



Лекция 6. Основы интеллектуального анализа текстовых данных



Дисциплина: **Интеллектуальный анализ
данных, текстов и изображений**

Лектор: к.т.н. **Буров Сергей
Александрович**

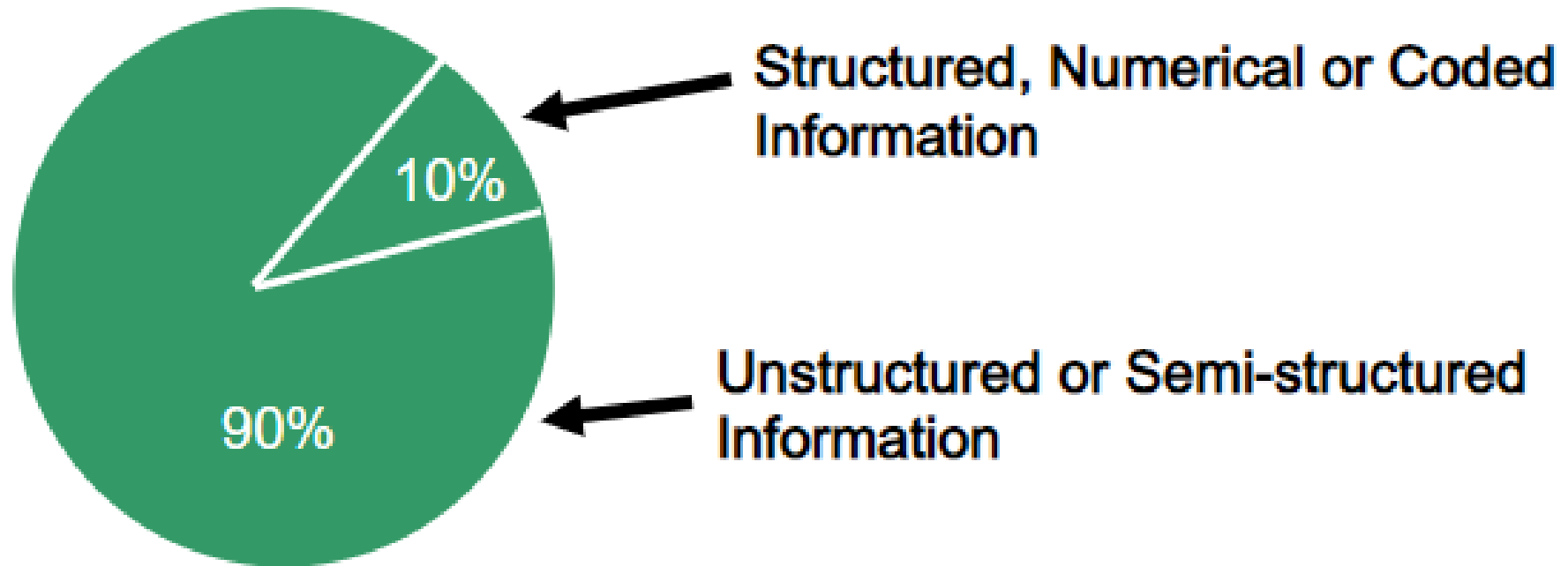
burov-sa@ranepa.ru

Вопросы лекции:

1. Определение и основные задачи интеллектуального анализа текстовых данных (text mining)
2. Содержание основных этапов интеллектуального анализ текстовых данных.
3. Кодирование и векторизация текста. Мешок слов. Метод tf-idf. Использование n-грамм.

Введение

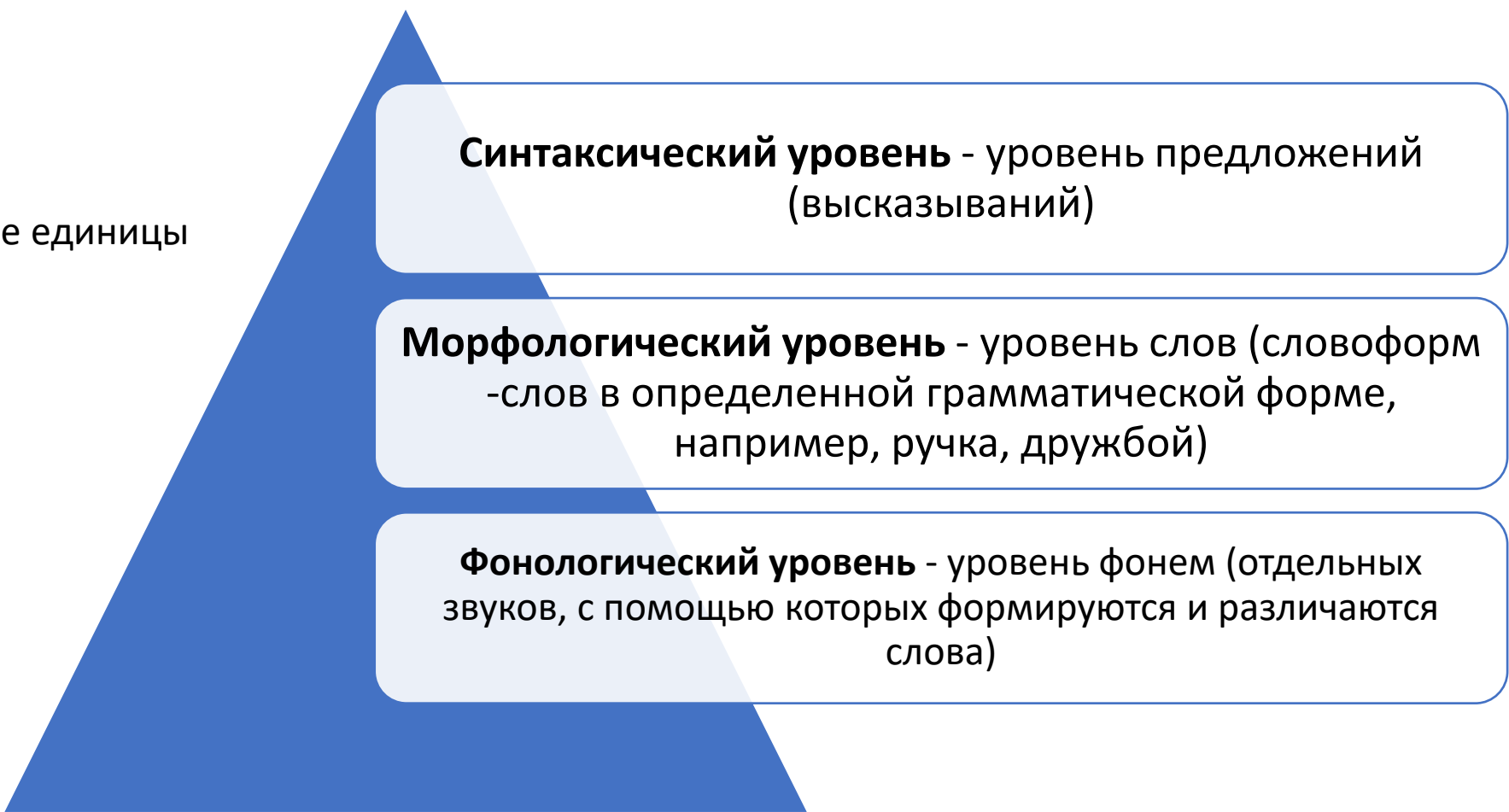
- По данным компании Oracle примерно 90% мировых данных хранятся в неструктурированных форматах
- Информационные бизнес-процессы требуют, чтобы мы перешли от простого поиска документов к обнаружению "знаний"



Введение

Естественный язык и текстовые данные - большая открытая многоуровневая система знаков, возникшая для обмена информацией в процессе практической деятельности человека, и постоянно изменяющаяся в связи с этой деятельностью.

Уровни разбиения текста не единицы



Синтаксический уровень - уровень предложений (высказываний)

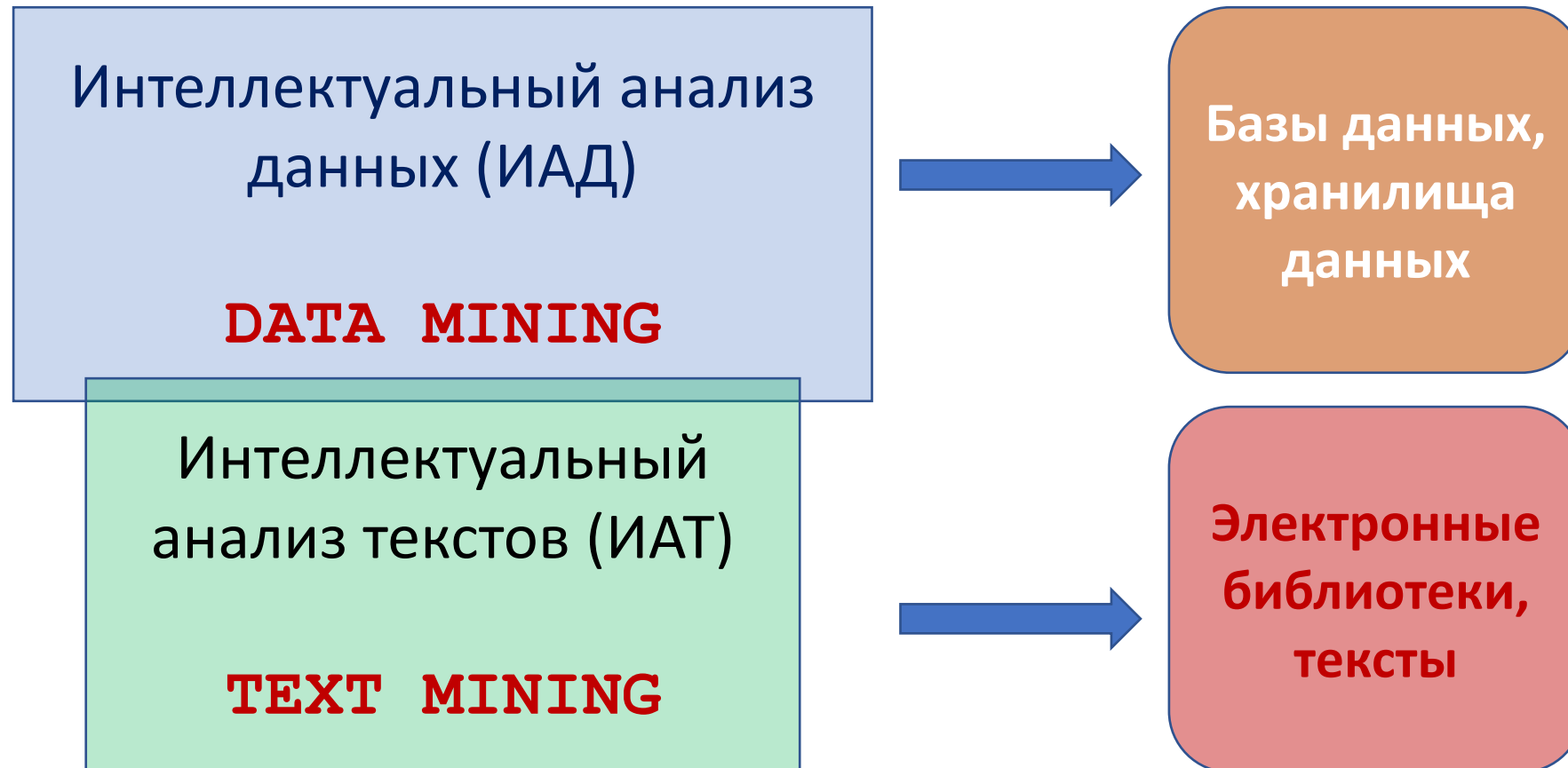
Морфологический уровень - уровень слов (словоформ -слов в определенной грамматической форме, например, ручка, дружбой)

Фонологический уровень - уровень фонем (отдельных звуков, с помощью которых формируются и различаются слова)

Введение

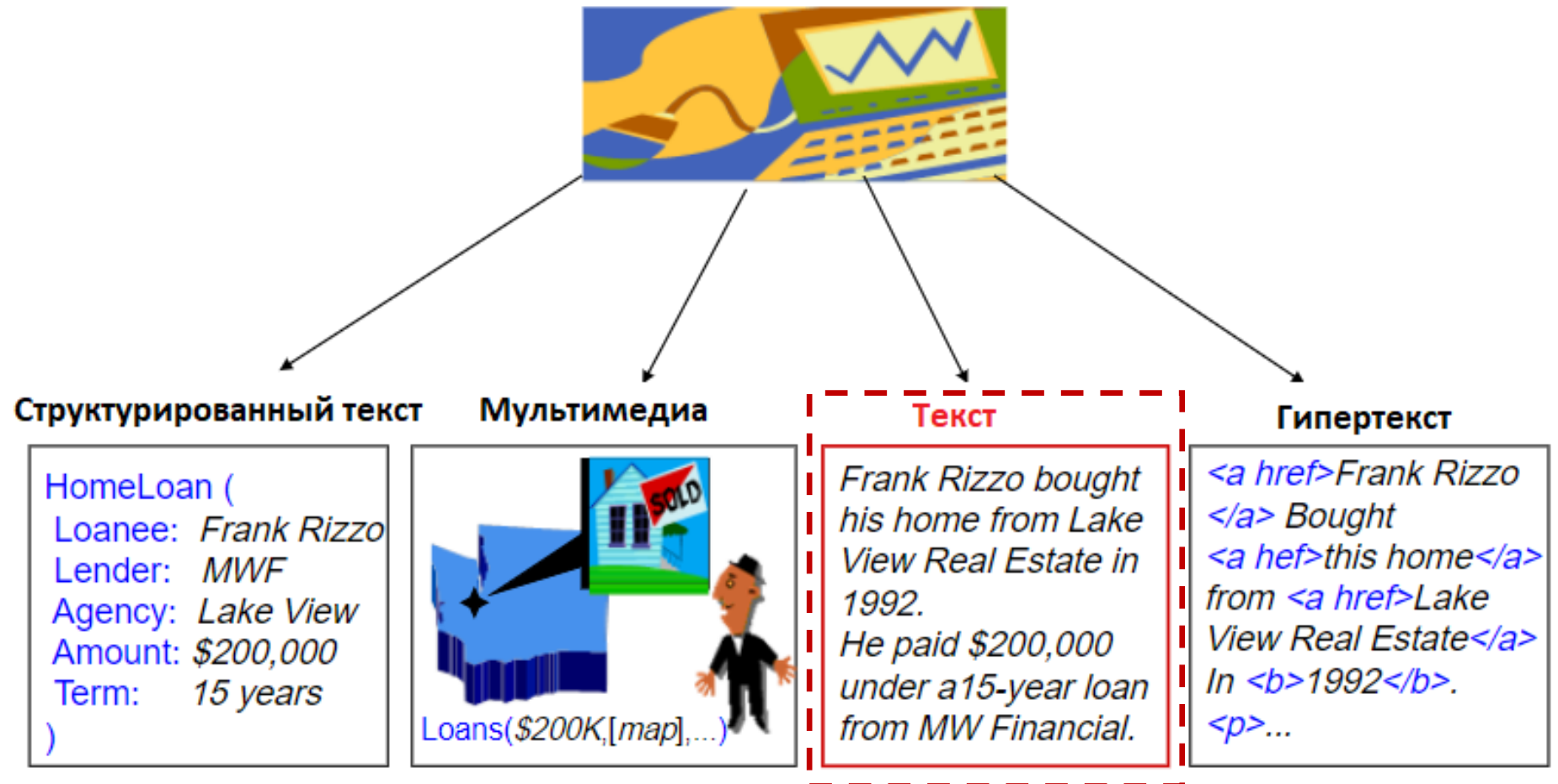
Интеллектуальный анализ текстов имеет схожие цели, подходы к переработке информации и сферы применения с интеллектуальным анализом данных.

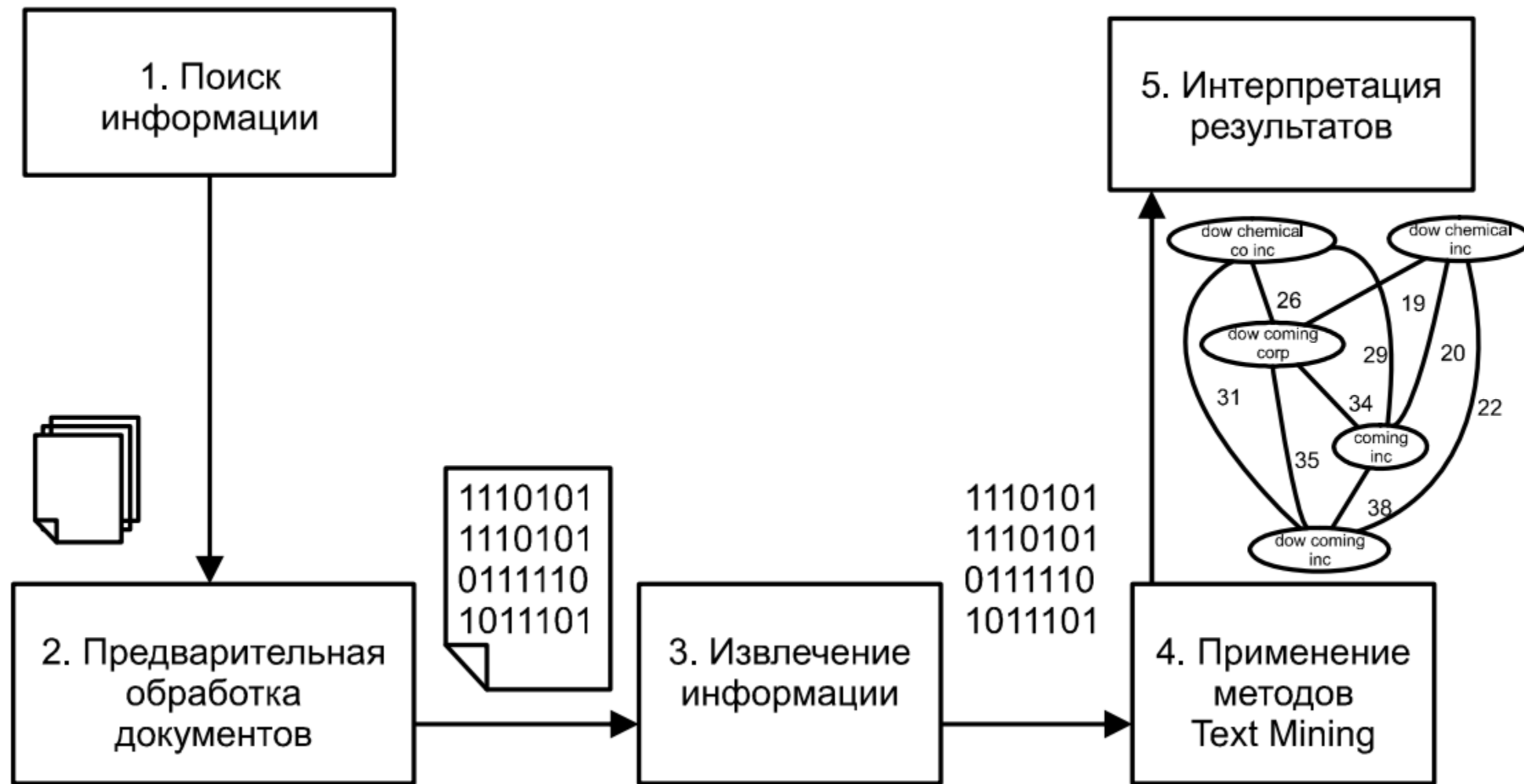
Интеллектуальный анализ текстов отличается специфичностью своих методов и формат обрабатываемых данных.



Язык — это *неструктурированные* данные, которые используются людьми для общения между собой. *Структурированные* или *полуструктурированные* данные, в свою очередь, включают поля или разметку, позволяющие компьютеру анализировать их.

Data Mining / Knowledge Discovery





Определение text mining

Интеллектуальный анализ текстов (ИАТ, англ. text mining) — область интеллектуального анализа данных, являющаяся одним из направлений искусственного интеллекта, целью которого является получение информации из коллекций текстовых документов, основываясь на применении эффективных в практическом плане методов машинного обучения и обработки естественного языка.

Основные задачи text mining

9

Классификация текстов (Categorization)	Определение для каждого документа одной и нескольких заранее заданных категорий, к которой этот документ относится
Рубрицирование (Text Classification)	Отнесение текста к одной из заранее известных тематических рубрик (обычно рубрики образуют иерархическое дерево тематик)
Кластеризация (Text Clustering)	Автоматическое выявление групп семантически схожих документов, среди заданного фиксированного множества
Автоматическое реферирование (Summarization) и аннотирование	Сокращение объёма текста с сохранением его смысла
Извлечение ключевых понятий (feature extraction)	Идентификация фактов и отношений в тексте
Навигация по тексту (Text-base navigation)	Перемещение по документам относительно нужных тем и значимых терминов
Поиск ассоциаций	Идентификация ассоциативных отношений между ключевыми понятиями
Анализ тональности (Sentiment Analysis) и выделение мнений (Opinion Mining)	Поиск мнений пользователей об объектах, анализ общей тональности высказываний и текста в целом

Основные этапы text mining

10



Корпус — это коллекция взаимосвязанных документов (текстов) на естественном языке.

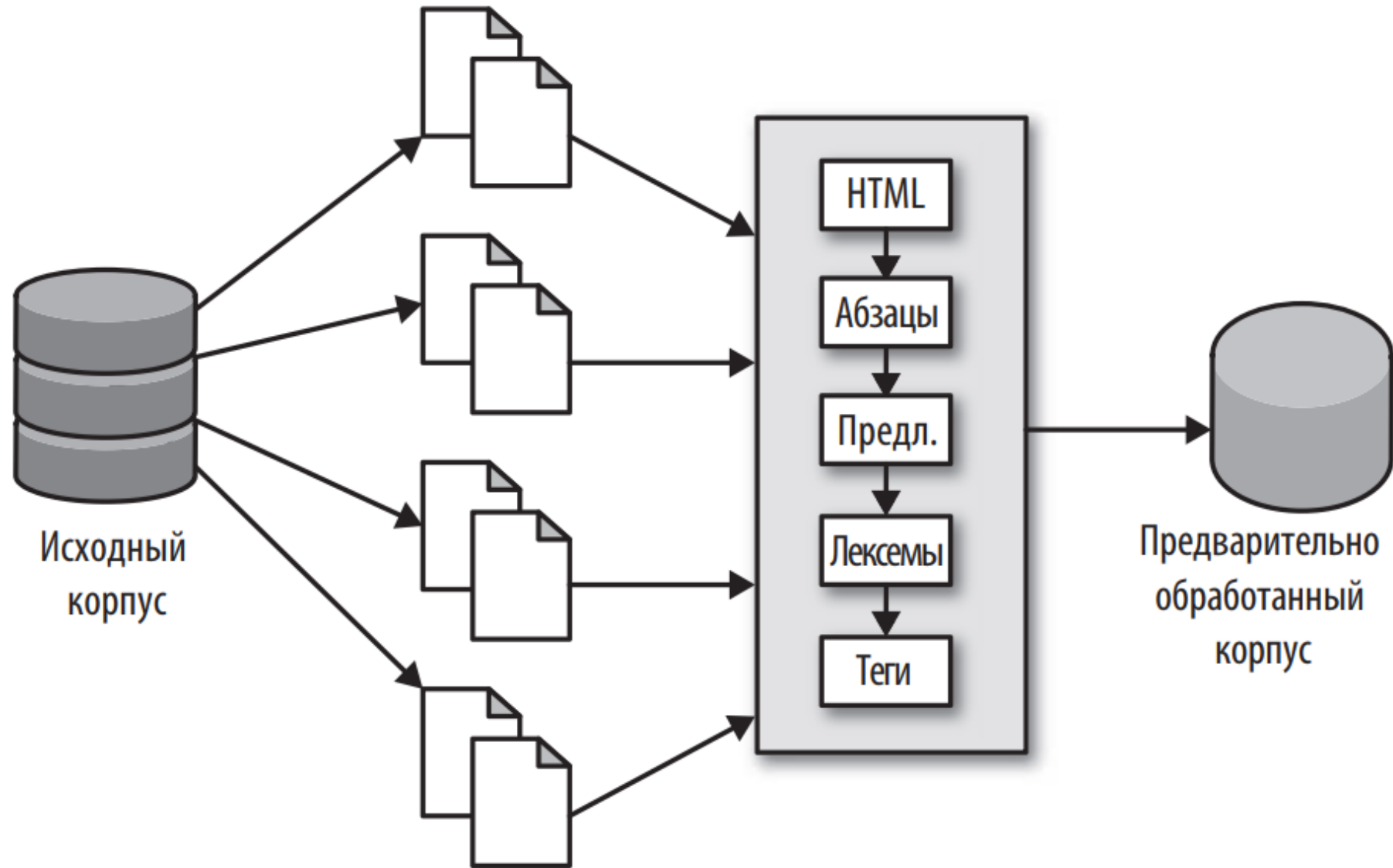
Корпус может быть большим или маленьким, но обычно состоит из десятков и даже сотен гигабайт данных в тысячах документов.

Например, учитывая средний объем почтового ящика в 2 Гбайт (для справки, полная версия корпуса переписки компании Enron, которому сейчас примерно 15 лет, включает 1 миллион электронных писем 118 пользователей и имеет размер¹ 160 Гбайт), компания небольшого размера, насчитывающая 200 сотрудников, способна сгенерировать корпус переписки размером в полтерабайта. Корпусы могут быть *аннотированными*, то есть текст или документы могут быть снабжены специальными метками для алгоритмов обучения с учителем (например, для фильтров спама), или *неаннотированными*, что делает их кандидатами на тематическое моделирование и кластеризацию документов (например, для изучения изменений в темах, скрытых в сообщениях, с течением времени).

Процедура	Вид лингвистического анализа
Токенизация	Графемный анализ
Стемминг Лемматизация Частеречный тэггинг Полный МО	Морфологический анализ
Парсинг	Синтаксический анализ
	Семантический анализ

Вариант подготовки корпуса к предобработке

13



Этап промежуточной обработки и получение трансформированного корпуса

Предобработка текстовых данных

14

NLP, Natural Language Processing – обработка естественного языка.

1.Токенизация – разбиение длинных участков текста на более мелкие (абзацы, предложения, слова).
Токенизация – это самый первый этап обработки текста.

2. Удаление стоп-слов. Стоп-словами называются слова, которые являются вспомогательными и несут мало информации о содержании документа.

Обычно заранее составляются списки таких слов, и в процессе предварительной обработки они удаляются из текста. Типичным примером таких слов являются вспомогательные слова и артикли, например: "так как", "кроме того" и т. п.;

NLP, Natural Language Processing – обработка естественного языка.

3.Нормализация – приведение текста к «рафинированному» виду (единый регистр слов, отсутствие знаков пунктуации, расшифрованные сокращения, словесное написание чисел и т.д.).

Это необходимо для применения унифицированных методов обработки текста. Отметим, что в случае текста термин «нормализация» означает приведение слов к единообразному виду, а **не** [преобразование абсолютных величин к единому диапазону](#).

4.Стеммизация – приведение слова к его корню путем устранения придатков (суффикса, приставки, окончания). Стемминг – морфологический поиск - преобразование каждого слова к его нормальной форме.

Нормальная форма исключает склонение слова, множественную форму, особенности устной речи и т.п. Например, слова "сжатие" и "сжатый" должны быть преобразованы в нормальную форму слова "сжимать". Алгоритмы морфологического разбора учитывают языковые особенности и вследствие этого являются языкозависимыми алгоритмами

NLP, Natural Language Processing – обработка естественного языка.

5. N-граммы — это альтернатива морфологическому разбору и удалению стоп-слов. N-грамма — это часть строки, состоящая из N символов.

Например, слово "дата" может быть представлено 3-граммой "_да", "дат", "ата", "та_" или 4-граммой "_дат", "дата", "ата_", где символ подчеркивания заменяет предшествующий или замыкающий слово пробел. По сравнению со стеммингом или удалением стоп-слов, N-граммы менее чувствительны к грамматическим и типографическим ошибкам. Кроме того, N-граммы не требуют лингвистического представления слов, что делает данный прием более независимым от языка. Однако N-граммы, позволяя сделать текст более строгим, не решают проблему уменьшения количества неинформативных слов.

NLP, Natural Language Processing – обработка естественного языка.

6. Лемматизация – приведение слова к смысловой канонической форме слова (инфинитив для глагола, именительный падеж единственного числа – для существительных и прилагательных). Например, «зарезервированный» – «резервировать», «грибами» – «гриб», «лучший» – «хороший».

7. Чистка – удаление стоп-слов, которые не несут смысловой нагрузки (артикли, междометья, союзы, предлоги и т.д.).

Токенизация = автоматический графемный анализ

Процедура выделения в тексте слов, чисел, а также нахождение границ устойчивых сочетаний и предложений.

Выделяемые текстовые единицы – **токены**

англ. *tokenization, token*

1. Разделение входного текста на элементы (слова, разделители и т.д.);
2. удаление нетекстовых элементов;
3. выделение и оформление нестандартных (нелексических) элементов, например:
 - элементов форматирования;
 - структурных элементов текста;
 - различных элементов текста, не являющихся словами;
 - имен (имя, отчество), написанных инициалами;
 - иностранных лексем, записанных латиницей и т.д.

- обработка дефиса и пробела;
- выделение составных предлогов, устойчивых оборотов, аналитических форм и др.;
- иноязычные фрагменты;
- нетекстовые элементы.

1. Межсловный дефис:

- объединительная функция (буква)?
кто-то, где-нибудь, давным-давно, бакш-таг, брейд-вымпел, генерал-аншеф

или

- разделительная функция (знак препинания)?
старик-художник, словарь-справочник, девочка-пионерка

2. Пробел:

- объединительная функция (буква)?
сто двадцать пять

или

- разделительная функция?
русский язык

Элементы текста, требующих специальной обработки

- Названия рисунков
- Сами рисунки
- Примечания
- Страницы форзаца
- Зачеркивания
- Титульные листы
- Списки литературы
- Цифры
- Иностр. язык в тексте
- Адреса, ссылки, гиперссылки
- Сокращения, аббревиатуры
- Пример поиска
- Адрес докладчика/университетата
- Тезисы докладов отдельным файлом
- Перечисления в тексте
- Текст списком
- Слова типа «рис1», «р2», Нкластеры
- Таблицы
- Формат
- Римские цифры
- Рус. яз. в иностранном тексте
- Формулы
- Значки для формул
- Схемы

- **преобразование текста**, при котором каждая словоформа текста представлена в виде пары <лемма + морфологическая характеристика>, где
- **Лемма** – это основная форма слова,
- **Морфологическая характеристика** указывает часть речи, падеж, род, число и т.д. соответствующей словоформы.

- **Лемматизация**, т.е. сведение различных словоформ к исходной форме, или **лемме**
- **Стемминг** – приведение разных словоформ к одной основе
- **Частеречный тэгинг (pos-tagging)**, т.е. указание части речи для каждой словоформы в тексте
- **Полный морфологический анализ** - приписывание грамматических характеристик (граммем) словоформе

- 1. словарный**, при котором задаётся словарь словоформ или словарь основ и окончаний. Такие системы, как правило, базируются на *Грамматическом словаре* А.А. Зализняка;
- 2. бессловарный**, при котором задаётся список возможных окончаний (или псевдоокончаний) с приписанной им информацией о возможных грамматических значениях, а также используются вероятностно-статистические методы.

Словарный подход к морфологическому анализу

Особенности

- Наиболее **лингвистический** метод
- Дает максимально **полный** анализ словоформы
- Этот подход реализован, например, в системах машинного перевода **ЭТАП** (разрабатывавшиеся под руководством Ю.Д. Апресяна и основанные на модели «СМЫСЛ ↔ ТЕКСТ»), **ПРОМТ**.

Минусы:

- Проблема **большого объема словаря**, который создается **вручную**
- Проблема **анализа новых** слов (для данной системы, то есть относительно используемого словаря) Не существует абсолютно полных словарей – лексика языка непрерывно пополняется
- Невозможно включить в словарь **всю** существующую терминологию, имена, фамилии и т.д.

Основа – принятие гипотезы о грамматических характеристиках на основе аналогий

ВАРИАНТ 1

спряжение по образцу слова ПИРОВАТЬ

* значение грамматического признака (ГП) «вид» неизвестно *
(выбран несовершенный вид)

КРОВАТЬ

КРУЙ КРУЙТЕ

КРУЮ (БУДУ КРОВАТЬ)

КРУЕШЬ (БУДЕШЬ КРОВАТЬ)

КРУЕТ (БУДЕТ КРОВАТЬ)

КРУЕМ (БУДЕМ КРОВАТЬ)

КРУЕТЕ (БУДЕТЕ КРОВАТЬ)

КРУЮТ (БУДУТ КРОВАТЬ)

КРОВАЛ КРОВАЛА КРОВАЛО КРОВАЛИ

КРУЯ КРОВАВ

ВАРИАНТ 2

склонение по образцу слова ПЕЧАТЬ

* значение ГП «одушевленность» неизвестно *

КРОВАТЬ КРОВАТИ

КРОВАТИ КРОВАТЕЙ

КРОВАТИ КРОВАТЯМ

КРОВАТЬ КРОВАТЕЙ / КРОВАТИ

КРОВАТЬЮ КРОВАТЯМИ

КРОВАТИ КРОВАТЯХ

ВАРИАНТ 3

КРОВАТЬ

как неизменяемое слово (по аналогии с ДЕСКАТЬ)

Бессловарный подход к морфологическому анализу (**достоинства**)

- + **Более экономичный**, т.к. не нужен словарь основ или словоформ
- + Позволяет одинаковым способом **обрабатывать все слова как «новые»**, не найденные в словаре.
- Для этого задаются **списки грамматических морфем** языка: флексий, предлогов, союзов, частиц

Бессловарный подход к морфологическому анализу (**недостатки**)

- 👉 Не имеет выхода к семантическому анализу, для которого нужно знать леммы.
- 👉 Все слова трактуются как новые для анализатора
→ большее количество ошибочных решений
- Для снижения их численности используются **элементы синтаксического анализа** (учитываются возможные списки сочетаний грамматических морфем)

Например, *На –ом –е* соответствует определенному типу синтаксических структур, в частности, словосочетанию *На золотом крыльце*).

Сложности при морфологическом анализе

Омонимы – слова одинаковые по произношению и написанию, но разные по лексическому значению

Виды омонимов:

- Омоформы** — совпадающие в определенных грамматических формах.
Русская *печь* — *печь* пирожки; стая *голубей* — небо стало *голубей*
- Омофоны** — совпадают только фонетически, но не графически. *Посидеть* на лавочке — *посесть* от старости; *поласкать* щенка — *полоскать* белье
- Омографы** — написание одинаковое, но произносятся по-разному.
Старинный *замок* — починить дверной *замок*

На завод привезли **стекло**.

Масло **стекло** на пол.

Данные эксперименты являются ошибочными.

Последние **данные** являются ошибочными.

Эти типы стали есть в цехе



Омнонимы в английском языке

32

air — heir воздух/проветрить — наследник	<ul style="list-style-type: none">• fresh air — свежий воздух• the heir to the throne — наследник престола
band — band отряд/группа — завязка	<ul style="list-style-type: none">• a rock band — рок-группа• a rubber band — резинка для волос
bank — bank насыпь/берег — банк	<ul style="list-style-type: none">• the bank of the river — берег реки• the bank of England — банк Англии
bare — bear — bear голый — нести/родить — медведь	<ul style="list-style-type: none">• with bare hands — голыми руками• bear in mind — иметь в виду• a polar bear — полярный медведь

Извлечение ключевых понятий из текста

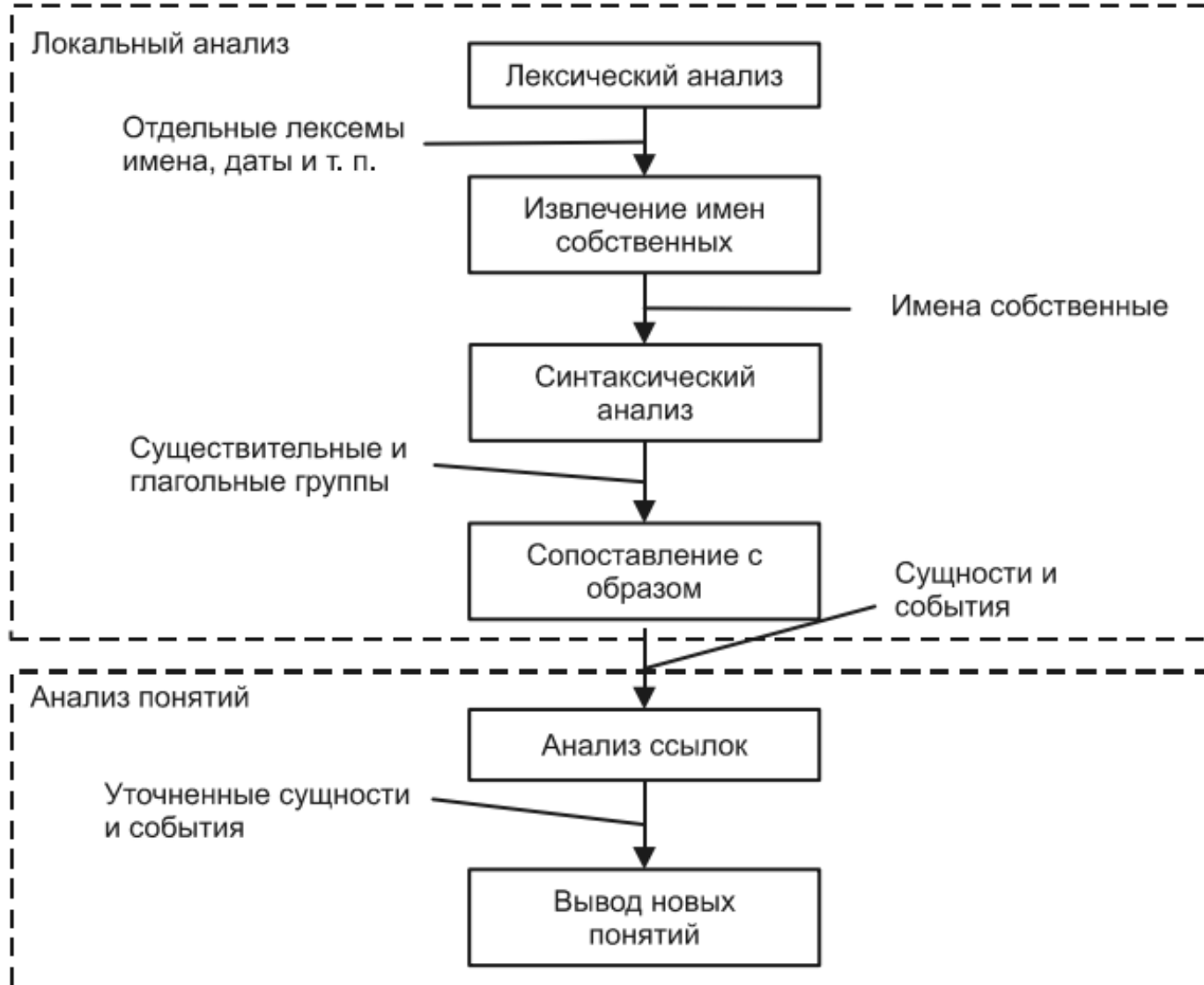
(feature extraction)

Извлечение ключевых понятий из текста может рассматриваться и как отдельный этап анализа текста, и как определенная прикладная задача. В первом случае извлеченные из текста факты используются для решения различных задач анализа: классификации, кластеризации и др.

Большинство методов Data Mining, адаптированные для анализа текстов, работают именно с такими отдельными понятиями, рассматривая их в качестве атрибутов данных.

Извлечение ключевых понятий из текста

(feature extraction)



Извлечение фактов выполняется при помощи сопоставления:

текст ↔ **образцы**

выражение сопоставляется с текстовыми сегментами, то такие сегменты помечаются метками. При необходимости этим сегментам приписываются дополнительные свойства. Образцы организуются в наборы. Метки, ассоциированные с одним набором, могут ссылаться на другие наборы.

Каждый образец имеет связанный с ним набор действий (например, пометить текстовый сегмент новой меткой), но могут быть и другие действия.

Основной целью сопоставления с образцами является выделение в тексте сущностей, связей и событий. Все они могут быть преобразованы в некоторые структуры, которые могут анализироваться стандартными методами Data Mining.

**Как представить слова, чтобы они были
понятны машине?**

Мешок слов (**bag-of-words**)

36



Мешок слов (bag-of-words)

Матрица частот совместного появления слов

Блок 1
The fast cat
wears no hat.

Блок 2
The cat in
the hat ran
fast.

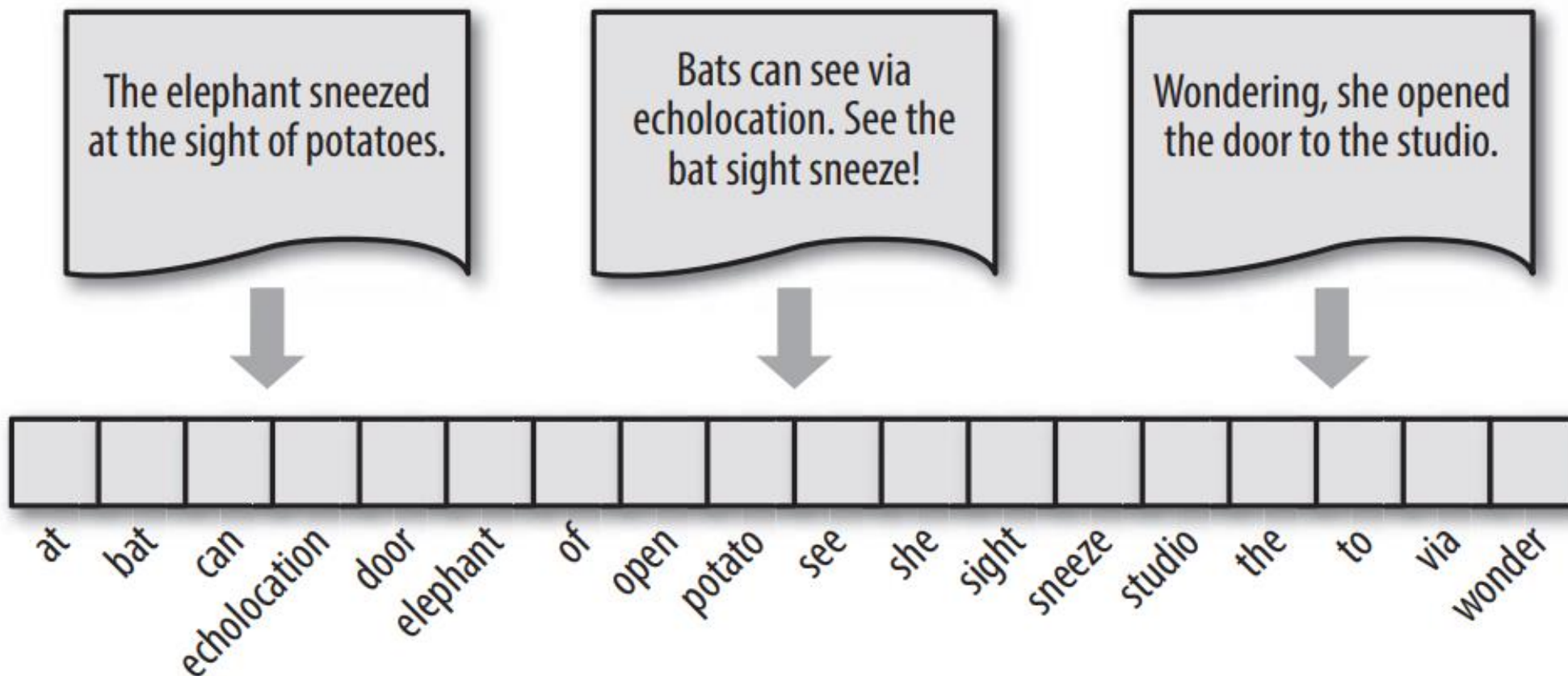


cat
fast
hat
in
no
ran
the
wears

	cat	fast	hat	in	no	ran	the	wears
0								
2	0							
2	2	0						
1	1	1	1	0				
1	1	1	1	0	0			
1	1	1	1	1	0	0		
3	3	3	3	2	1	2	1	
1	1	1	1	0	1	0	1	0

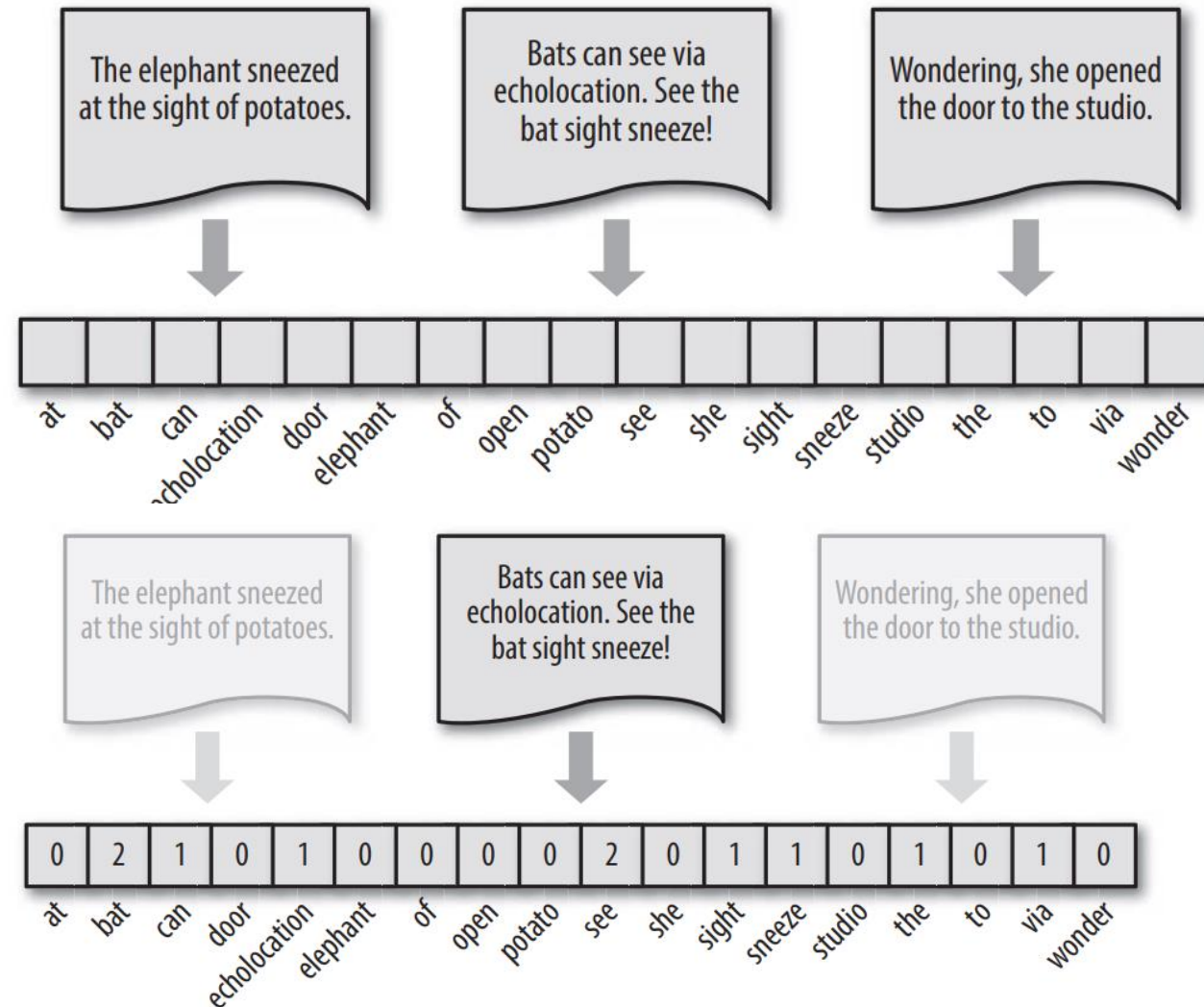
Мешок слов (**bag-of-words**)

Представление документов в виде векторов



Мешок слов (bag-of-words)

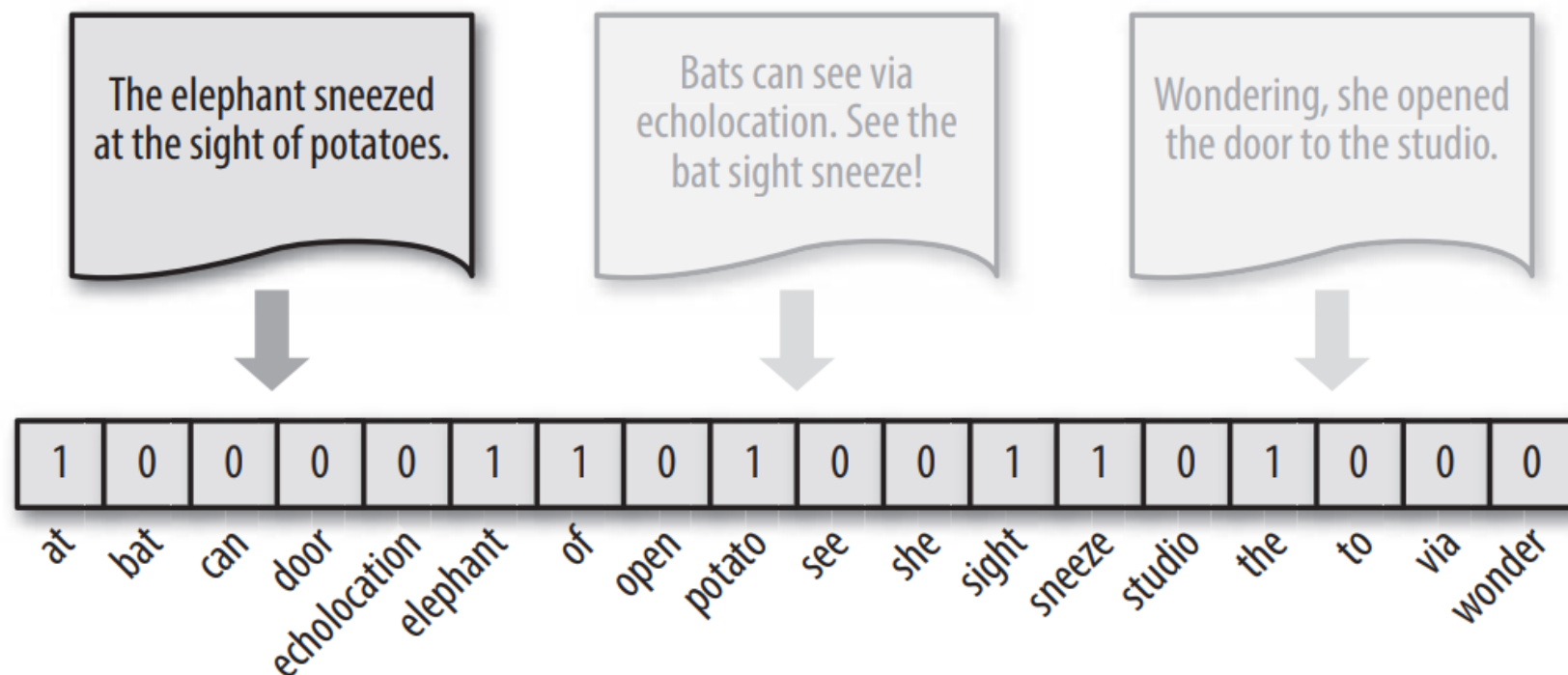
Представление документов в виде векторов



Прямое кодирование

Игнорируя грамматику и относительные позиции слов в документах, частотные методы кодирования страдают проблемой *вытянутого хвоста*, или распределения Ципфа (Zipfian distribution), характерной для естественных языков. В результате лексемы, встречающиеся очень часто, оказываются на порядки более «значимыми», чем другие, встречающиеся намного реже. Это может оказывать существенное влияние на некоторые модели (например, обобщенные линейные модели), которые предполагают нормальное распределение признаков. Решением этой проблемы является

Прямое кодирование – метод логической векторизации, который помещает в соответствующий элемент вектора значение **true (1)**, если лексема присутствует в документе, и **false (0)** — если отсутствует.



1 – лексема присутствует
0 – лексема отсутствует

Кодирование вектора слов для описания семантического пространства



Как удалить несущественные признаки?
Как учитывать контекст?

Масштабирование данных. Метод **tf-idf**

Вместо исключения несущественных признаков можно масштабировать признаки в зависимости от степени их информативности

Метод **tf-idf** – метод частота термина-обратная частота документа (**term frequency-inverse document frequency**)

Основная идея – присвоить больший вес термину, который часто встречается в конкретном документе, но при этом редко встречается в остальных документах.

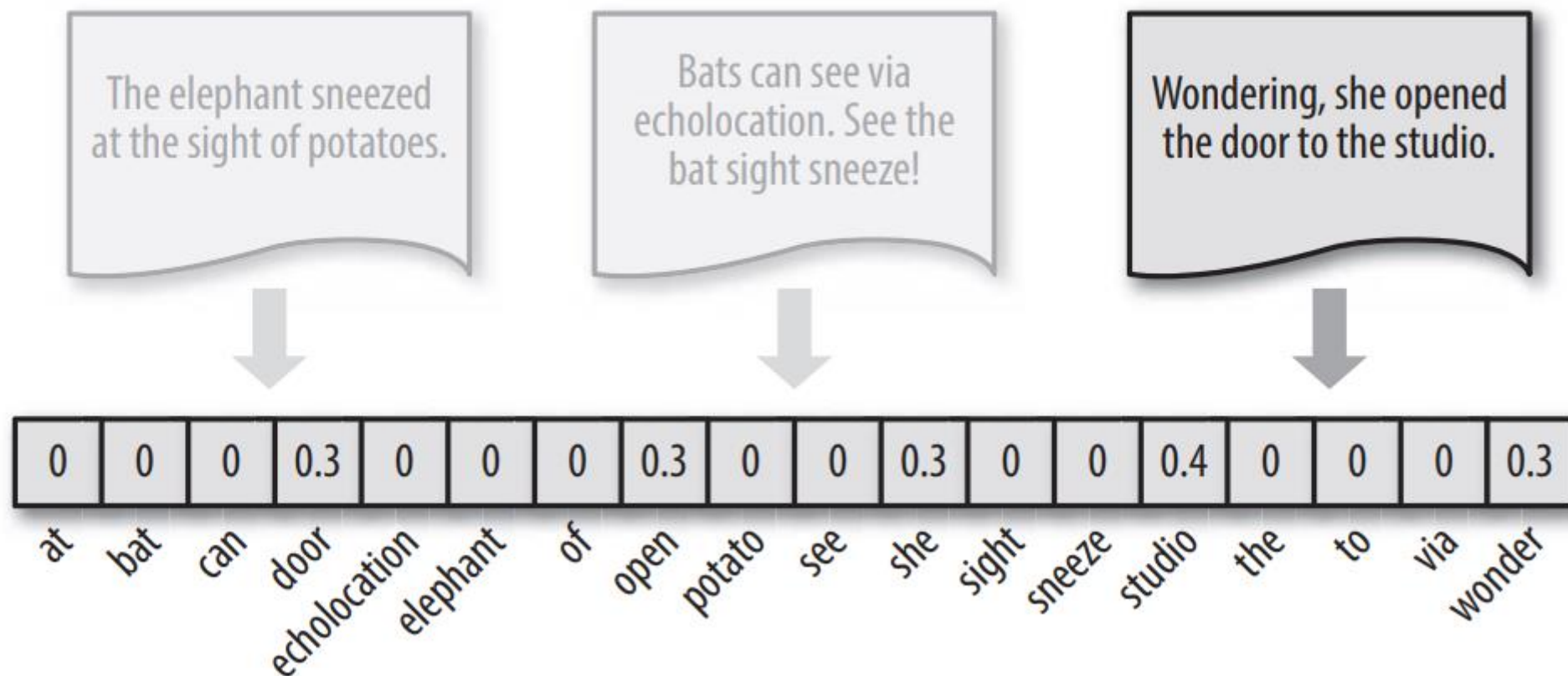
Основной смысл документа закодирован в более редких словах.

Масштабирование данных. Метод **tf-idf**

$$\text{tfidf}(w, d) = \text{tf} \log\left(\frac{N+1}{N_w+1}\right) + 1$$

где N – это количество документов в обучающем наборе, N_w – это количество документов обучающего набора, в которых встретилось слово w , и tf (частота термина) – это частота встречаемости термина в запрашиваемом документе d (документе, который вы хотите преобразовать). Кроме того, оба класса применяют L2 нормализацию после того, как вычисляют представление tf-idf. Другими словами, они масштабируют векторизованное представление каждого документа к единичной евклидовой норме (длине). Подобное масштабирование означает, что длина документа (количество слов) не меняет его векторизованное представление.

Масштабирование данных. Метод **tf-idf**



Недостаток метода **tf-idf** -> **игнорирование** **порядка слов**

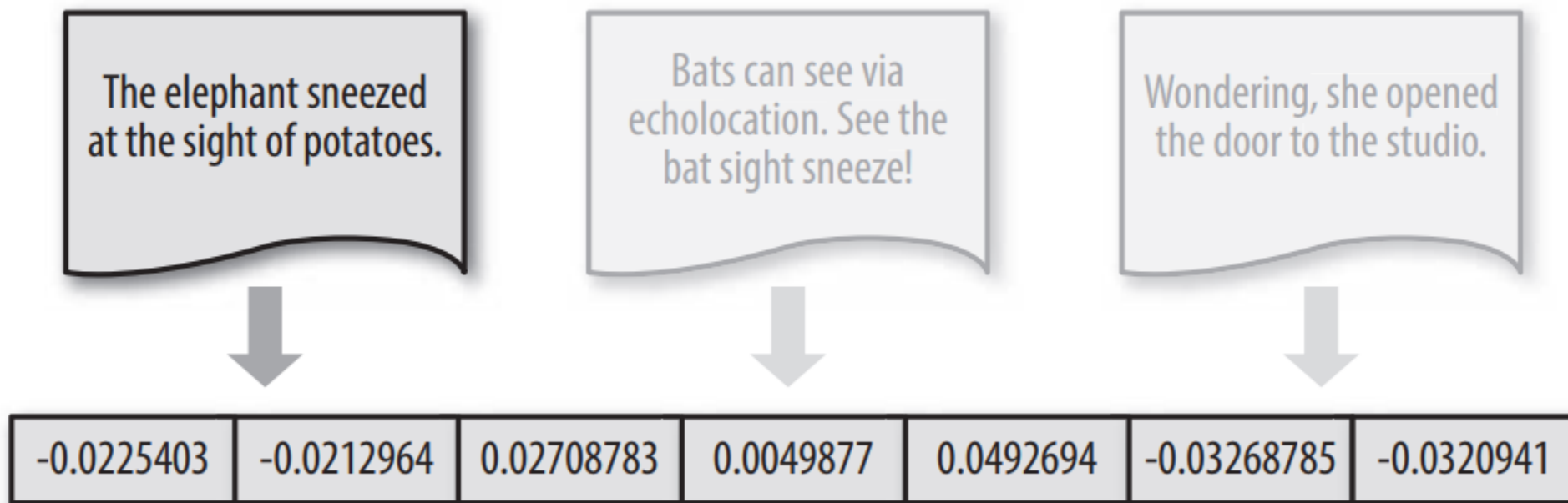
«it's bad, not good at all» и «it's good, not bad at all»

«это плохо, это всё нехорошо» и «это хорошо, это всё неплохо»

Контекст важен!

Распределённое представление

Если в контексте приложения сходство между документами играет важную роль, текст можно закодировать в виде числовой последовательности методом распределенного представления. При таком подходе вектор документа является не простым отображением позиций лексем в их числовые значения, а набором признаков, определяющих сходство слов. Сложность пространства признаков (и длина вектора) определяется особенностями обучения этого представления и напрямую не связана с самим документом.



Сравнение методов векторизации текста

48

Метод векторизации	Принцип действия	Хорошо подходит для	Недостатки
Частотный	Подсчет частоты вхождения лексем	Байесовские модели	Самые часто встречающиеся слова не всегда являются самыми информативными
Прямое кодирование	Определение логического признака присутствия лексемы (0, 1)	Нейронные сети	Все слова оказываются равноудаленными, поэтому очень важна нормализация
TF-IDF	Нормализация частоты лексем по документам	Приложения общего назначения	Умеренно часто встречающиеся слова могут быть не репрезентативными для темы документа
Распределенные представления	Кодирование сходства лексем на основе контекста	Моделирование сложных отношений	Большой объем вычислений, сложность масштабирования без применения дополнительных инструментов (например, Tensorflow)

Использование n-грамм

n-грамма – несколько рядом стоящих единиц текста (токенов)

Биграммы (bigrams) – пары токенов

Триграммы (trigrams) – тройки токенов

Естественный язык и обработка текста – это крупная научная область, и обсуждение деталей передовых методов выходит за рамки не только данной лекции, но и дисциплины.

Рекомендуемая литература:

книга издательства O'Reilly [Natural Language Processing with Python](#), написанную Стивеном Бердом, Эваном Кляйном и Эдвардом Лопером, в которой дан обзор NLP, а также рассказывается о питоновском пакете `nltk` для NLP.

Больше теории – справочное издание [Introduction to Information Retrieval](#), написанная Кристофером Меннингом, Прабхакаром Рагхаваном и Генрихом Шютце и посвященная основным алгоритмам информационного поиска, NLP и машинного обучения