

*Решение задач выявления плагиата,  
нечетких дубликатов и определения  
авторства текста*

Курс «Основы анализа текстовых данных»  
Кафедра управления и интеллектуальных технологий  
НИУ «МЭИ»  
2023 г.

# Что такое дубликат?

Под дубликатом текстового документа понимают копию уже ранее существующего документа. Из этого понятия вытекают два смежных определения:

*В узком смысле:* дубликат текстового документа – это документ полностью идентичный по лексическому содержанию исходному (определяется с помощью методов обработки и анализа документов).

*В широком смысле:* дубликат текстового документа – это документ идентичный по смысловому содержанию исходному. Дубликатами в данном случае называются документы, которые имеют идентичное смысловое наполнение, которое можно определить в большинстве случаев только экспертным путём и только по полному текстовому описанию.

Лексически **уникальные текстовые документы** – это документы, обладающие существенно различными наборами терминов. Лексическая уникальность в большинстве случаев соответствует смысловой уникальности текстового документа

# Классификация дубликатов текстовых документов

**Полные дубликаты** – полностью совпадающие документы. Такие дубликаты легко обнаруживаются, для этого может быть применён, например, метод расчета хеш-функции по всему тексту.

ABCDE - **ABCDE**

**Явные дубликаты** – документы, частично идентичные друг другу. К этому типу относятся документы, которые являются частичным вложением другого документа.

ABCDE – XY**ABCD**Z

**Нечёткие дубликаты** – документы, имеющие близкое лексическое содержание. Один из таких документов может быть как развитием предыдущего, так и его модификацией.

ABCDE – **ABXEYCD**Z

# Пример нечеткого дубликата статьи

## **ВЫСОКОНАДЕЖНОЕ УПРАВЛЕНИЕ ПОТОКАМИ ЖИДКОСТЕЙ И ГАЗОВ С ПОМОЩЬЮ АНАП-РЕГУЛЯТОРА**

МИСиС совместно с Институтом проблем управления РАН и производственным объединением «ОВЕН» (Финляндия) разработан автоматически настраивающийся адаптивный промышленный (АНАП) регулятор, на основе которого можно создавать системы автоматического управления, сочетающие в себе точность и быстродействие классических пропорционально-интегрально-дифференциальных (ПИД) систем с высокой надежностью и ресурсом импульсных систем при управлении с помощью регулирующих органов запорной арматуры технологическими потоками жидкостей, газов, пара и пароводяных смесей. Эффективность АНАП-регулятора наглядно иллюстрируется примерами сравнения работы последнего со стандартным автоматически настраиваемым ПИД-регулятором и импульсным регулятором. Высокие потребительские возможности АНАП-регулятора в сопоставлении с регулятором «TROVIS 6493» немецкой фирмы «SAMSON» показаны при их использовании на одном из технологических процессов Московского нефтеперерабатывающего завода.

## **ВЫСОКОНАДЕЖНОЕ УПРАВЛЕНИЕ АНАП РЕГУЛЯТОРОМ ПОТОКАМИ ЖИДКОСТЕЙ И ГАЗОВ**

Институтом проблем управления им. В.А. Трапезникова Российской академии наук (РАН) и производственной фирмой ОВЕН разработан автоматически настраивающийся адаптивный промышленный (АНАП) регулятор, на основе которого можно создавать САУ, сочетающие в себе точность и быстродействие классических ПИД систем с высокой надежностью и ресурсом импульсных систем при управлении с помощью регулирующих органов запорной арматуры технологическими потоками жидкостей, газов, пара и пароводяных смесей. Эффективность АНАП регулятора наглядно иллюстрируется примерами сравнения его работы с работой стандартного автоматически настраиваемого ПИД регулятора и импульсного регулятора. Высокие потребительские возможности АНАП регулятора показаны сравнением его работы с работой регулятора TROVIS 6493 немецкой фирмы SAMSON на одном из ТП Московского нефтеперерабатывающего завода.

# Нечеткие дубликаты. Чем это плохо?

- Показателем эффективности научной работы заявлено количество публикаций



- Необходимость написания большого количества статей



- Уменьшение времени на оригинальные исследования



- Результаты публикуются повторно.



Топуляризация полезна.



Усложняется поиск материалов для новых исследователей.



Тратится время ученых.



Искажается картина ценности идей и результатов

# Коэффициент ассоциативности Жаккара

$$J = \frac{A}{A + B + C}$$

A – число терминов присутствующих в обоих документах,

B – число терминов присутствующих в первом документе и отсутствующих во втором,

C - число терминов присутствующих во втором документе и отсутствующих в первом.

Коэффициент Жаккара изменяется в диапазоне от 0 до 1, где 0 показывает, что документы совершенно различны, а 1 – полностью идентичны

# Метод шинглов

Shingles — чешуйки.

Данный метод основан на расчёте контрольных сумм для каждой последовательности слов в тексте

Шинглы — выделенные из статьи подпоследовательности слов. Необходимо из сравниваемых текстов выделить подпоследовательности слов, идущих друг за другом по  $k$  штук (длина шингла). Выборка происходит внахлест, а не встык. Таким образом, разбивая текст на подпоследовательности, мы получим набор шинглов в количестве равному количеству слов минус длина шингла плюс один ( $\text{кол\_во\_слов} - \text{длина\_шингла} + 1$ ).

Далее для каждого шингла вычисляется хэш-сумма.

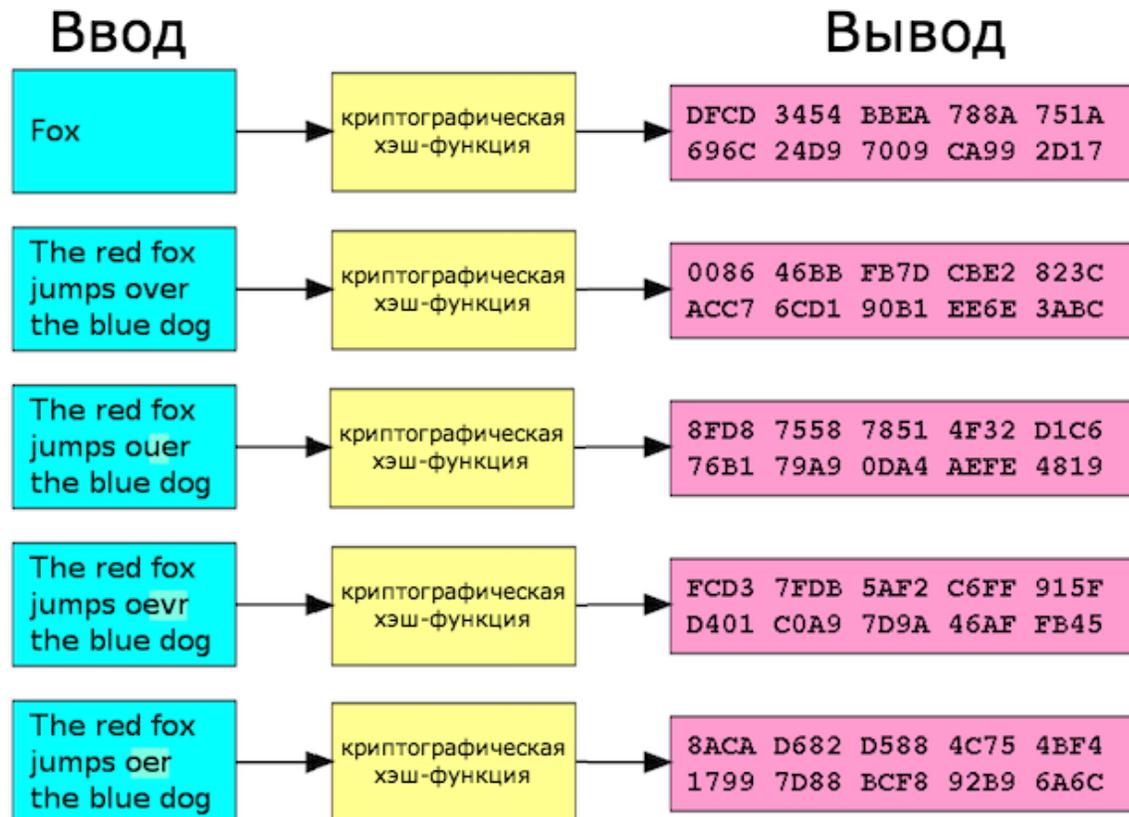
Для проверки на оригинальность документа сравнивают набор шинглов характеризующий проверяемый документ с набором шинглов характеризующим оригинальный документ.

Сравнение проводят, например, на основе коэффициентов ассоциативности (Жаккар).

Таким образом, суть метода заключается в том, чтобы производить сравнение не последовательности слов, а чисел, однозначно поставленные в соответствие этим последовательностям.

# Хэширование

**Хэширование** (*hashing*) — преобразование массива входных данных произвольной длины в (выходную) битовую строку фиксированной длины, выполняемое определённым алгоритмом. Функция, реализующая алгоритм и выполняющая преобразование, называется «хеш-функцией» или «функцией свёртки». Исходные данные называются входным массивом, «ключом» или «сообщением». Результат преобразования (выходные данные) называется «хэшем», «хеш-кодом», «хеш-суммой».



**Наиболее известные алгоритмы хэширования:**  
Secure Hash Algorithm (SHA-0, SHA-1, SHA-2, SHA-3)  
Message Digest (MD2, MD3, MD4, MD5, MD6)

**CRC16/32** — контрольная сумма (не криптографическое преобразование)

# «Супершинглы», «мегашинглы»

Принцип алгоритма шинглов заключается в сравнении контрольных сумм шинглов (подпоследовательностей) двух текстов между собой.

Проблема алгоритма заключается в количестве сравнений, ведь это напрямую отражается на производительности. Увеличение количества шинглов для сравнения характеризуется экспоненциальным ростом операций, что критически отразится на производительности.

Предлагается сравнивать не все шинглы, а лишь часть.

- Выбрать N шинглов случайным образом (плохо)
- Выбирать N шинглов с минимальными значениями хэш-суммы

Обычно<sup>1</sup> N=84.

Далее, 84 шингла разбиваются на 6 групп по 14 шинглов в каждой – «супершинглы».

Если два документа имеют сходство, например,  $p \sim 0.95$  (95%), то 2 соответствующих супершингла в них совпадают с вероятностью  $p^{14} \sim 0.95^{14} \sim 0.49$  (49%).

Таким образом, для эффективной проверки совпадения не менее 2-х супершинглов (и, следовательно, подтверждения гипотезы о сходстве содержания) каждый документ представляется всевозможными попарными сочетаниями из 6 супершинглов, которые называются «мегашинглами». Число таких мегашинглов равно 15 (число сочетаний из 6 по 2).

Два документа сходны по содержанию, если у них совпадает хотя бы один мегашингл.

<sup>1</sup> [http://rcdl2007.pereslavl.ru/papers/paper\\_65\\_v1.pdf](http://rcdl2007.pereslavl.ru/papers/paper_65_v1.pdf)

# Метод *Winnowing*

Winnowing – развеивание.

1. Текст разбивается на шинглы длиной  $k$  и хэшируется.
2. Захэшированный набор шинглов разбивается на «окна» размером  $(t-k+1)$ , где  $t$  – гарантированный порог (минимальная длина подстроки, которая гарантированно будет найдена у двух схожих документов).
3. Из каждого окна выбирается минимальное значение хэш-функции (если в последующем окне то же минимальное значение хэш-функции, что и в предыдущем, то данное значение не добавляется в набор (слепок, fingerprint))
4. Сравниваются полученные наборы значений хэш-функции каждого документа, например, с помощью коэффициента ассоциативности Жаккара.

Данный алгоритм обладает высокой скоростью работы и гарантирует, что если у двух сравниваемых текстовых документов есть общая подстрока длиной как минимум  $t$ , то она будет найдена.

В оригинальной статье<sup>1</sup> в качестве синтаксической единицы используется буква, а не слово, однако метод работает и для разбиения по словам. В этом случае, на предварительном этапе рекомендуется удалить все неинформативные признаки такие, как предлоги, местоимения, предлоги, союзы и т.д.

<sup>1</sup><http://theory.stanford.edu/~aiken/publications/papers/sigmod03.pdf>

# Метод Winnowing

Пусть на первом этапе был получен следующий набор хэш-кодов:

77 74 42 17 98 50 17 98 08 88 67 39 77 74 42 17 98

Пусть длина окна выбрана равной 4. Разбиение на окна будет выглядеть следующим образом:

(77, 74, 42, 17)

(74, 42, 17, 98)

(42, 17, 98, 50)

(17, 98, 50, 17)

(98, 50, 17, 98)

(50, 17, 98, 08)

(17, 98, 08, 88)

(98, 08, 88, 67)

(08, 88, 67, 39)

(88, 67, 39, 77)

(67, 39, 77, 74)

(39, 77, 74, 42)

(77, 74, 42, 17)

(74, 42, 17, 98)

Выбираем минимальное  
значение в каждом окне, если  
оно совпадает с предыдущим,  
то пропускаем



17 08 39 17 – получаем  
«слепок» документа

# Метод SpotSigs

Работа данного метода заключается в построении набора слов, характеризующих документ, на основе опорных слов, и суждении о схожести 2-х документов на основании совпадения наборов, характеризующих эти документы.

1. Выбираются опорные слова
2. Из текста извлекаются последовательности из  $n$  слов, стоящих за опорными словами
3. Для каждой последовательности вычисляется значение хеш-функции
4. Проводится сравнение 2-х векторов, например, с помощью коэффициента ассоциативности Жаккара или косинусной меры

Под **опорными словами** понимаются слова, которые будут встречаться с высокой частотой в любом тексте.

В качестве опорных слов для английского языка можно использовать формы модальных глаголов, а также артикли: be, is, are, was, will, would, shall, should, can, could, has, have, had, do, did, does, done, a, an, the.

Для русского языка – можно использовать стоп-слова.

Качество работы SpotSigs во многом зависит от выбора набора опорных слов.

# Метод SpotSigs, пример

*«At a rally to kick off a weeklong campaign for the South Carolina primary, Obama tried to set the record straight from an attack circulating widely on the Internet that is designed to play into prejudices against Muslims and fears of terrorism.»*

n=2  


Последовательности слов	Хеш-суммы
a: rally, kick	0xDB00886E
a: weeklong, campaign	0xA8F8D1EA
the: south, Carolina	0x1FC7939F
the: record, straight;	0xCB4E6E0E
an: attack, circulating;	0x71C381F6
the: internet, designed;	0xB8F18059
is: designed, play	0xF1B3DC8C

# Коэффициент Джаро-Винклера

Коэффициент Джаро-Винклера представляет собой меру схожести строк для измерения расстояния между двумя последовательностями символов.

$$d_w = d_j + (l * p(1 - d_j))$$

Это вариант, который в 1999 году предложил Уильям Э. Винклер (William E. Winkler) на основе расстояния Джаро (1989, Мэтью А. Джаро, Matthew A. Jaro).

Неформально, расстояние Джаро между двумя словами — это минимальное число односимвольных преобразований, которое необходимо для того, чтобы изменить одно слово в другое.

$$d_j = \frac{1}{3} \left( \frac{m}{S_1} + \frac{m}{S_2} + \frac{m - t}{m} \right)$$

где  $m$  — количество соответствующих символов (символы считаются соответствующими, если они равны и находятся не далее, чем  $(0.5 * \max\{S_1, S_2\} - 1)$  друг от друга);

$t$  — количество перестановок (вычисляется как число соответствующих символов, расположенных в различном порядке, деленное на 2);

$S_1, S_2$  — длины сравниваемых строк;

$l$  — число общих начальных символов (не больше 4-х);

$p$  — масштабный коэффициент,  $[0 - 0,25]$ , обычно 0,1

# Коэффициент Джаро-Винклера (2)

Даны строки  $s_1$  MARTHA и  $s_2$  MARHTA. Представим их пересечение в табличном виде:

	M	A	R	T	H	A
M	1	0	0	0	0	0
A	0	1	0	0	0	0
R	0	0	1	0	0	0
H	0	0	0	0	1	0
T	0	0	0	1	0	0
A	0	0	0	0	0	1

Здесь максимальное расстояние составляет  $6/2 - 1 = 2$ . В желтых ячейках приведенной таблицы указаны единицы, когда символы идентичны (имеется совпадение), и нули в противном случае.

Получается:

- $m = 6$
- $|s_1| = 6$
- $|s_2| = 6$
- Есть несовпадающие символы T/H и H/T, в результате:  $t = \frac{2}{2} = 1$

Расстояние Джаро:

$$d_j = \frac{1}{3} \left( \frac{6}{6} + \frac{6}{6} + \frac{6-1}{6} \right) = 0.9(4)$$

Чтобы найти результат Джаро — Винклера с помощью стандартного веса  $p = 0.1$  мы продолжаем искать:

$$\ell = 3$$

Таким образом:

$$d_w = 0.9(4) + (3 \cdot 0.1(1 - 0.9(4))) = 0.96(1)$$

\*[https://ru.wikipedia.org/wiki/Сходство\\_Джаро\\_—\\_Винклера#Расстояние\\_Джаро](https://ru.wikipedia.org/wiki/Сходство_Джаро_—_Винклера#Расстояние_Джаро)

Даны строки  $s_1$  DIXON и  $s_2$  DICKSONX. Получается:

	D	I	X	O	N
D	1	0	0	0	0
I	0	1	0	0	0
C	0	0	0	0	0
K	0	0	0	0	0
S	0	0	0	0	0
O	0	0	0	1	0
N	0	0	0	0	1
X	0	0	0	0	0

Здесь закрашенные клетки — это окно соответствия для каждого символа. Единицы в ячейке указывает на совпадение. Заметим, что два икса (X) не считаются совпавшими, поскольку они находятся за пределами третьего окна совпадения.

- $m = 4$
- $|s_1| = 5$
- $|s_2| = 8$
- $t = 0$

Расстояние Джаро:

$$d_j = \frac{1}{3} \left( \frac{4}{5} + \frac{4}{8} + \frac{4-0}{4} \right) = 0.7(6)$$

Чтобы найти результат Джаро-Винклера с помощью стандартного веса  $p = 0.1$  мы продолжаем искать:

$$\ell = 2$$

Таким образом:

$$d_w = 0.7(6) + (2 \cdot 0.1(1 - 0.7(6))) = 0.81(3)$$

## I-Match

Данный метод основан на идеи, что документ наилучшим образом характеризуют термины, которые имеют среднюю частоту встречаемости в тексте. В свою очередь словами, которые имеют высокую частоту встречаемости, будут предлоги, союзы, местоимения и прочие неинформативные термины, которые вовсе не уникальны для документа и плохо характеризуют его. Слова, которые имеют низкую частоту встречаемости, также плохо характеризуют документ.

Метод имеет два этапа – предварительный и основной.

На предварительном этапе, сначала находятся и удаляются неинформативные слова такие, как предлоги, местоимения, предлоги, союзы и т.д. Затем для всех исследуемых документов вычисляется словарь уникальных терминов, в который включаются только термины, обладающие средним значением  $tf-idf$ .

На основном этапе, для каждого документа формируется набор уникальных слов и определяется пересечение этого набора со словарём уникальных терминов, полученного на предварительном этапе. Полученное пересечение и представляет собой некий «слепок», («дактилограмму»,) документа. Список слов, входящих в пересечение упорядочивается, и для него вычисляется значение хеш-функции методом SHA1. Полученное значение называют I-Match сигнатурой.

Два документа считаются дубликатами, если у них совпадают I-Match сигнатуры, то есть имеет место коллизия хеш - кодов.

## I-Match (2)

Данный метод достаточно легко реализовать, и он обладает достаточно высокой эффективностью, с вычислительной точки зрения.

К сожалению, у данного алгоритма есть и свой недостаток - при небольшом изменении содержания он показывает свою неустойчивость. Чтобы исключить данный недостаток, авторы решили подвергнуть алгоритм изменению и усовершенствовать его. Была предложена новая техника многократного случайного перемешивания основного словаря. Суть модификаций заключается в следующем: к основному словарю  $L$  создаются  $K$  различных словарей  $L_1, \dots, L_K$ , которые образуются методом случайного удаления из исходного словаря определенной закрепленной части  $p$  слов. Эта небольшая группа  $p$  слов составляет приблизительно 30%-35% от исходного объема  $L$ . Для каждого документа вместо одной, вычисляется  $(K+1)$  I-Match сигнатур по алгоритму, который описан выше. Получается, что документ демонстрируется как вектор размерности  $(K+1)$ . В таком случае два документа между собой будут считаться одинаковыми, если одна из координат у них совпадает. На практике, в качестве самых оптимальных значений параметров хорошо зарекомендовали себя такие показатели:  $p = 0.33$  и  $K = 10$

# Законное (правомерное) цитирование

Цитата - (лат. Citatum/citare - приводить, провозглашать) - дословная выдержка из какого-либо текста, сочинения или дословно приводимые чьи-либо слова

Цитирование не запрещено, но должно быть правильно оформлено (ГОСТ Р 7.0.5 – 2008 «Библиографическая ссылка. Общие требования и правила составления»)

Как отмечают в своей работе Кузьминов и Юдкевич, «у выпускников, которые остаются работать в вузе, фактически нет выхода на рынок труда — и, следовательно, нет доступа к механизмам внешней экспертной оценки полученного ими образования и компетенций, а у вузов соответственно нет стимулов к выпуску конкурентоспособных специалистов для академической сферы»<sup>1</sup>.

<sup>1</sup> Кузьминов Я.И., Юдкевич М.М. Университеты в России и Америке: различия академических конвенций // Вопросы образования, 2007, №4, с. 144.

#### Список литературы

1. Кузьминов Я.И., Юдкевич М.М. Университеты в России и Америке: различия академических конвенций // Вопросы образования, 2007, №4, с. 141-158.

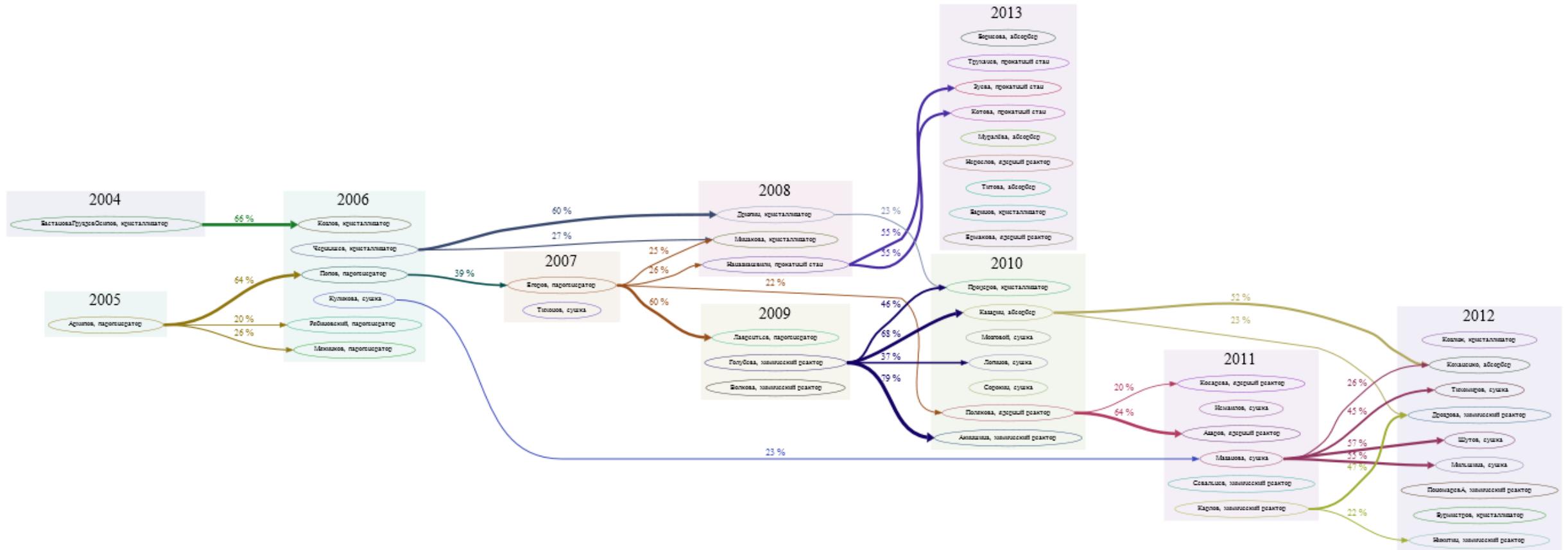
Как отмечают в своей работе Кузьминов и Юдкевич, «у выпускников, которые остаются работать в вузе, фактически нет выхода на рынок труда — и, следовательно, нет доступа к механизмам внешней экспертной оценки полученного ими образования и компетенций, а у вузов соответственно нет стимулов к выпуску конкурентоспособных специалистов для академической сферы» (Кузьминов, Юдкевич, 2007, с. 144).

#### Список литературы

1. Кузьминов Я.И., Юдкевич М.М. Университеты в России и Америке: различия академических конвенций // Вопросы образования, 2007, №4, с. 141-158.

Системы выявления плагиата должны уметь обрабатывать правильно оформленное «белое цитирование» и не включать подобные заимствования в общую оценку уникальности работы.

# Определение авторства текста



Дербенев Н. В., Козлюк Д. А., Никитин В. В., Толчеев В. О. Разработка программно-алгоритмических средств выявления плагиата в учебных и научных кафедральных работах. VI Всероссийская мультikonференция по проблемам управления (МКПУ-2013). т. 1, с. 59.

# Определение авторства текста

