

# Основы обработки текстов

лекция 1

# О курсе

- Лекторы: **Турдаков Денис Юрьевич**
  - Майоров Владимир Дмитриевич
  - Перминов Андрей Игоревич
  - Недумов Ярослав Ростиславович
- Лекции каждую **среду в 10.30 ауд. П-8а**
  - предполагаются минимальные знания
    - линейной алгебры,
    - теории вероятности и математической статистики
    - программирования
  - не все имеют одинаковые знания
    - предполагается, что студенты могут быстро учиться

# План на сегодня

- Подробнее о курсе и практикуме
- Язык программирования Python
- Проблемы обработки текстов

## Часть 1

# О курсе

- Курс состоит из
  - лекций,
  - практикума и
  - итогового экзамена
- Язык программирования Python 3
- Вся информация: <http://tpc.at.ispras.ru>

# Практическая часть

- Практические задачи обработки текстов
  - В этом году:
    - Вводная задача на регулярные выражения
      - Пароли, e-mail, URL, даты
    - Распознавание именованных сущностей
      - Имена людей, названия мест и организаций, даты
    - Нормализация (приведение к стандартной форме)
      - Имена людей, даты
- Веб-интерфейс для проверки и первое задание будут доступны на следующей неделе

## Часть 2

# Python

«Читаемость имеет значение»

PEP 20 - The Zen of Python

- Код читается гораздо чаще, чем пишется
  - PEP 8 -- Style Guide for Python Code
  - Используйте средства автоматической проверки на соответствие PEP 8 (Pylint, PyFlakes, ...) или комбинированные средства (Pylama, Flake8)
- Комментарии помогают
  - PEP 257 -- Docstring Conventions
  - Автоматическая генерация документации (Sphinx)
  - Автогенерация тестов (Doctest)



# Python

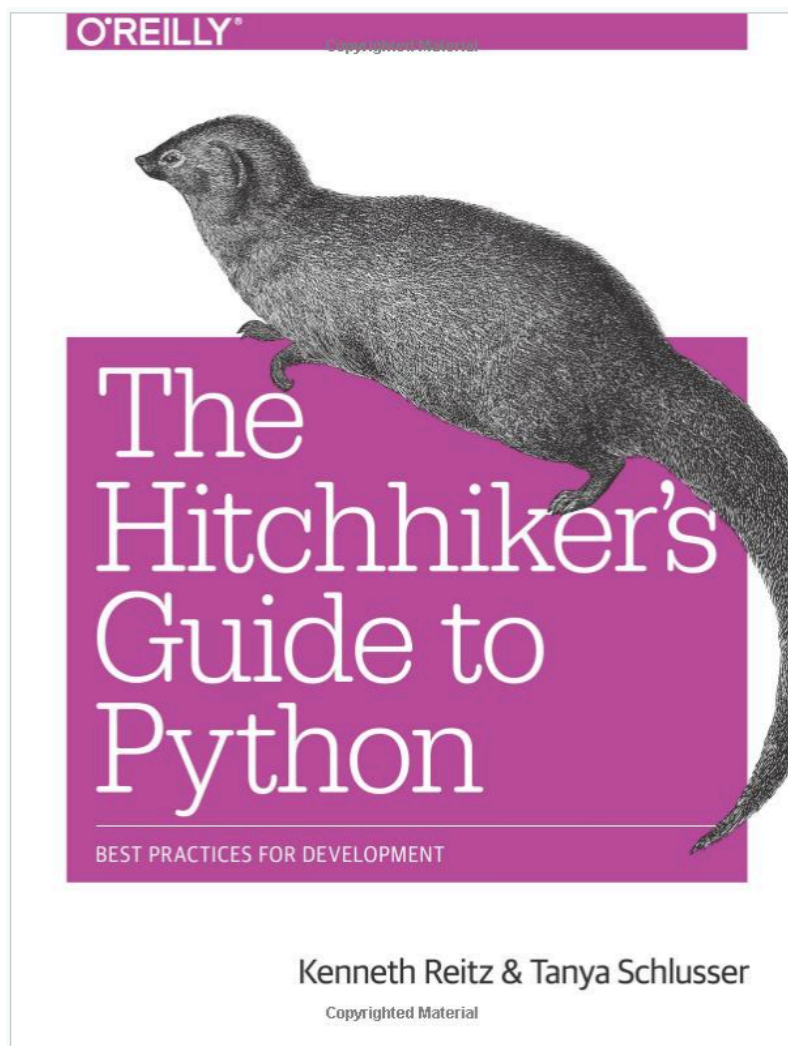
- Значимые пробелы

```
if x==1:  
    print ('x is 1')  
    print ('внутри блока')  
print ('вне блока')
```

- Используйте IDE: PyCharm, Jupyter и т.д.

# Python

- Правильная установка и использование
  - Прочитайте про `pip` и `virtualenv`
- Прочитайте о возможностях языка, и попробуйте их использовать в своих проектах
  - Но прежде чем отсылать решение подумайте, действительно ли стало проще читать ваш код?



<https://docs.python-guide.org>

# Регулярные выражения

- Обновить цену товара в прайс-листе:  
для конкретного товара за 1000р. сделать 999.99р.
- Заменить все вхождения одного слова в тексте на другое
  - для части слова (Википедия -> Энциклопедия)
  - с учетом контекста
- Найти сообщения о терроризме
- Фильтрация нецензурных высказываний на форумах



# Регулярные выражения

The screenshot displays the regex101.com website interface. At the top, the header includes the site name "regular expressions 101" and navigation links for social media, donations, and contact. The main workspace is divided into three sections: "REGULAR EXPRESSION" with a placeholder "insert your regular expression here", "TEST STRING" with a placeholder "insert your test string here", and a "no match" status indicator. On the left sidebar, there are panels for "SAVE & SHARE", "FLAVOR" (listing PCRE2, PCRE, ECMAScript, Python, Golang, and Java 8), "FUNCTION" (listing Match, Substitution, List, and Unit Tests), and "TOOLS" (listing Code Generator and Regex Debugger). On the right sidebar, there are panels for "EXPLANATION", "MATCH INFORMATION", and "QUICK REFERENCE" (which includes a search bar and a list of common tokens like [abc], [^abc], [a-z], etc.). The bottom of the page features a "SPONSOR" section for Layer0 and Jamstack at Scale.

# Регулярные выражения

- Регулярные выражения - алгебраическая нотация для записи множества строк
- Функции Python

```
import re  
re.search("в", "пиво").group(0) # в  
re.sub("о", "ко", "пиво") # пивко  
re.findall("cd", "abcdcde") # ["cd", "cd"]
```

# Регулярные выражения

- Последовательность букв: *abcd*
- Чувствительны к регистру: “Пиво” “пиво”
- Дизъюнкция: *[П|п]иво, [abc], [1234567890]*
- Интервал: *[A-Z], [0-9], [A-Za-z]*

```
for letter in re.findall("[a-o]", "пиво"):
    print(letter, end=' ')
> и в о
```

- Знак  $\wedge$ : *[^a]* - все кроме “a”
- “.” - любой символ, кроме  $\backslash n$

# Регулярные выражения

- ? - условие для 0 или 1 вхождения символа

```
re.findall("пивк?о", "пиво или пивко")  
> ["пиво", "пивко"]
```

- Как найти "Goooooogle"?
- Счетчики
  - Gooo\*gle
  - Goo+gle

```
print(re.sub("Goo+gle", "Google", "Goooooogle"))  
> Google
```

# Регулярные выражения

- Якоря

^ - начало строки

\$ - конец строки

```
re.search("^cat(1|2)","cat1 cat2").group(0)
> cat1
re.search("cat(1|2)$","cat1 cat2").group(0)
> cat2
```



# Регулярные выражения

- Память (Memory)

```
text="A conditional random field (CRF) is a ..."  
print(re.search("\([^)]+\)",text).group(0))  
> (CRF)  
print(re.search("\(([^)]+)\)",text).group(1))  
> CRF
```

```
text1 = "the faster they ran, the faster we ran"  
text2 = "the faster they ran, the faster we ate"  
re.search("the (.*?)er they (.*), the \\1er we \\2", text1) # Match  
re.search("the (.*?)er they (.*), the \\1er we \\2", text2) # Not match
```

- Приоритет операций

Круглые скобки	()
Счетчики	* + ? { }
Последовательности и якоря	the ^my end\$
Дизъюнкция	

# Python и машинное обучение

- scikit-learn
- <http://scikit-learn.org>

```
from sklearn.naive_bayes import GaussianNB
x = [[0,0],[1,1]]
y = [0,1]

classifier = GaussianNB()
trained_classifier = classifier.fit(x,y)
predicted_value = trained_classifier.predict([0.6,0.6])

> [1]
```

- Keras, PyTorch, TensorFlow

См. спецкурс:  
<https://mlcourse.at.ispras.ru/>

## Часть 3

# Классические задачи обработки текстов

- Информационный поиск (IR)
- Извлечение информации (IE)
- Вопросно-ответные системы (QA)
- Классификация и кластеризация
- Автоматическое аннотирование и реферирование
- Диалоговые системы
- Машинный перевод

# Приложения обработки текстов

# Уровни обработки текстов

- Морфологический
  - I'm - I am
  - кошка-кошки, дно-?
- Синтаксический
  - Мне один черный кофе и один сладкий булка...
- Семантический
  - Сколько китайского шелка было экспортировано в Западную Европу в конце 18 века?
  - лексическая и композиционная семантика
- Прагматический (дискурс)
  - Сколько тогда было штатов в США?
  - установление кореферентности (coreference resolution)

# Многозначность

- Ключевая проблема обработки текстов
- Я траву **косил косой**,  
Дождик вдруг пошел **косой**.  
Бросил я тогда **косить**  
И на Стешу стал **косить**.  
Ну а Стеша, ох, краса,  
Как огонь её **коса**!

# Многозначность

- Морфологическая

- часть речи

- мой (-- нос, -- руки)

- look ( look at me, have a look)

Алгоритмы определения  
частей речи (part of  
speech tagging)

- Синтаксическая

- мужу изменять нельзя

- мать любит дочь

- Flying planes can be dangerous

Синтаксический  
разбор (parsing)



# Многозначность

- Лексическая (семантическая)

- Омонимия (ключ)

- полисемия (платформа)

- семантическая многозначность (лиса)

разрешение  
лексической  
многозначности (word  
sense disambiguation)

- Прагматическая

- Огонь! (в армии или в комнате)

- You have a green light

# Многозначность и перевод

- Help для Windows 95

... Мышь может неадекватно реагировать на щелчок по почкам. Но не спешите! Это могут быть физические проблемы, а не клоп Окон 95.

Почистите вашу мышь.

Отсоедините ее поводок от компьютера, вытащите гениталий и промойте его и ролики внутренностей спиртом. Снова зашейте мышь.

Проверьте на переломы поводка.

Подсоедините мышь к компьютеру.

Приглядитесь к вашей прокладке (подушке) - она не должна быть источником мусора и пыли в гениталии и роликах.

Поверхность прокладки не должна стеснять движения мыши.

...

# Сложность языка

- Естественный язык:
  - многозначен на всех уровнях
  - сложное, едва уловимое использование контекста для передачи значения
  - включает знания и рассуждения о мире
- Но обработка естественного языка может быть иногда очень простой
  - использование грубых признаков часто позволяют достичь очень хороших результатов

# Понимание

- Тест Тьюринга
- Китайская комната
- ELIZA (1966)
  - „I am unhappy.“
  - „How long have you been unhappy?“
  - „Six months. Can you help me?“
  - „What makes you think I help you?“
  - „You remind me of my sister.“
  - „Can you tell me more about your sister?“
  - „I like teasing my sister.“
  - „Does anyone else in your family like teasing your sister?“
  - „No. Only me.“
  - „Please go on.“

# ELIZA

User: Men are all alike.

ELIZA: IN WHAT WAY?

User: They're always bugging us about something other

ELIZA: CAN YOU THINK OF A SPECIFIC EXAMPLE?

User: My boyfriend made me come here

ELIZA: YOUR BOYFRIEND MADE YOU COME HERE

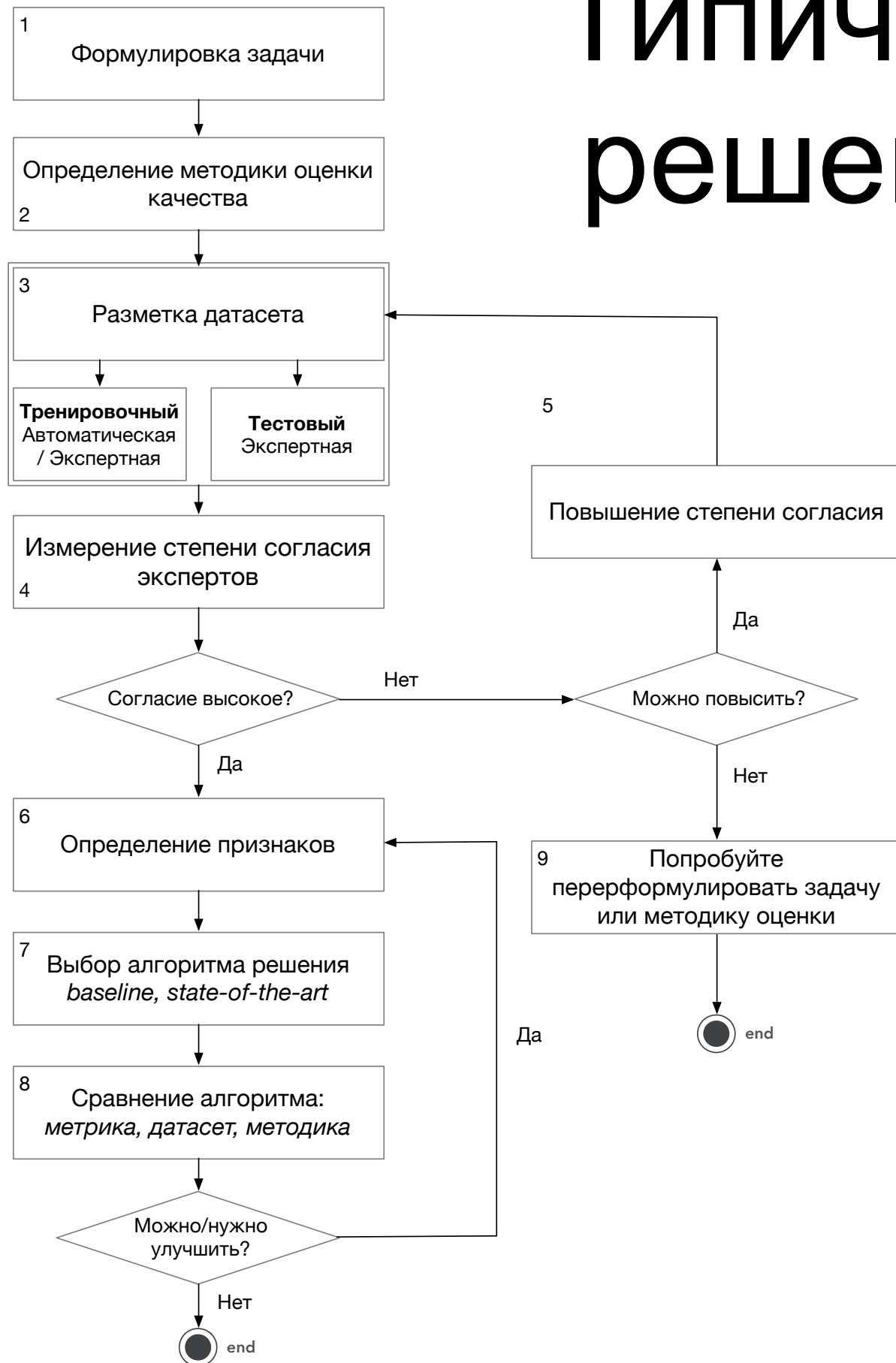
User: He says I'm depressed much of the time

ELIZA: I AM SORRY TO HEAR YOU ARE DEPRESSED

```
import re
...
sentence=re.sub("I'm", "YOU ARE", sentence)
...
sentence=re.sub(".* YOU ARE (depressed|sad) .*", "I'M SORRY TO HEAR YOU ARE \\1", sentence)
sentence=re.sub(".* all .*", "IN WHAT WAY?", sentence)
sentence=re.sub(".* always .*", "CAN YOU THINK OF A SPECIFIC EXAMPLE?", sentence)
```

# Типичная схема решения задач

- Способ оценки качества часто влияет на постановку задачи
- Тестировать алгоритм надо на данных, которые он никогда “не видел”
- Согласие экспертов показывает разрешимость задачи людьми и определяет *верхнюю границу* качества
- Для честного сравнения алгоритмов должны быть зафиксированы датасет и методика
- В реальных задачах шаги 1-5 занимают до 90% всего времени



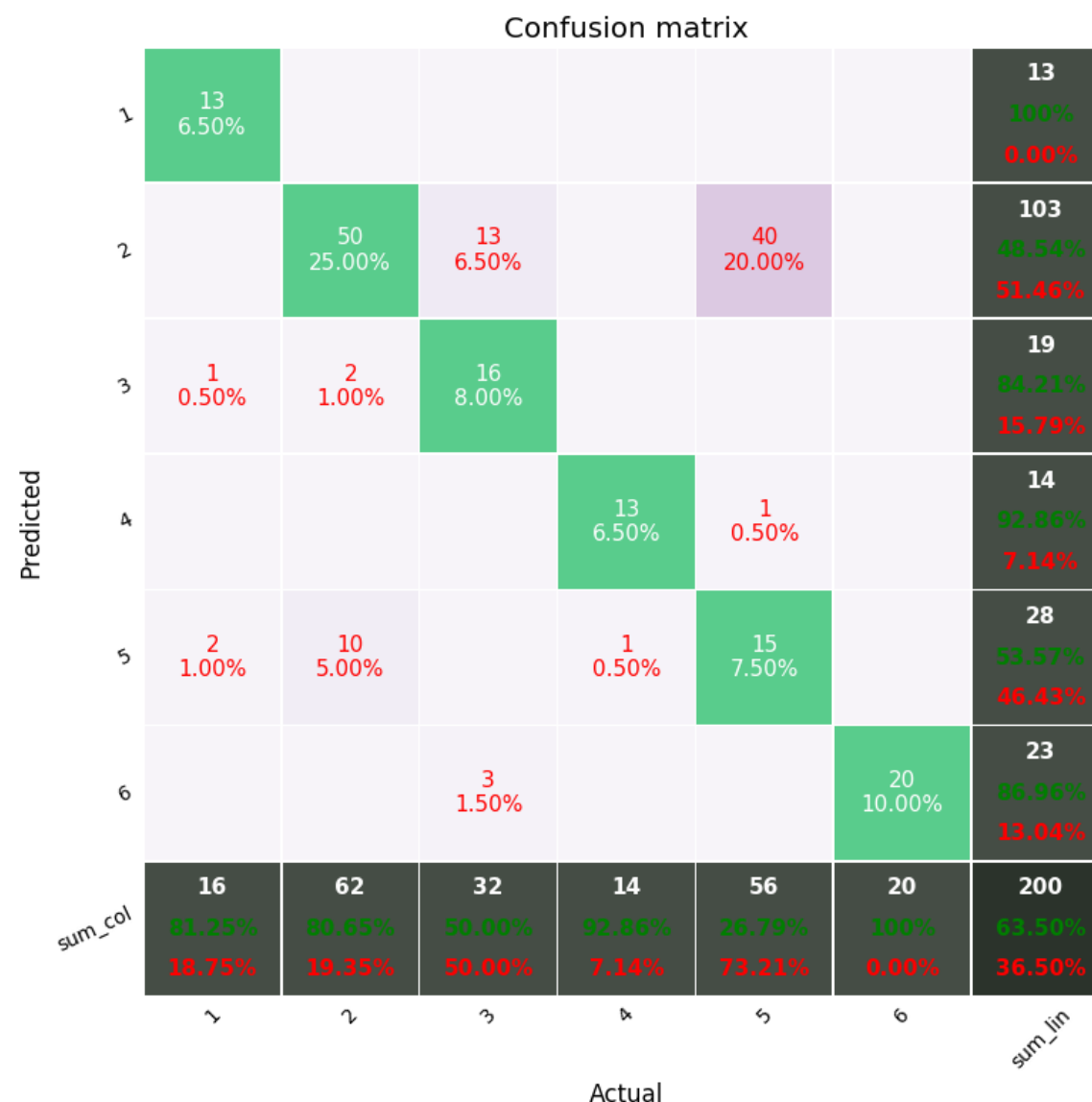
# Пример

*В начале 2019 года аппарат миссии NASA "Вояджер" (Voyager-2) пересек гелиопаузу и вышел в межзвездное пространство. Его брат-близнец (Voyager-1) сделал это шестью годами ранее. Зонды, запущенные в 1977-м для исследования планет-гигантов, до сих пор работоспособны, запасов радиоактивного топлива хватит до 2030 года.*

- Определите ключевые слова текста

# Матрица ошибок

- Confusion matrix
- Позволяет увидеть различие в ответах





# Каппа Коэна

- Cohen's kappa
- Позволяет измерить согласие двух экспертов по сравнению со случайным совпадением ответов

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

$p_o$  — наблюдаемое согласие

$p_e$  — случайное согласие

# Каппа Коэна

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

	Спам	Не спам	Сумма
Спам	20	3	23
Не спам	5	34	39
Сумма	25	37	62

$$p_o = \frac{a + d}{a + b + c + d} = \frac{20 + 34}{62} = 0.87$$

# Каппа Коэна

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

	Спам	Не спам	Сумма
Спам	20	3	<b>23</b>
Не спам	5	34	<b>39</b>
<b>Сумма</b>	<b>25</b>	<b>37</b>	<b>62</b>

$$p_{\text{spam}} = \frac{a + b}{a + b + c + d} \cdot \frac{a + c}{a + b + c + d} = \frac{23}{62} \times \frac{25}{62} = 0.15$$

# Каппа Коэна

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

	Спам	Не спам	Сумма
Спам	20	3	<b>23</b>
Не спам	5	34	<b>39</b>
<b>Сумма</b>	<b>25</b>	<b>37</b>	<b>62</b>

$$p_{\text{no spam}} = \frac{c + d}{a + b + c + d} \cdot \frac{b + d}{a + b + c + d} = \frac{39}{62} \times \frac{37}{62} = 0.38$$

# Каппа Коэна

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

$$p_e = p_{\text{spam}} + p_{\text{no spam}} = 0.15 + 0.38 = 0.53$$

$$p_o = \frac{a + d}{a + b + c + d} = \frac{20 + 34}{62} = 0.87$$

$$\kappa = 1 - \frac{1 - 0.87}{1 - 0.53} = 0.72$$

# Интерпретация

0.01 – 0.20	Незначительное
0.21 – 0.40	Удовлетворительное
0.41 – 0.60	Умеренное
0.61 – 0.80	Существенное
0.81 – 1.00	Почти идеальное

- Не всегда работает (в частности каппа может быть отрицательной)
- Нужно смотреть одновременно с матрицей ошибок
- <https://idostatistics.com/cohen-kappa-free-calculator/>

# Задание

- Что делать, если оценщиков больше двух?
  - Самостоятельно изучить статистику “Каппа Флейса (Fleiss' kappa)”
  - Объяснить как с помощью нее оценить согласие для задачи определения ключевых слов (потенциальный вопрос на экзамене)

# Резюме

- Хороший способ понять проблемы обработки текстов - сделать систему машинного перевода, вопросно-ответную систему или разговорного агента
- Обработка текста основана на формальных моделях
- Основы обработки текста лежат в компьютерных науках, математике, лингвистике, электротехнике и психологии
- Сейчас - удивительное время, когда революционные разработки используются повсеместно



# Дополнительные ресурсы

- Конференции: ACL, EACL, COLING, CoNLL, EMNLP, Диалог
- <http://www.aclweb.org/anthology-new/>
- Книги:
  - D. Jurafsky, J.H. Martin. Speech and Language processing.
  - C. Manning, H. Schutze. Foundations of Statistical Natural Language Processing
  - Ian Goodfellow, Yoshua Bengio, Aaron Courville. Deep learning. MIT Press. 2016
- Курс 2016: <https://www.youtube.com/playlist?list=PL5cBzMOPJgCUn6TbfhqilyToW5lScOdd3>

# Следующая лекция

- Методы классификации текстов
  - Логистическая регрессия
  - Машины опорных векторов (SVM)
  - Скрытая марковская модель (HMM, MEMM)
  - Условные случайные поля (CRF)
- Задача распознавания и классификации именованных сущностей (NERC)