятся свои множества *i* и *l*, после чего в них помещается текущая позиция и количество букв. Далее восстанавливается сохраненная текущая позиция и алгоритм переходит к следующему правилу.

По окончании перебора всех правил текущая позиция увеличивается на величину, сохраненную в множестве *l*.

4. Этап ***перевода промежуточного фонетического написания слова во внутреннее представление слова на языке транскрипции*** аналогичен этапу 3, но имеет противоположные задачи. Он служит для того, чтобы сформировать последовательность букв, отражающих полученное звучание слова в языке транскрипции. Работа этапа осуществляется по тем же принципам, что и этапа 3. Здесь правила являются не столь многозначными, как на этапе 3, так как при создании множества правил $\mathfrak{R}_t$ имеется возможность задать одно определенное правило для передачи данного набора звуков при наличии альтернативы.

5. Преобразование внутреннего представления слова на языке транскрипции в написание слова на языке транскрипции является обратным относительно этапа 1. Здесь могут использоваться те же самые правила, что и на этапе 1, так как в большинстве случаев должно существовать взаимнооднозначное соответствие между графемой и

буквой с данным набором параметров. Буквы с графемами BEG и END удаляются, знаки препинания передаются соответствующими символами.

Предложенный метод позволяет формально подойти к проблеме машинной транскрипции в многоязыковых системах. Это позволит строго сформулировать требования к языку-посреднику и языкам, участвующим в транскрипции, исследовать их особенности и свойства. Формализация процесса транскрипции упрощает переход к решению задачи машинной транскрипции.

Программа машинной транскрипции «ТрансСкриба»

На основе приведенной выше модели нами была создана программа машинной транскрипции «ТрансСкриба». Данная программа предназначена для транскрипции иностранных имен на русский язык. На данный момент были подготовлены базы правил для транскрипции со следующих языков: английский, немецкий, французский, испанский, итальянский, шведский, польский, китайский (пиньин и система Уэйда), японский (ромадзи, система Хёпберна и система нашей собственной разработки, разрешающая наличие символов латиницы, отсутствующих в двух предыдущих), корейский (северный, южный и старый вариант записи), вьетнамский, арабский и турецкий. Также были проведены эксперименты по транскрипции с приведенных выше языков на, например, английский, немецкий, французский и испанский.

Однако разработанная система имеет существенные ограничения при транскрипции на слоговые языки, такие как японский и китайский, так как перед транскрипцией на них слово должно быть записано средствами данного языка, то есть с использованием его

слогов. Но в настоящий момент существуют только приблизительные рекомендации по записи произвольного слова с помощью фиксированного набора слогов, а формальной методики, позволяющей получить подобный результат, не существует.

Нами было проведено тестирование полученной системы. Для проверки каждого из входных языков транскрибировалось на русский язык от 1 до 5 тысяч слов. При этом количество полученных ошибок не превышало 1%. На данный момент ведется проверка качества транскрипции на другие языки.

Список литературы

1. Реформатский А.А. Введение в языкознание. Гл. 3. Фонетика. М.: Аспект Пресс, 1996;
2. // Вестник ВИНИТИ НТИ серия 2. 2004 г., № 4, сс.
3. Трубецкой Н.С. Основы фонологии. М.: НЛ, 1960.