

КЛАССИФИКАЦИЯ ОТЗЫВОВ ПОЛЬЗОВАТЕЛЕЙ С ИСПОЛЬЗОВАНИЕМ ФРАГМЕНТНЫХ ПРАВИЛ

Васильев В. Г. (wg_2000@mail.ru),
Худякова М. В. (mariya.kh@gmail.com),
Давыдов С. (davydov_sergey@hotmail.com)

ООО «ЛАН-ПРОЕКТ», Москва, Россия

В работе рассматривается подход к анализу отзывов пользователей, основанный на задании правил классификация и выделения значимых фрагментов на специальном языке. Проводится анализ эффективности автоматического построения и коррекции правил путем обучения на примерах. Приводятся результаты экспериментов в рамках соответствующей дорожки РОМИП 2011.

Ключевые слова: анализ отзывов пользователей, классификация, фрагментные правила

1. Введение

В настоящее время в связи с активным развитием социальных сетей, форумов и блогов вопросы автоматизации анализа мнений пользователей сети по различным вопросам (отношение к товарам и услугам, событиям, высказываниям, сообщениям) вызывают большой интерес у многих организаций, что приводит к активизации научных исследований и экспериментов в данной области. Обычно задача анализа мнений пользователей ставится как задача классификации текстов на два или более класса, которые разделяют мнения на позитивные и негативные, а также их оттенки.

В работе [1] рассматриваются подходы к классификации отзывов на фильмы, основанные на сравнении числа положительных и отрицательных слов с учетом усиливающих терминов, а также использовании стандартного классификатора на основе машин опорных векторов. Как показывают проведенные авторами эксперименты при использовании первого подхода F-мера достигается порядка 60%–70%, а при использовании второго подхода порядка F-мера порядка 80%–85%. В работе [2] также как и в предыдущей работе рассматривается использование словарного подхода и обучения на примерах с использованием SVM. При этом реализована итерационная процедура пополнения словарей положительных и отрицательных терминов за счет классификации неразмеченных текстов. В целом эксперименты на массиве отзывов о различных товарах на китайском языке достигаются значения F-меры порядка 85%–90%. В работе [3] для классификации отзывов все предложения

(высказывания) в тексте предварительно разбивают на личные и нейтральные и осуществляют построение трех классификаторов, которые обучаются на личных, нейтральных и всех предложениях с использованием метода SVM. Как показывают авторы такой подход позволяет несколько повысить качество по сравнению с базовым уровнем. В работе [4] приводится пример построения кросс языкового классификатора для анализа отзывов пользователей. Обучающая выборка представлена на английском языке, а обрабатываются переводы отзывов с китайского языка. Для обучения классификатора используется метод SVM и процедура использования неразмеченных текстов. В целом на отзывах о различных цифровых устройствах авторами были получены значения F-меры порядка 75%–80%. В работе [5] в отличие от предыдущих рассмотренных работ рассматривается задача классификации не отзывов о товарах, а мнений политиков о поправках к законам и результатов голосования. Помимо словарного и векторного подхода (метод SVM) для анализа отзывов в ряде работ строятся специальные вероятностные модели. Например, в [6] учитывается дерево синтаксического разбора предложений и зависимости между словами, а в работе [7] строится совместная тематико-оценочная вероятностная модель. Также в ряде работ авторы явно задают правила оценки текстов. В частности, в работе формулируются различные правила для определения области действия инверсных слов типа «не».

Таким образом, в работах по классификации отзывов применяются как стандартные методы классификации текстов, так и модифицированные методы, в которых учитывается возможная инверсия значений оценочных слов, синтаксическая структура предложений, зависимости между словами [6]. Целью настоящей работы является исследование эффективности использования стандартных методов классификации текстов основанных на задании правил и обучении на примерах применительно к задаче классификации отзывов на русском языке, а также определение перспективных направлений совершенствования и развития данных алгоритмов. При этом в качестве основного рассматривается подход к классификации отзывов на основе правил, сформированных экспертами. Оценка эффективности рассматриваемых методов производится в рамках дорожки классификации отзывов пользователей на два класса.

2. Описание используемых подходов

2.1. Классификация на основе правил

Для задания правил в данной работе применяется подход, описанный в работе [8]. В данном случае оцениваемый текст D рассматривается как последовательность элементов (слов, цифр, знаков препинания), т. е. $D = (d_1, \dots, d_n)$, где $d_i \in T$ — отдельный элемент текста, $T = (t_1, \dots, t_m)$ — множество всех допустимых элементов, n — длина текста, m — число различных допустимых элементов текстов.

Множество $F = \{(p, q) \mid 1 \leq p \leq q \leq n\}$ будем называть множеством всех фрагментов текста длины n . Фрагментами текста будем называть отдельные элементы данного множества $f = (f_l, f_r) \in F$, которые задают левую f_l и правую f_r границы фрагмента (номер начального и конечного элемента текста). Результатом выполнения произвольного правила Q для текста D является множество $F_Q \subset F$, содержащее все фрагменты удовлетворяющие правилу Q . При этом, если $F_Q \neq \emptyset$, то будем говорить, что текст D удовлетворяет правилу Q .

Операции для задания правил можно разбить на следующие группы:

- элементарные — выделяют фрагменты, соответствующих отдельным словам;
- сложные — выделяют сложные многословных выражений;
- определяющие — задание общих понятий и множеств;
- управляющие — задают параметры классификации и обучения на примерах.

Элементарные операции — выделяют отдельные слова в тексте, предложения, строки, разделы документа. Например, правило $\$FirstUp$ — выделяет все слова в тексте с большой буквы, правило Липецкая — все слова, являющиеся словоформами слова «липецкая», правило «обл*» — все слова начинающиеся на «обл»; $\$Sentence$ — все предложения в документе, $\#section$ — раздел документа с определенным именем (например, заголовок).

Сложные операции — задания преобразования множеств фрагментов. Приведем примеры определения отдельных операций для построения сложного правила Q на основе правил Q_1, \dots, Q_k .

$Q = Q_1 \vee Q_2$ — бинарная операция ИЛИ, $F_Q \equiv R(F_{Q_1} \vee F_{Q_2})$, $F_{Q_1} \vee F_{Q_2} = \{f \in F \mid \exists f_1 \in F_{Q_1}, f \supseteq f_1 \text{ или } \exists f_2 \in F_{Q_2}, f \supseteq f_2\}$. Например, правило *искажение блеклый неуклюжий тьфу* выделяет фрагменты, равные соответствующие отдельным словам.

$Q = Q_1 \Delta_{n_1} Q_2$ — бинарная операция И с ограничением на расстояние между фрагментами, $F_Q \equiv R(F_{Q_1} \Delta_{n_1} F_{Q_2})$, $F_{Q_1} \Delta_{n_1} F_{Q_2} = \{f \in F \mid \exists f_1 \in F_{Q_1} \text{ и } \exists f_2 \in F_{Q_2}, \text{ т. что } f \supseteq f_1, f \supseteq f_2 \text{ и } d(f_1, f_2) \leq n_1\}$. Например, правило *смазанные &3 образы*, выделяет фрагменты, где расстояние между «смазанный» и «образ» не более 3 слов.

$Q = Q_1 \square_{n_1, n_2} Q_2$ — бинарная операция последовательности с ограничением на расстояние между фрагментами, $F_Q \equiv R(F_{Q_1} \square_{n_1, n_2} F_{Q_2})$, $F_{Q_1} \square_{n_1, n_2} F_{Q_2} = \{f \in F \mid \exists f_1 \in F_{Q_1} \text{ и } \exists f_2 \in F_{Q_2}, \text{ т. что } f_1 < f_2, d(f_1, f_2) > 0, f \supseteq f_1, f \supseteq f_2 \text{ и } n_1 \leq d(f_1, f_2) \leq n_2\}$. Например, *отказаться :3 (снимать производство)*, выделяет фрагменты, в которых после «отказаться» на расстоянии 3 слов находятся слова «снимать» или «производство».

$Q = \bowtie(Q_1, \dots, Q_k)$ — множественная операция последовательности соседних элементов (осуществляет отбор смежных фрагментов), $F_Q \equiv R(\bowtie(F_{Q_1}, \dots, F_{Q_k}))$, $\bowtie(F_{Q_1}, \dots, F_{Q_k}) = \{f \in F \mid \exists f_i \in F_{Q_i}, i=1, \dots, k, \text{ т. что } f_1 < f_{i+1}, d(f_i, f_{i+1})=1 \text{ для } i=1, \dots, k-1 \text{ и } f \supseteq f_i \text{ для } i=1, \dots, k\}$. Например, правило «начальник руководитель директор» («главное управление» управление организация отдел) (МВД МЧС МинФин) — выделяет словосочетания соответствующие руководителям различных ведомств.

$Q = Q_1 \wp Q_2$ — бинарная операция нахождения пересечения фрагментов, $F_Q \equiv \{f \in F_{Q_1} \mid \exists f_1 \in F_{Q_1} \wedge f \in F_{Q_2}\}$. Например, правило [великая \$FirstUp] — выделяет слова «великая», которые написаны с большой буквы.

$Q = Q_1 \triangleleft_{n_1, n_2}$ — унарная операция ограничения длины фрагмента, $F_Q \equiv \{f \in F_{Q_1} \mid n_1 \leq |f| \leq n_2\}$. Например, правило (Нижегородская & Владимирская) #IN #INTERVAL(2w/3w) — выделяет фрагменты, содержащие заданные слова длиной от 2 до 3 слов.

Для возможности построения правил включающих отрицания и условные операторы (наличие выражения проверяется, но оно не включается в итоговый фрагмент) используются специальные варианты бинарных правил, в которых один из операндов считается отрицательным или условным. В частности, символом \square_{n_1, n_2}^+ обозначается операция нахождения последовательности, в которой второй операнд берется с отрицанием, символом \square_{n_1, n_2}^- — операция, в которой первый операнд берется с отрицанием, $\square_{n_1, n_2}^{\#}$ — операция, в которой первый операнд является условным. Определение $\square_{n_1, n_2}^{\#}$ имеет следующий вид $Q = Q_1 \square_{n_1, n_2}^{\#} Q_2$, где $F_Q \equiv \{f \in F_{Q_1} \mid \exists f_2 \in F_{Q_2} \text{ т. что } f < f_2, 0 < n_1 \leq d(f, f_2) \leq n_2\}$.

Например, операция без ^:3 отличн* — выделяет слова, начинающиеся на «отличн» перед которыми нет слова «без».

Определяющие операции — задают понятия в форме шаблонных подстановок (#define) и в форме сохраненных множеств фрагментов (#set). Для обращения к подстановке и множеству фрагментов используются операторы @ и @@. Например, операция

#define Bad плохой глупый

задает понятие Bad, к которому можно обращаться из текста правила с помощью выражения @Bad.

Управляющие операции — задают параметры классификации и обучения. Например, операция #option train задает необходимость автоматического формирования правил путем обучения на примерах.

Для классификации отзывов был разработан набор понятий, из которых были сформированы правила для выделения положительных и отрицательных отзывов. Например, правило для определения отрицательных отзывов имеет следующий вид

@CheckBadBegin

@Bad & ^ (@CheckGoodBegin @CheckBadBegin @Good)

Оно работает следующим образом, сначала проверяются первые два предложения на содержание оценочных слов с использованием правила @CheckBadBegin, а затем проверяется наличие отрицательных слов при условии, что не найдены в начале текста положительные или отрицательные оценки.

Правило проверки на отрицательность начала текста проверяет, что в первых двух предложениях от начала документа перед отрицательным

фрагментом нет положительного фрагмента и слова «не» или двойных кавычек. При этом понятие `@@isbad` является более строгим, а понятие `@@badmark` менее строгим.

Правило для проверки на отрицательность текста целиком `@Bad` является более сложным, но в целом похожим на `@CheckBadBegin`. Основу правил составляют понятия `@@isgood`, `@@isbad`, `@@goodmark`, `@@badmark`, которые выделяют множества исходных положительных и отрицательных фрагментов без учета модификаторов перед ними. Определение каждого такого понятия включает около сотни выражений.

В целом алгоритм оценки отзыва при использовании построенных правил имеет следующий вид.

Алгоритм 1. Классификация отзывов на основе правил

- Шаг 1. Проверка начала текста, если решение однозначно, то завершить работу.
- Шаг 2. Проверка текста в целом, если решение однозначно, то завершить работу.
- Шаг 3. Вычисление веса положительных и отрицательных фрагментов;
- Шаг 4. Отнесение текста к классу с наибольшим весом.

Построение правил классификации вручную является достаточно трудоемкой процедурой. По этой причине в используемом языке имеются операции, которые позволяют уточнить ранее построенное правило путем анализа результатов классификации обучающей подборки документов. В частности, правило для классификации отрицательных фрагментов было изменено следующим образом.

```
@CheckBadBegin  
(@Bad @AutoSupplementQuery) & ^ (@CheckGoodBegin @CheckBadBegin)
```

```
#define AutoSupplementQuery $True
```

В приведенном правиле понятие `@AutoSupplementQuery` вычисляется автоматически таким образом, чтобы максимально повысить полноту правила, без снижения точности. Для формирования данного правила используется модифицированный вариант жадного алгоритма построения решающего списка [9], в котором в качестве множества положительных примеров используются неправильно классифицированные отрицательные тексты, а в качестве множества отрицательных примеров все положительные тексты.

Формирование обновленного правила происходит в соответствии со следующей схемой.

Алгоритм 2. Формирование обновленного правила

- Шаг 1. Выполнить классификацию обучающего множества с помощью правила, в котором `@AutoSupplementQuery` не задан.

Шаг 2. Выполнить оценку качества классификации и построить @ *AutoSupplementQuery* с использованием модифицированного варианта жадного алгоритма построения решающего списка.

Шаг 3. Выполнить дополнительную коррекцию построенного правила экспертом.

После построения модифицированного правила классификация отзывов происходит с использованием алгоритма 1.

2.2. Классификация с использованием обучаемых алгоритмов

В настоящее время разработано большое количество алгоритмов машинного обучения для решения задач классификации текстов. В данной работе было решено провести тестирование следующих стандартных алгоритмов [10]:

- алгоритм k-ближайших соседей;
- алгоритм построения деревьев решений C4.5;
- алгоритм на основе машин опорных векторов;
- байесовский классификатор на основе смеси многомерных нормальных распределений;
- байесовский классификатор на основе смеси распределений фон Мизеса-Фишера;
- центроидный классификатор Роччио.

Общая схема алгоритма обучения в данном случае является достаточно стандартной и имеет следующий вид.

Алгоритм 3. Обучение классификатора на примерах

1. Формирование векторного представления текстов в рамках модели «Bag Of Words».
2. Снижение размерности (селекция признаков по частоте) и вычисление весов признаков (TF_IDF).
3. Обучение и оценка классификатора на обучающей выборке с использованием 5-шаговой процедуры кросс-проверки.

3. Эксперименты

3.1. Описание тестовых массивов и показателей качества

В данной работе эксперименты по оценке качества проводились в рамках дорожки РОМИП 2011 классификации отзывов на два класса. Данная дорожка содержала три обучающих массива текстов:

- массив отзывов о фильмах — содержит 15 718 текстов, предоставленных онлайн-сервисом рекомендаций IMHONET, каждый отзыв оценен по 10-балльной шкале;
- массив отзывов о книгах — содержит 24 159 текстов, предоставленных онлайн-сервисом рекомендаций IMHONET, каждый отзыв оценен по 10-балльной шкале;
- массив отзывов о цифровых фотоаппаратах — содержит 10 370 текстов, предоставленных Yandex, каждый отзыв оценен по 5-балльной шкале.

Для тестирования использовался набор из 16 821 текстов, содержащих описание различных объектов интереса пользователей. Задачей дорожки было отнести каждый текст к классу положительных, либо к классу отрицательных отзывов.

Для оценки качества работы классификаторов в настоящей работе использовались следующие стандартные показатели качества: точность, полнота, F1-мера, аккуратность и среднее евклидово расстояние. Для первых трех показателей вычислялись значения, как для отдельных классов, так и макро-оценки.

3.2. Результаты экспериментов

Эксперименты проводились в два этапа. На первом этапе была выполнена самооценка качества классификации с использованием обучающего множества текстов, предоставленного организаторами дорожки. На втором этапе была выполнена обработка тестового множества текстов с использованием отдельных классификаторов и получены оценки качества от организаторов дорожки.

В следующей таблице приведены результаты самооценки качества, полученные с использованием классификаторов на основе правил. При этом классификатор на основе ручных правил обозначен Q1, а классификатор на основе обученных правил Q2.

Таблица 1. Результаты самооценки качества классификации с использованием правил

Классификатор	Объект	Точность положительные	Полнота положительные	Точность отрицательные	Полнота отрицательные
Q1	book	65%	66%	85%	43%
Q1	camera	71%	86%	83%	77%
Q1	film	60%	64%	71%	35%
Q2	book	71%	62%	84%	56%
Q2	camera	69%	88%	83%	81%
Q2	film	61%	67%	72%	37%

Как можно заметить из приведенной таблицы 1 использование процедуры обучения повышает полноту классификации на обучающем множестве, но при этом снижает немного точность.

Также были проведены эксперименты по оценке качества работу обучающих алгоритмов. В следующей таблице, в качестве примера, приведены показатели качества для массива отзывов о книгах. В таблице 2 используются следующие обозначения алгоритмов: SVM — классификатор машин опорных векторов, GMM — байесовский классификатор на основе смеси многомерных нормальных распределений, ROC — классификатор Роччио, KNN — классификатор k-ближайших соседей, VMF — классификатор фон Мизеса-Фишера, TREE — классификатор на основе деревьев решений.

Таблица 2. Результаты оценки качества для массива отзывов о книгах

Классификатор	Объект	Точность положительные	Полнота положительные	Точность отрицательные	Полнота отрицательные
SVM	book	86%	99%	41%	44%
GMM	book	88%	73%	27%	42%
ROC	book	92%	18%	27%	8%
KNN	book	87%	78%	23%	30%
VMF	book	94%	47%	31%	57%
TREE	book	90%	70%	27%	30%

Как можно заметить из приведенной таблицы показатели качества для отрицательных текстов при использовании обучающих алгоритмов заметно хуже, чем при классификации на основе правил. При этом наиболее высокие показатели продемонстрировали алгоритмы: SVM, KNN, TREE. Алгоритм SVM и так достаточно часто используется в различных работах, по этой причине было решено отправить организаторам конкурса результаты обработки тестового массива с помощью алгоритмов KNN и TREE.

Организаторами дорожки для уменьшения субъективности оценок экспертов были рассмотрены 2 схемы оценок качества:

- схема И — учитываются только те отзывы, для которых совпадают оценки экспертов.
- схема ИЛИ — ответ алгоритма считается правильным, если он совпадает с ответом одного из экспертов.

Результаты экспериментов по каждой схеме приведены в следующих двух таблицах. В данные таблицы включена наилучшая оценка по дорожке и результаты оценки качества для 4 прогонов: Q1 — классификатор на основе правил, Q2 — модифицированный классификатор на основе правил, Q3 — классификатор на основе деревьев решений, Q4 — классификатор k-ближайших соседей.

Таблица 3. Результаты оценки качества в соответствии со схемой И

Метод	Объект	Макро-Точность	Макро-Полнота	Макро-F1
Q1	book	0.53	0.58	0.53
Q2	book	0.55	0.66	0.58
Q3	book	0.52	0.54	0.53
Q4	book	0.54	0.51	0.51
xxx-20	book	0.96	0.61	0.67
Baseline	book	0.46	0.5	0.48
Q1	camera	0.81	0.88	0.84
Q2	camera	0.79	0.87	0.83
Q3	camera	0.50	0.47	0.48
Q4	camera	0.93	0.54	0.53
xxx-24	camera	0.91	0.93	0.92
Baseline	camera	0.42	0.5	0.45
Q1	film	0.67	0.70	0.68
Q2	film	0.66	0.70	0.68
Q3	film	0.54	0.53	0.50
Q4	film	0.54	0.52	0.52
xxx-23	film	0.76	0.78	0.77
Baseline	film	0.42	0.5	0.45

Таблица 4. Результаты оценки качества в соответствии со схемой ИЛИ

Метод	Объект	Макро-Точность	Макро-Полнота	Макро-F1
q1	book	0.56	0.62	0.56
q2	book	0.57	0.69	0.61
q3	book	0.52	0.55	0.47
q4	book	0.54	0.51	0.51
xxx-20	book	0.73	0.74	0.73
Baseline	book	0.46	0.5	0.48
q1	camera	0.83	0.90	0.86
q2	camera	0.83	0.89	0.85
q3	camera	0.53	0.52	0.51
q4	camera	0.93	0.54	0.53
xxx-24	camera	0.92	0.94	0.93
Baseline	camera	0.43	0.5	0.48
q1	film	0.69	0.73	0.71
q2	film	0.68	0.73	0.70

Метод	Объект	Макро-Точность	Макро-Полнота	Макро-F1
q3	film	0.56	0.57	0.53
q4	film	0.54	0.53	0.53
xxx-23	film	0.78	0.80	0.79
Baseline	film	0.42	0.5	0.46

Анализ результатов, приведенных в таблицах 3 и 4, позволяет сделать следующие выводы. Методы классификации на основе правил показали более высокое качество работы. Значительно лучше обрабатывается массив с камерами, что связано с тем, что первоначальная настройка правил делалась именно на нем. Обучаемые методы показали низкое качество работы возможно по следующим причинам: учитывались все признаки в текстах, не учитывался контекст употребления слов, методы на основе деревьев решений и классификатор ближайших соседей на обучающей выборке работали хуже метода SVM.

4. Выводы

Таким образом, в настоящей работе рассмотрены несколько подходов к классификации отзывов пользователей. Наиболее эффективным оказался подход, основанный на ручном построении правил экспертами. Использование традиционных методов обучения на примерах, а также расширения запросов с помощью отдельных терминов не приводит к высокому качеству классификации. Это связано, по-видимому, с тем, что в стандартных методах используется теоретико-множественная модель текстов, в которой не учитывается контекст употребления слов.

В качестве перспективных направлений дальнейших исследований можно сформулировать следующие: реализация специальных обучающих алгоритмов для формирования контекстных правил для заданных пользователем оцениваемых объектов, что позволит значительно снизить трудоемкость формирования правил экспертами; выполнение обучения классификаторов не на полных текстах, а на отдельных предложениях, содержащих ссылки на оцениваемый объект; использование при обучении на примерах только словарных признаков, отобранных экспертами; задание весов различным терминам при формировании правил экспертами и реализация специальных инструментальных средств для упрощения работы экспертов-лингвистов по формированию правил.

References

1. *Kennedy A., D. Inkpen* (2006) Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, Vol.22, No 2, pp. 110–125.
2. *Qiu L., Zhang W., Hu C., Zhao K.* SELC: a self-supervised model for sentiment classification. *Proceedings of the 18th ACM conference on Information and knowledge management (CIKM '09)*, New York, USA, 2009, pp. 929–936.
3. *Li S. et al.* Employing Personal/Impersonal Views in Supervised and Semi-supervised Sentiment Classification. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 2010, pp. 414–423.
4. *Wan X.* Co-Training for Cross-Lingual Sentiment Classification. *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, 2009, pp. 235–243.
5. *Thomas M., Pang B., Lee L.* Get out the vote: Determining support or opposition from congressional floor-debate transcripts. *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 2006, pp. 327–335.
6. *Nakagawa T., Inui K., S. Kurohashi.* Dependency Tree-based Sentiment Classification using CRFs with Hidden Variables. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL*, 2010, pp. 786–794.
7. *He Y., Lin C., Alani H.* Automatically Extracting Polarity-Bearing Topics for Cross-Domain Sentiment Classification. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 2011, pp. 123–131.
8. *Vasilyev V. G.* Fragment extraction and text classification by logical rules [Klasifikatsija i vydelenie fragmentov v tekstah na osnove logicheskikh pravil] *Digital libraries: Advanced Methods and Technologies, Digital Collections RCDL'2011, Voronezh*, 2011, pp. 133–139.
9. *Marchand M., Shawe-Taylor J.* Learning with the set covering machine. *Proc. 18th International Conf. on Machine Learning*, 2001, pp. 345–352.
10. *Vasilyev V. G.* (2008) Complex technology of automatic text classification [Kompleksnaja tehnologija avtomaticheskoi klassifikacii tekstov]. *Komp'uternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferencii "Dialog 2006"* [Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog 2006"]. *Bekasovo*, 2008, pp. 83–90.