***Compare sequences***

SPSS macros by Kirill Orlov

kior@akado.ru, ttnphns@gmail.com

<https://www.spsstools.net/en/KO-spssmacros>

All rights reserved

*Сравнение последовательностей или документов.* Сравнение последовательностей элементов с вычислением меры попарного сходства между последовательностями. При сравнении можно учитывать или не учитывать цепочки элементов или их местоположение – т.е. сравнивать как крядные последовательности или просто как документы. Вы можете выбрать, каким путем устанавливать сходство: максимальное паросочетание, простое выравнивание, максимальная общая цепочка и др.

*Прочтите «*[*О SPSS макросах*](https://www.spsstools.net/ru/KO-aboutmacros)*» что они такое и как их запускать.*

*Ошибка “Protected directory”.* Некоторые из макросов, описанных в текущем документе, пишут временные файлы на жесткий диск. Если вы не обладаете полными правами Администратора вашего компьютера, это может вызвать ошибку, сообщающую среди прочего: *“SPSS Statistics cannot access a file... specifies a protected directory...”* и значащую, что дефолтная директория, какую макрос хочет использовать, защищена на вашем ПК. Чтобы решить эту проблему, в окне синтаксиса скомандуйте: CD 'myfolder'., где 'myfolder' есть путь/имя некоторой папки, куда вам разрешено сохранять файлы.

# МАКРОС !KO\_SEQSIM: СРАВНЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТЕЙ (ИЛИ ДОКУМЕНТОВ) И МЕРА ИХ СХОДСТВА

Version 3, Jul 2023 (Version 1, Jun 2014). Tested on SPSS Statistics 22, 26, 28.

!KO\_seqsim docvar= *seq* /\*Короткая текстовая переменная-идентификатор последовательностей

/wordvar= *word* /\*Переменная со "словами", текстовая или числовая

/compare= FIRST /\*Задание: сравнивать последовательности все попарно и вычислить сходства

/\*(ALLPAIRS, тж п/у), сравнить первую с каждой из остальных и вычислить сходства

/\*(FIRST), или сравнить первую с каждой из остальных и вычленить длиннейшую

/\*общую цепочку (FIRSTEXT)

/trv= /\*Опционально: задать релевантности терминов: файл/массив или AUTO

/trsm= /\*Опционально: задать релевантности и схожести терминов: файл

/trotr= /\*Релевантности для "прочих" терминов (п/у =1)

/chmaxw= *4* /\*Максимальный вес словам в зависимости от длины цепочки: число

/\*или NOLIM (нет ограничения, наблюдаемая длина);

/\*после можно кл слово SQUARE

/chminw= *1* /\*Минимальный вес словам в зависимости от длины цепочки: число;

/\*после можно кл слово INI

/chpass= /\*Если задано TRV/TRSM и CHMAXW<>1: один/два проходных балла

/\*для дихотомизации перед пометкой цепочек

/chbward= /\*Учитывать цепочки с обратным порядком: NO (тж п/у) или YES

/diag= /\*Опционально: проверять диагональность матрицы совстреч: MAIN, MIDDLE или BAND

/diagls= /\*Опционально, для DIAG: параметры location и shape в "функции наказания"

/rescrnd= /\*Если задано DIAG или TRV/TRSM: перешкалировать/округлить веса совстреч,

/\*ради скорости паросочетания

/method= GREEDY PRINT /\*Метод начисления сходства: жадное паросочетание (GREEDY, тж п/у,

/\*после GREEDY можно кл слова REORD, PRINT); венгерское паросочетание (HUNGAR);

/\*максимальный след (MXTRACE); максимальная цепочка (MXCHW);

/\*максимальный вес (MXW); средний положительный вес (MEANPW);

/\*сумма весов (SUM)

/dlen= /\*Нормирующая длина в знаменатель формулы: MIN (тж п/у), MAX, MEAN, GMEAN,

/\*HMEAN, NONE (сырое сходство)

/divisor= /\*Нормирующий дефлятор весов совстреч: DIVISOR1, DIVISOR2, DIVISOR3 (тж п/у),

/\*DIVISOR4

/wmax= /\*Опционально, Wmax для дефлятора: целое положительное число

/savem1= /\*Опционально, если последовательностей две: сохранить начальную

/\*матрицу совстреч (путь/файл или заявленный массив для сохранения)

/savem2= /\*Опционально, если последовательностей две: сохранить матрицу совстреч

/\*перед начислением сходства (путь/файл или заявленный массив для сохранения)

/savemxaw= *'d:\exercise\mxw2.sav'* /\*Опционально: сохранить максимальный вес в каждом

/\*сравнении, полученный

/\*при пометке цепочек (путь/файл или заявленный массив для сохранения)

/print= MEDIUM /\*Опционально: динамическая распечатка (SHORT, MEDIUM или LONG)

/dataset= *data* /\*Если COMPARE=FIRSTEXT или задано TRSM: имя входящего массива или RENAME

/newdata= /\*Опционально: имя для выходящего массива.

Минимум надо задать DOCVAR, WORDVAR, METHOD.

Что нового Sep 2023:

Пометка цепочек с длинными последовательностями теперь идет быстрее. Скорость пометки цепочек теперь почти не зависит от параметра CHMAXW.

Макрос сравнивает между собой попарно последовательности каких-либо элементов. Последовательности похожи, если (во первых) содержание, *состав* элементов у них сходен; но особенно они похожи, если (во-вторых) при этом и *порядок* следования *смежных* элементов в них сходен. Сравнение только по составу – это «сравнение документов», а с учетом порядка смежников – «сравнение последовательностей». Макрос дает вам выбрать, настроить, в какой степени он должен учитывать сходство порядка сверх сходства только содержания. Вы можете также выявить и сохранить наиболее длинную цепочку (серию смежных элементов) с одинаковым порядком элементов в двух сравниваемых последовательностях.

Входящими данными в макрос являются две переменные (см. подробнее в описании подкоманд DOCVAR и WORDVAR). Одна из них есть идентификатор документов (последовательностей), а вторая есть собственно элементы (назовем их для удобства «словами»); каждое наблюдение массива - это «слово». Порядок слов в каждом документе – т.е. порядок наблюдений в массиве – важен: он должен быть такой, каков он в изучаемых вами последовательностях. Если вы собираетесь сравнивать документы чисто по составу, т.е. без учета порядка смежных слов, то порядок не важен.

*Отношение к разрывам («препинаниям») при сравнении последовательностей*. Вам разрешается иметь во входящих данных разрывы между элементами (например, CD∙ABA∙B) – см. подкоманду WORDVAR. Однако макрос на входе принимает позиции разрывов за фиксированные. Следствием этого является то, что CD∙ABA∙B и CD∙AB∙AB – это не идентичные последовательности для макроса. Этот макрос не сдвигает/раздвигает слова, чтобы «установить» максимальную конгруэнтность последовательностей. И данный макрос не занимается сравнением последовательностей, называемым *оптимальным паросочетанием* – употребляющим редакторские операции (вставка, удаление, замена) в целях добиться лучшего “глобального” выравнивания последовательностей.

*Если вы изучаете предложения естественного языка*. !KO\_SEQSIM не занимается лингвистической стемматизацией / лемматизацией, оставляя эту предварительную работу пользователю. Также, заглавные и строчные буквы – это разные буквы для макроса, он не игнорирует регистр.

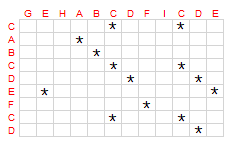
Макрос выдает главный результат (меру сходства или цепочки) как новый безымянный массив данных (но есть опция дать ему имя). По умолчанию выдается квадратная матрица сходств между всеми последовательностями. Во входящем массиве макрос создает переменную *WORD\_.#$*, и иногда *NSEQW\_.#$* и *RELEV\_.#$*. Последующие пуски макроса удаляют и пересоздают эти переменные.

**Краткий обзор макроса**

Матрица совстреч

Существует много методов и алгоритмов сравнения документов и сравнения последовательностей. У них разные идеологии и приложения. Данный макрос реализует некоторые из подходов, основанных на обработке *матрицы совстреч*. Матрица совстреч между элементами двух документов/последовательностей, к примеру, CABCDEFCD и GEHABCDFICDE, выглядит как

**Fig. 1**

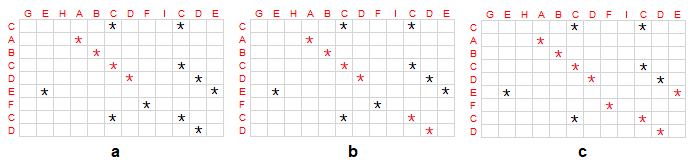


Это в сущности есть таблица смежности, соответствующая двудольному графу. Веса совстреч, тут единицы (показаны звездочками), могут быть и другими положительными числами – это зависит от задания вами входящих параметров для макроса. Раздел «Алгоритм» подробно описывает все этапы анализа. Макрос анализирует матрицу как есть, фиксированную; он не двигает позиции совпадений туда-сюда и не производит редакторских операций (вставка, удаление и т.д.) над последовательностями, чтобы «нащупать» максимальное соответствие.

Разные подходы к определению сходства

Каково сходство этих двух документов (или последовательностей)? Существует множество путей его посчитать, начислить. Вот те, которые использует макрос. Один подход (**Fig. 2 a**) – выяснить длину самой длинной цепочки из смежных совстреч. Это метод *максимальной общей цепочки*. Очевидно, что этот подход относится к сравнению документов как к сравнению последовательностей из смежных «слов». Другой подход (**Fig. 2 b**) – выяснить наибольшую диагональную (косую) сумму совстреч. Это метод *максимального выравнивания* (в его простой, базовой форме). Этот подход относится к сравнению документов как к сравнению последовательностей из «слов», но не обязательно смежных. Третий подход (**Fig. 2 c**) – учесть по-максимуму совстреч в матрице, спаривая все еще по принципу: один ряд матрицы может спариться не более чем с одним столбцом, и наоборот. Это метод *максимального паросочетания*[[1]](#footnote-1). Этот подход относится к сравнению документов более широко – как к сравнению текстов (составу из «слов»), где последовательность не обязательно важна. Но вы можете усилить его восприимчивость к последовательностям смежных слов, если запросите ценить (взвешивать) сильнее более длинные цепочки, совпадающие в двух документах. О дифференцировании весов совстреч – см. далее.

**Fig. 2**



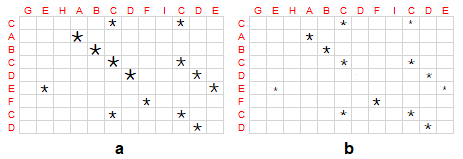
Кроме упомянутых трех подходов к определению величины сходства (общим признаком которых служит то, что они «спаривают» слова от сравниваемых документов, по принципу один-с-одним), макрос предлагает и *другие*, более простые подходы к начислению сходства. Так, вы можете просто подсчитать/суммировать все совстречи в матрице – это и будет величина сходства документов. Или вы можете выяснить один, максимальный вес в матрице. Или усредненный вес. (Два последних способа имеют смысл, если совстречи разнятся весами.)

Подходы, на которых основан !KO\_SEQSIM, применяются чаще в социальных исследованиях, нежели, скажем, в молекулярной биологии. Вы можете вполне использовать макрос в анализе текстов (соответственно подготовив сначала данные).

Разные веса у совстреч

Вы можете дать совстречам в матрице совстреч разные веса. Во-первых (**Fig. 3 a**), можно «наградить» (повысить вес) цепочки смежных слов, совпадающие в документах, за их длину (вы можете также «наказать» недостаточно длинные для вас цепочки). Если вы делаете это – значит, вы относитесь – в той или иной мере – к сравнению документов как к сравнению *последовательностей* *смежных* слов. Во-вторых (**Fig. 3 b**), вы можете «наказать» (снизить вес) совстречи в тех рядах и столбцах матрицы, где совстречи присутствуют далеко от диагональной позиции. Если вы делаете это – значит, для вас похожие документы – это почти копии друг друга: вы требуете не только совпадения по словам и/или цепочкам слов, но и по *расположению* совпадающих частей. В-третьих, вы можете изначально дать совстречам разные веса – в зависимости от *релевантности* для вас конкретных слов в деле сравнения (вы можете также дать ненулевой вес совстречам некоторых неодинаковых слов, если они для вас «синонимы»).

**Fig. 3**



ПРИМЕР 1.

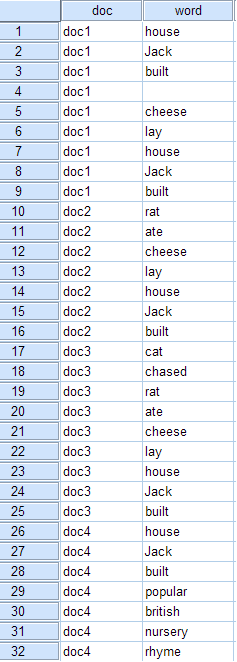
doc1 = “This is the **house** that **Jack built**. This is the **cheese** that **lay** in the **house** that **Jack built**.”

doc2 = “This is the **rat** that **ate** the **cheese** that **lay** in the **house** that **Jack** **built**.”

doc3 = “This is the **cat** that **chased** the **rat** that **ate** the **cheese** that **lay** in the **house** that **Jack built**.”

doc4 = “This Is the **House** That **Jack Built** is a **popular British nursery rhyme**”.

В сравнение документов возьмем только выделенные слова и создадим массив, готовый к работе:



Разрыв между **built** и **cheese** в doc1 введен потому, что в тексте тут находится разрыв предложений и мы не хотим, чтобы ‘**built cheese**’ могло трактоваться как цепочка смежных слов.

data list free /word (a10).

begin data

house Jack built ' ' cheese lay house Jack built

rat ate cheese lay house Jack built

cat chased rat ate cheese lay house Jack built

house Jack built popular british nursery rhyme

end data.

string doc (a8).

do if $casenum<10.

compute doc= 'doc1'.

else if $casenum<17.

compute doc= 'doc2'.

else if $casenum<26.

compute doc= 'doc3'.

else.

compute doc= 'doc4'.

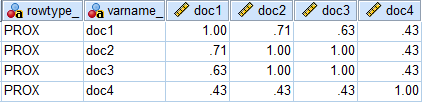
end if.

execute.

dataset name data.

* Данные введены.

!KO\_seqsim docvar= doc /wordvar= word /method= GREEDY.



* Макрос выдал квадратную матрицу коэффициентов сходства между 4-мя последовательностями попарно. По умолчанию параметры CHMAXW=1, CHMINW=1, что значит, что только словесный состав будет учитываться как основание подобия, но не порядок смежных слов (не цепочки слов).
* Метод начисления сходства выбран – жадное паросочетание.
* DLEN=MIN по умолчанию: длина более короткого из двух сравниваемых документов будет нормирующим знаменателем в формуле коэффициента сходства.
* Переменная *WORD\_.#$*, созданная макросом во входящем массиве данных – это *WORD*, перекодированная командой AUTORECODE. Случайное отрицательное число в этой переменной отвечает разрыву между словами.

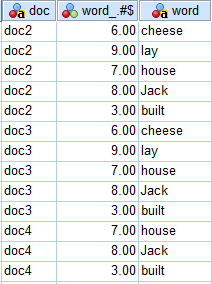
!KO\_seqsim docvar= doc /wordvar= word /chmaxw= 3 /method= HUNGAR /dlen= NONE.

* В этом пуске CHMAXW=3 придает повышенный вес словам, идущим в одинаковом смежном порядке в документах, а именно: цепочка совпадения длиной в 3 слова или больше – ценится весом 3 (например, ‘**ate cheese lay house Jack built**’). Цепочка совпадения длиной в 2 слова – ценится весом 2. Совпадение изолированным словом ценится весом 1 (в предыдущем пуске весом 1 ценились слова в независимости от того, в цепочках они или изолированы).
* Алгоритм паросочетания венгерский употреблен вместо жадного метода.
* Нормирование отменено (DLEN=NONE): сходство в сыром виде (только числитель формулы) будет выдано. На диагонали матрицы будет условное 999.

!KO\_seqsim docvar= doc /wordvar= word /chmaxw= NOLIM /method= MXCHW /dlen= NONE /print= LONG.

* В этом пуске CHMAXW=NOLIM придает повышенный вес словам, идущим в одинаковом смежном порядке в документах, а именно: цепочка совпадения длиной *l* ценится весом *l*. PRINT=LONG показывает пользователю матрицы совстреч. В них видна пометка цепочек.
* Метод начисления сходства – максимальная общая цепочка. Сходство в паре документов будет равно длине их наидлиннейшей общей цепочке, взятой в квадрат (в условиях без нормирования, DLEN=NONE).

!KO\_seqsim docvar= doc /wordvar= word /compare= FIRSTEXT /dataset= data.



* В этом пуске макрос выдает не меру сходства (число), а наидлиннейшую непрерывную подпоследовательность (цепочку смежных слов), совпадающую в документах doc1 и doc2, в документах doc1 и doc3, в документах doc1 и doc4.

**Терминология**

Документ для нас это конкретная последовательность некоторых элементов – «*слов»*. Далее мы будем писать без кавычек: слово. Так, документ ‘A B C B D’ состоит из 5 слов. Мы понимаем *документ* и *последовательность* как синонимы, кроме когда оговариваем их различение. *Термин* это различное слово, разновидность слов. В последовательности ‘A B C B D’ термин ‘A’ имеет одну *встречу* (вхождение), термин ‘B’ – две встречи, и т.д. *Совстреча* (или совпадение) – это каждый случай, когда термин (или слово) одного документа встречается с собой в другом документе. Например, в паре документов ‘A B C B D’ и ‘B F B B A’ термин A совстречается 1×1=1 раз, а термин ‘B’ совстречается 2×3=6 раз. Совстречу термина с другим термином мы также назовём совстречей[[2]](#footnote-2). *Сплошная подпоследовательность* (она же *цепочка*) – это несколько смежных слов. Так, в ‘A B C B D’ можно выделить разные цепочки, например, ‘A B C’, ‘C B’, ‘B C B D’ и др. (Синонимы этого понятия, встречающиеся в литературе: подстрока, серия, пробег.) *Сплошная подпоследовательность (цепочка) совпадения*, или общая цепочка – цепочка, встреченная как в одном, так и во втором документе. Так, в последовательностях ‘A B C B D’ и ‘B F B B A’ нет совпадающих цепочек (длиной больше чем в одно слово).

**Алгоритм**

Существуют очень разные подходы для установления подобия между двумя документами/последовательностями. Макрос !KO\_SEQSIM предлагает несколько подходов на выбор. Среди них интересны следующие три:

1. Подход *максимального паросочетания* (maximal matching): сходство двух документов – это величина взаимооднозначного паросочетания, извлеченного из матрицы совстреч между этими документами. Установление паросочетания заключается в подсчете совстреч, причем одна встреча может уделить в паросочетание не более одной совстречи (т.е. паросочетание – взаимооднозначное). Цель максимального паросочетания – спарить между собой как можно больше рядов и столбцов в матрице, иначе говоря – суметь включить в сходство как можно больше совстреч. Опционально можно признавать (награждать) существование в документах (последовательностях) одинаковых цепочек, серий смежных слов. В этом случае сходство между документами, начисленное в акте паросочетания, будет отражать сходство не только по составу слов, но и по порядку смежных слов: сходство склонно быть тем выше, чем больше совпадающих между документами цепочек и чем эти цепочки длиннее. Таким образом, паросочетательный подход может как учитывать, так и не учитывать последовательность слов в документах.
2. Подход *максимального выравнивания* (maximal aligning): сходство двух документов – это наибольшая диагональная (косая) сумма в матрице совстреч между этими документами. Этот метод относится к документу всегда как к последовательности слов, хотя не обязательно смежных. Его цель другая, чем у максимального паросочетания, и сводится к нахождению такого конкретного, частного соприлегания между последовательностями, в котором обнаруживается максимальное количество совстреч. Как и в предыдущем случае, опционально можно награждать существование в последовательностях одинаковых подпоследовательностей смежных слов.
3. Подход *максимальной общей цепочки* (maximal common chain): сходство двух документов – это наидлиннейшая цепочка совпадения (или наиболее весомая из длинных цепочек совпадения) между этими документами. Этот метод относится к документу всегда как к последовательности смежных слов. В отличие от предыдущего подхода, диагональная сумма в пределах только одной, непрерывной подпоследовательности, играет роль.

Эти три подхода являются сердцевиной алгоритма работы макроса. Они действуют в этапе 3 «Начисление сырого сходства». Все этапы алгоритма описаны ниже.

Работа, алгоритм сравнения двух документов, движется в 4 этапа (см. тж. **Fig. 16**):

1. Создание начальной матрицы совстреч. Опциональное приложение релевантностей/схожестей.
2. Опциональное изменение весов совстреч (любой из 3-х подэтапов может быть опущен):
   1. Взвешивание длиной цепочки смежных слов: совстреча, являющаяся звеном совпадающей непрерывной подпоследовательности, ценится (награждается) выше, чем изолированная совстреча. Используйте этот подэтап, если вы относитесь к сравниваемым документам как к последовательностям смежных слов. Основная подкоманда, определяющая этот подэтап – CHMAXW.
   2. Проверка диагональности: совстреча наказывается снижением веса, если совстречи в ее ряду и ее столбце отклоняются от диагональной позиции. Используйте этот подэтап, если вам важно общее расположение в сравниваемых документах частей, совпадающих между ними. Подкоманда, заведующая этим подэтапом – DIAG.
   3. Округление весов с целью их дискретизации (опционально и носит технический характер).
3. Начисление сырого сходства (числителя). Выбор совстреч для суммирования их весов в сырое сходство делается выбранным вами методом.
4. Нормирование (деление на знаменатель) в коэффициент сходства (опционально).

Ниже этапы рассмотрены подробно.

1. **Создание начальной матрицы совстреч**

Пусть последовательность D1 есть CDABAB, последовательность D2 есть EABACCCBB. Буква тут соответствует «слову». (Слово может быть любым значением, числовым или текстовым, на входе макрос всегда перекодирует значения в числа 1, 2, ... и внутренне работает уже с ними.) Создадим двоичную *матрицу совстреч* **M**, в которой 1 помечает каждую совстречу: слово одного документа встретилось с таким же словом второго документа. Матрица совстреч это двоичная матрица смежности двудольного графа.

**Fig. 4**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  | 1 | 1 | 1 |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 1 |  | 1 |  |  |  |  |  |
| B |  |  | 1 |  |  |  |  | 1 | 1 |
| A |  | 1 |  | 1 |  |  |  |  |  |
| B |  |  | 1 |  |  |  |  | 1 | 1 |

Матрица двоична – вместо нулей на рисунке показаны пустые ячейки.

*Матрица совстреч в условиях задания подкоманд TRV или TRSM*. Матрица **M** может создаваться не двоичной, а состоящей из значений диапазона [0,1], если задана п/к TRV или TRSM. В условиях TRV совстречи одинаковых слов могут быть весом от 0 до 1; эти значения называются релевантностями терминов. В условиях TRSM имеет место такая же ситуация с релевантностями и, кроме того, совстречи неодинаковых слов могут быть ненулевыми, они могут быть весом от 0 до 1; эти значения называются схожестями терминов. Релевантности и схожести для терминов предоставляет пользователь в качестве справки, но есть и опция и вычислить релевантности автоматически. Вот так может, например, выглядеть **M** при заданной TRSM:

**Fig. 5**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C | 0.1 |  |  |  | 0.6 | 0.6 | 0.6 |  |  |
| D |  |  |  |  |  |  |  | 0.2 | 0.2 |
| A |  | 1 |  | 1 |  |  |  |  |  |
| B |  |  | 0.8 |  |  |  |  | 0.8 | 0.8 |
| A |  | 1 |  | 1 |  |  |  |  |  |
| B |  |  | 0.8 |  |  |  |  | 0.8 | 0.8 |

Вместо нулей на рисунке показаны пустые ячейки. Заметьте, что некоторые совстречи терминов не с собой – схожести – не нулевые.

(Релевантность и схожесть – понятия одного порядка, они рядоположны: релевантность – это важность совстречи термина с самим собой, а схожесть – это важность совстречи двух разных терминов. В обоих случаях важность совстречи играет «за» подобие сравниваемых документов, если она близка к 1, и играет «против» их подобия, если она близка к 0. По умолчанию, т.е. в отсутствие п/к TRV/TRSM, релевантность всякого термина принята за 1, и все схожести между терминами приняты за 0.)

См. п/к TROTR, объясняющую, чем придание релевантности 0 похоже и чем отличается от физического удаления термина из данных.

1. **Изменение весов совстреч**

Этот этап или любые его подэтапы могут быть пропущены по желанию пользователя.

1. Взвешивание длиной цепочки совпадения

Если CHMAXW=1 (что есть дефолт) этот подэтап пропускается и **M** переходит в подэтап 2b. При CHMAXW=1 цепочки совпадения любой длины имеют вес 1, такой же единичный вес, как отдельные элементы матрицы совстреч. Т.е. цепочки совпадений между D1 и D2 не ценятся выше отдельных совпадений. Следствие этого: произойдет сравнение лишь словесного состава, содержания документов, но не порядка смежных слов в документах.

Если CHMAXW больше 1 или CHMAXW=NOLIM, то элементам, составляющим цепочки совпадений, придается больший *вес*, чем 1. И это значит, что будет учитываться порядок смежных слов. При CHMAXW=*n* элементы цепочки совпадения длиной ≥*n* будут помечены (награждены, взвешены) целочисленным весом *n*, элементы цепочки совпадения длиной *n*-1 будут помечены весом *n*-1, элементы цепочки совпадения длиной *n*-2 будут помечены весом *n*-2, и так далее. Значение *n* это предельный вес. Если же CHMAXW=NOLIM, то явный предел не полагается пользователем, и элементы цепочки совпадения будут помечены всегда весом, равным длине цепочки.

При CHMAXW=2 наша матрица **M** (**Fig. 4**) принимает вид:

**Fig. 6**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  | 1 | 1 | 1 |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 2 |  | 1 |  |  |  |  |  |
| B |  |  | 2 |  |  |  |  | 1 | 1 |
| A |  | 2 |  | 2 |  |  |  |  |  |
| B |  |  | 2 |  |  |  |  | 1 | 1 |

а при CHMAXW=3 или NOLIM наша матрица примет вид:

**Fig. 7**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  | 1 | 1 | 1 |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 3 |  | 1 |  |  |  |  |  |
| B |  |  | 3 |  |  |  |  | 1 | 1 |
| A |  | 2 |  | 3 |  |  |  |  |  |
| B |  |  | 2 |  |  |  |  | 1 | 1 |

Мы видим присутствие общих между D1 и D2 цепочек ABA и AB, причем в случае CHMAXW=3 или NOLIM цепочка из 3-х слов будет цениться выше, чем цепочка из 2-х слов.

Если задано CHMINW=*m*, то цепочки совпадения длиной <*m* навсегда стираются из **M**: совстречи, их составлявшие, не будут учитываться далее в алгоритме. Только совпадения длиной ≥*m* могут быть основанием сходства. Так, если CHMAXW=3 и CHMINW=2, наша **M** предстанет такой:

**Fig. 8**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  |  |  |  |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 3 |  |  |  |  |  |  |  |
| B |  |  | 3 |  |  |  |  |  |  |
| A |  | 2 |  | 3 |  |  |  |  |  |
| B |  |  | 2 |  |  |  |  |  |  |

Совстречи, образующие цепочки длиной меньше 2, т.е. не образующие цепочек в данном случае, были стерты.

Использование ключевого слова INI в CHMINW=*m* не стирает, а удерживает вес 1 у совстреч, образующих цепочки совпадения длиной <*m*. В этом случае в нашем конкретном примере **M** получится такой, как на **Fig. 7**, а не как на **Fig. 8.**

*Замечание 1*. При COMPARE=FIRSTEXT макрос исполняет подэтап 2a в режиме CHMAXW=NOLIM, с целью вычленить длиннейшую цепочку совпадения, и затем выходит из алгоритма.

*Замечание 2*. При CHBWARD=YES помечаются цепочки не только идущие из верха-лева вниз-право, но и идущие из верха-права вниз-лево. Элемент матрицы получает вес, больший из этих двух результатов.

*Резюме*. Взвешивание длиной цепочки есть умножение исходного веса *w* совстречи (т.е. 1) на фактор (целочисленный, ≥1), прямо отражающий длину цепочки совпадения, которой принадлежит эта совстреча. Совстреча начинает весить (*w*), как представитель цепочки определенной длины. Взвешивание цепочек переводит сравнение документов на рельсы сравнения последовательностей, а именно учета порядка смежных слов.

*Взвешивание длиной цепочки в условиях задания подкоманд TRV или TRSM*. Описанный подэтап 2a, награждение цепочек совпадения, нуждается в двоичной **M** на входе в него (1=совстреча, 0=несовстреча). Когда матрица **M** создавалась под диктовку подкоманд TRV/TRSM, она скорей всего недвоична, состоит из дробных значений в пределах [0,1]. Тогда необходимо сначала дихотомизовать ее. Значения ≥ пропускной уровень (он указывается пользователем в подкоманде CHPASS) перекодируются в 1, а остальные – в 0. После чего делается пометка цепочек так, как описано выше. Когда пометка будет сделана, так что **M** содержит целочисленные веса, то она будет умножена поэлементно на исходную **M** с дробными весами. Таким образом, после выхода из подэтапа 2a веса станут опять дробными.

Пусть возьмем матрицу на **Fig. 5**. При CHMAXW=3 и проходном балле CHPASS= 0.7 0.2 (1-е значение – для релевантностей, а 2-е – для схожестей) эта матрица после пометки цепочек и заключительного умножения на исходные веса окажется такой:

**Fig. 9**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  |  |  |  |  |  |
| D |  |  |  |  |  |  |  | 0.2 | 0.2 |
| A |  | 3 |  | 1 |  |  |  |  |  |
| B |  |  | 2.4 |  |  |  |  | 0.8 | 0.8 |
| A |  | 2 |  | 3 |  |  |  |  |  |
| B |  |  | 1.6 |  |  |  |  | 0.8 | 0.8 |

Если вы хотите знать максимальный вес в **M** на выходе из подэтапа (однако до того, как эта **M** будет умножена на начальные веса – в случае, когда задана TRV/TRSM), то обратитесь к п/к SAVEMXAW.

*Комментарий о нулевом весе*. Совстречи (положительные элементы матрицы **M**), допущенные к пометке цепочек и сохранившие в ней ненулевой вес, никогда позднее не могут занулиться. Другими словами, если цепочка признана (награждена), она в дальнейшем не расчленяется. Это проистекает из двух обстоятельств: (i) совстречи с релевантностью/схожестью 0 не допускаются в пометку (CHPASS не может быть 0), следовательно, «поэлементное умножение на исходную **M** с дробными весами», о котором написано двумя абзацами выше, не может превратить ненулевой присужденный в пометке вес «обратно» в нулевой; (ii) дальнейшие подэтапы 2b (подкоманда DIAG) и 2c (подкоманда RESCRND) не зануляют весов совстреч. Нулевой вес в **M** на входе в этап 3 (начисление сходства) – это либо (i) нулевой вес изначально, в начальной матрице совстреч, либо (ii) совстреча, потерявшая полностью вес в подэтапе 2a (пометка цепочек) из-за п/к CHPASS (совстреча не допущена в пометку) или из-за п/к CHMINW (совстреча убита по итогам пометки).

1. Проверка диагональности

При незаданной подкоманде DIAG (что есть дефолт) этот подэтап пропускается и **M** переходит в подэтап 2c.

DIAG снижает сходство документов, если они не копии друг друга. Эту редукцию пользователь налагает, если для него сходство между документами велико только при условии, что одинаковые между ними слова или подпоследовательности слов занимают похожие позиции внутри этих документов. В такой ситуации совстречи или цепочки совстреч в матрице **M** тяготеют к диагональному положению, и места вдали от диагонали преимущественно пусты.

Например, пользователь может мыслить, что, хотя число совстреч слева и справа на следующем рисунке одинаково (по пять), сходство слева на самом деле выше, чем справа, потому что слева единичные слова или цепочки, из них состоящие, находятся примерно в одних и тех же участках обоих документов (и потому структура близка к диагональной), справа же – находятся в разных участках:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | C | D |  |  | C | D | A | B | C |
| A | 1 |  |  |  |  |  | A |  |  | 1 |  |  |
| B |  | 1 |  |  |  | vs | B |  |  |  | 1 |  |
| C |  |  | 1 | 1 |  |  | C | 1 |  |  |  | 1 |
| D |  |  |  |  | 1 |  | D |  | 1 |  |  |  |

Или, например, пользователь может считать, что сходство на следующем рисунке слева выше, чем справа, по причине того, что слева слова не повторяются и потому занимают уникальные позиции, тогда как справа разнообразие слов недостаточное, отчего попросту нет и уникальных позиций:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E |  |  | A | A | A | A | E |
| A | 1 |  |  |  |  |  | A | 1 | 1 | 1 | 1 |  |
| B |  | 1 |  |  |  | vs | A | 1 | 1 | 1 | 1 |  |
| C |  |  | 1 |  |  |  | A | 1 | 1 | 1 | 1 |  |
| D |  |  |  | 1 |  |  | A | 1 | 1 | 1 | 1 |  |

В обоих этих примерах проверка диагональности «накажет» варианты, показанные справа: сходство для них в итоге будет ниже. Варианты слева ближе к тому положению дел, когда документы – копии друг друга. Без проверки диагональности макрос посчитает, что на рисунках справа документы имеют ту же величину сходства, что слева (допустим, при начислении сходства путем максимального паросочетания).

Проверка диагональности делатся так. Изготовляется матрица отклонений от диагонали, офсет-матрица **O**, размером с **M**. Вот офсет-матрица 5×5:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
| 1 | 0 | 1 | 2 | 3 |
| 2 | 1 | 0 | 1 | 2 |
| 3 | 2 | 1 | 0 | 1 |
| 4 | 3 | 2 | 1 | 0 |

Если матрица **M** неквадратна, офсет-матрица будет такой, какую предпочтет пользователь в подкоманде DIAG.

DIAG=MAIN. За основу берется главная (левая) диагональ:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | 0 | 1 | 2 | 3 | 4 |
| 2 | 1 | 0 | 1 | 2 | 3 |
| 3 | 2 | 1 | 0 | 1 | 2 |

DIAG=BAND. Все диагонали берутся, как полоса диагоналей:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 1 | 2 | 3 |
| 1 | 0 | 0 | 0 | 1 | 2 |
| 2 | 1 | 0 | 0 | 0 | 1 |
| 3 | 2 | 1 | 0 | 0 | 0 |

DIAG=MIDDLE. За основу берется срединная диагональ полосы:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | 0 | 1 | 2 | 3 | 4 |
| 2 | 1 | 0 | 1 | 2 | 3 |
| 3 | 2 | 1 | 0 | 1 | 2 |
| 4 | 3 | 2 | 1 | 0 | 1 |

Если полоса состоит из четного числа диагоналей, то срединных диагоналей – две.

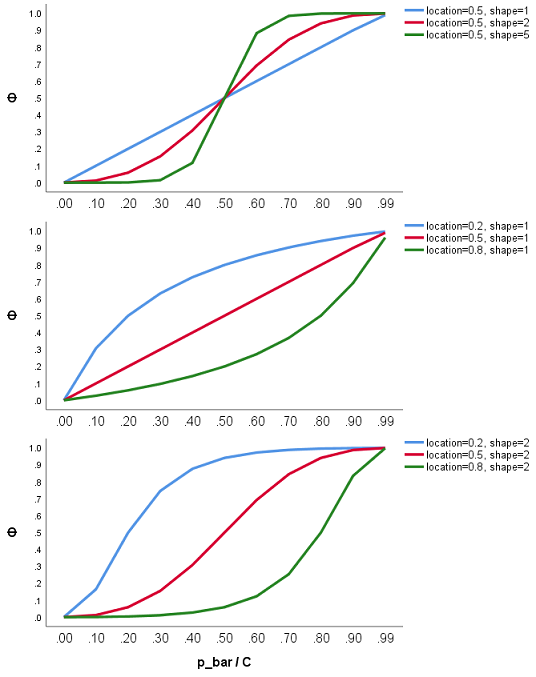
Пусть **P** есть **O**-матрица, в которой нулем замещены элементы, нулевые (пустые) в **M**-матрице, входящей в подэтап 2b. Тогда

где это средняя ненулевых элементов в ряду *r* матрицы **P** (0, если таковых нет); константа *C* это число столбцов в **P**; это средняя ненулевых элементов в столбце *c* матрицы **P** (0, если таковых нет); константа *R* это число рядов в **P**.

Функция «наказания» Φ по умолчанию – это функция тождественности (т.е. сам ее аргумент), что соответствует прямолинейной зависимости «наказания» от величины отклоненности от диагонали. Но можно потребовать криволинейную зависимость. В общем случае,

и Φ(*x*) = 0 при *x* = 0

где *x* из области [0,1), *l* это параметр положения кривой: число в диапазоне (0,1), и *s* это параметр формы, или крутизны кривой: число ≥ 1. Параметры задаются в п/к DIAGLS. При *l* = 0.5 и *s* = 1 мы имеем прямую линию, функцию тождественности, как по умолчанию. На рисунке ниже показаны кривые Φ(*x*) при разных значениях параметров положения *l* и формы *s*. Чем выше кривая, тем сильнее наказание за недиагональность. Сильно сигмоидная кривая означает контрастное отношение: терпимость к малым отклонениям и нетерпимость к большим. Функция Φ – это частная гибкая разновидность логистической функции[[3]](#footnote-3).



Вес совстречи, , какой он был перед подэтапом 2b, обновляется умножением его на фактор . Это и есть проверка диагональности – коррекция недиагональностью.

Смысл этой поправки есть наказание совстреч в зависимости от не-сконцентрированности совстреч в данном ряду и данном столбце. Все совстречи ряда *r* должны получить одинаковое наказание, тем сильнейшее, чем больше средняя отклоненность совстреч в этом ряду от диагональной позиции в нем. Подобно, все совстречи столбца *c* должны получить одинаковое наказание, тем сильнейшее, чем больше средняя отклоненность совстреч в этом столбце от диагональной позиции в нем. В итоге совстреча в ячейке *r*,*c* получает наказание гибридное, являющееся средней геометрической двух наказаний – если диагональная позиция есть и в *r*, и в *c*. Если же (из-за неквадратности матрицы) напротив ячейки *r*,*c* существует только одна диагональная позиция, то только одно, соответствующее наказание и налагается. Обратите внимание, что наказание совстречи не зависит от отклоненности от диагонали индивидуально этой совстречи, но зависит от отклоненности от диагонали всех совстреч в ее ряду и/или столбце: наказание является функцией ряда, функцией столбца, оно – коллективное.

Если совстреча лежит на диагональной позиции и в ее ряду и столбце нет совстреч вне диагональной позиции, тогда и только тогда ; иначе будет ниже 1 и, следовательно, будет наказан снижением.

Для иллюстрации покажем матрицу **M**, взятую из **Fig.6**. Вот как она будет выглядеть после редукции весов, т.е. умножения на :

**Fig. 10**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  | 0.38 | 0.27 | 0.44 |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 1.54 |  | 0.86 |  |  |  |  |  |
| B |  |  | 1.30 |  |  |  |  | 0.63 | 0.63 |
| A |  | 1.44 |  | 1.61 |  |  |  |  |  |
| B |  |  | 1.37 |  |  |  |  | 0.70 | 0.70 |

Проследим, например, совстречу B, находящуюся в ячейке (4,3). Ее вес был (**Fig. 6)** = 2. Пускай DIAG=MAIN, и тогда **O**-матрица есть:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 2 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| 3 | 2 | 1 | 0 | 1 | 2 | 3 | 4 | 5 |
| 4 | 3 | 2 | 1 | 0 | 1 | 2 | 3 | 4 |
| 5 | 4 | 3 | 2 | 1 | 0 | 1 | 2 | 3 |

и следовательно **P**-матрица есть (пустые ячейки это 0):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | 4 | 5 | 6 |  |  |
|  |  |  |  |  |  |  |  |  |
|  | 1 |  | 1 |  |  |  |  |  |
|  |  | 1 |  |  |  |  | 4 | 5 |
|  | 3 |  | 1 |  |  |  |  |  |
|  |  | 3 |  |  |  |  | 2 | 3 |

В ее ряду 4 есть три положительных элемента, и их средняя, деленная на число столбцов (9) = 0.37. В столбце 3 есть два положительных элемента, и их средняя, деленная на число рядов (6) = 0.333. Поэтому текущий вес (2) умножается на и превращается в 1.296 ≈ 1.30.

*Резюме*. Проверка диагональности есть умножение текущего веса совстречи на коэффициент (выше 0 и до 1 включительно), характеризующий данный ряд и данный столбец матрицы совстреч; коэффициент отражает сконцентрированность совстреч вблизи диагонали/ей матрицы. В отличие от награждения цепочек (подэтап 2a), проверка диагональности касается не микроструктуры последовательности (в виде смежных слов), а макроструктуры последовательности.

1. Округление веса совстречи

При незаданной подкоманде RESCRND (что есть дефолт) этот подэтап пропускается и **M** переходит в этап (3). Подэтап излишен и пропускается всегда, когда веса целочисленные, т.е. если не было подкоманд TRV/TRSM и не было подэтапа 2b.

Дробные значения **M**, особенно после этапа 2b, могут тяготеть к континуальности. В этих условиях быстродействие максимального паросочетания (этап 3, METHOD=GREEDY/HUNGAR) хуже, чем в условиях небольшого разнообразия дискретных весов. RESCRND дискретизует веса, ради скорости паросочетания, хотя это ведет к некоторой потере точности вычисляемого сходства. Пусть *dec* есть число 1, 2 или 3, заданное в RESCRND. Пусть *resc*=max(**M**). Макрос делит значения **M** на *resc*, что есть перешкалирование диапазона 0-*resc* в диапазон 0-1. Затем округляет значения до *dec* десятичных цифр, после чего умножает на *resc*. Если ненулевое значение в результате округления стало 0, оно заменяется на 10-9.

*Замечание*. Подэтап 2c не нужен для методов не паросочетательных, т.к. скорость тогда почти не зависит от дискретности весов. Тем не менее, вы можете употребить подэтап и в этом случае.

*Резюме*. Это дискретизация весов методом линейного перешкалирования-округления.

1. **Начисление сырого сходства**

Это ключевой этап: начисление (сырой) величины сходства *S* между документами D1, D2. Он регулируется подкомандой METHOD. Пользователю предлагается на выбор несколько методов, выражающих разные подходы к понятию «сходные последовательности/документы». Это в принципе разные идеи, и пользователь должен решить, что ему нужно. Так, идея паросочетания состоит в том, чтобы учесть *как можно больше* совпадений слов между сравниваемыми последовательностями, без дублирования (одно слово может спариться только раз). Идея выравнивания состоит в том, чтобы найти такое соприлегание двух последовательностей вдоль друг друга, при котором максимально много совпадений слов, оказавшихся *напротив друг друга*.

*Подход максимального паросочетания*

Используется (какой-либо) алгоритм вычисления максимального паросочетания (спаривания) между долями двудольного графа. Идея в том, чтобы так спарить в **M** ряды со столбцами, чтобы по возможности *максимизировать* сумму элементов (весов совстреч, *w*) на пересечении спаренных рядов/столбцов. Эта сумма и есть величина сходства между двумя последовательностями.

Существенна взаимооднозначность (one-to-one) паросочетания: ряд может спариться не более чем с одним столбцом, столбец может спариться не более чем с одним рядом. Другими словами, одна встреча термина в одном документе может спариться (т.е. проголосовать за сходство) только с одной встречей этого[[4]](#footnote-4) термина во втором документе. Если один документ есть, положим, “A B A C”, а второй есть “A A D A D A”, то величина сходства равна 2: каждая из двух A первого документа нашла себе в пару ровно одну A в документе-визави. Вообще, каждому элементу одной последовательности дается шанс отыскать себе в пару единственный элемент второй последовательности; и если этот последний уже спарен, то шанс пропадает. Рассмотрим примеры.

При CHMAXW=1 **(Fig. 4**) паросочетание окажется, скорей всего, таким:

**Fig. 11**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  | 1 | 1 | 1 |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 1 |  | 1 |  |  |  |  |  |
| B |  |  | 1 |  |  |  |  | 1 | 1 |
| A |  | 1 |  | 1 |  |  |  |  |  |
| B |  |  | 1 |  |  |  |  | 1 | 1 |

с величиной сходства *S*(D1, D2) = 1+1+1+1+1 = 5.

При CHMAXW=3 (**Fig. 7**) паросочетание окажется таким:

**Fig. 12**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  | 1 | 1 | 1 |  |  |
| D |  |  |  |  |  |  |  |  |  |
| A |  | 3 |  | 1 |  |  |  |  |  |
| B |  |  | 3 |  |  |  |  | 1 | 1 |
| A |  | 2 |  | 3 |  |  |  |  |  |
| B |  |  | 2 |  |  |  |  | 1 | 1 |

с величиной сходства *S*(D1, D2) = 3+3+3+1+1 = 11.

При CHMAXW=3 и дробными весами из-за подкоманды TRSM (**Fig. 9**) паросочетание окажется таким:

**Fig. 13**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | E | A | B | A | C | C | C | B | B |
| C |  |  |  |  |  |  |  |  |  |
| D |  |  |  |  |  |  |  | 0.2 | 0.2 |
| A |  | 3 |  | 1 |  |  |  |  |  |
| B |  |  | 2.4 |  |  |  |  | 0.8 | 0.8 |
| A |  | 2 |  | 3 |  |  |  |  |  |
| B |  |  | 1.6 |  |  |  |  | 0.8 | 0.8 |

с величиной сходства *S*(D1, D2) = 3+3+2.4+0.8+0.2 = 9.4.

В данных примерах во всех трех (**Fig. 11 – Fig. 13**) случаях спаривающими оказались практически одни и те же элементы **M** (эти совстречи выделены красным). В общем случае не обязательно будет так под разными режимами (заданиями). Ряды и столбцы **M** спариваются так, чтобы максимизировать *S*(D1, D2), сумму спаривающих совстреч. Интересна итоговая *S*, а не то, какие конкретно элементы стали спаривающими. Совстречи большей величины (веса *w*) имеют приоритет стать спаривающими, т.е. суммироваться в *S*.

Макрос предлагает два алгоритма паросочетания на выбор. Жадный алгоритм METHOD=GREEDY подходит открыто *иерархически*: порядок спаривания рядов/столбцов происходит в порядке рассмотрения положительных значений в **M** от большего к меньшему. Нередко эта манера – именно то, что удовлетворит интуицию пользователя: начислить первым делом сходство за счет самых весомых совстреч (вес совстреч мы дифференцировали в этапе 2 алгоритма, если этот этап не был пропущен), а потом уже – за счет того, что останется учесть (т.е. спарить). Более медленный алгоритм METHOD=HUNGAR действует щепетильно и *гарантирует* максимальное возможное *S*, достижимое вообще с данной матрицей **M**. Подробнее – см. п/к METHOD.

Элементы **M**, за счет которых произошло спаривание – спаривающие совстречи – назовем *полезными*. Таким, образом, *сырое сходство* двух документов – это сумма весов от *U* полезных совстреч:

Если этапа 2 алгоритма (изменение весов) не было, все *w* = начальные веса. Если не было подэтапа 2a (CHMAXW=1), то имеем сравнение документов только по составу. При этом, однако, мог присутствовать подэтап 2b, наказывающий некоторые *w* в матрице редукцией (следовательно, понижающий *S*). Если был подэтап 2a (CHMAXW не 1), то некоторые *w* могли получить вес 2, 3 и выше, и то есть уже, в той или иной мере, сравнение документов по составу слов и порядку смежных слов, т.е. сравнение последовательностей. При этом мог присутствовать подэтап 2b, наказывающий некоторые *w* в матрице редукцией (следовательно, понижающий *S*). Наконец, подкоманды TRV/TRSM допускающие то, чтобы веса совстреч уже на момент создания матрицы **M** были ниже 1, есть еще один – и самый первый по времени – источних понижения *S*. См. **Fig. 16.**

Сырое сходство *S* это числитель коэффициента сходства.

*Замечание*. При методе GREEDY можно заказать сделать перед спариванием переупорядочивание рядов и столбцов матрицы **M** по величине краевой суммы в ряду/столбце (кл. слово REORD). Когда **M** высокодискретна (имеет мало различных значений), но не дихотомична, это может помочь жадному алгоритму набрать более высокое, близящееся к глобальному максимуму, *S*.

*Резюме.* Максимальное паросочетание оставляет в матрице совстреч не более одного положительного веса в каждом ряду и столбце. Сумма этих спаривающих (полезных) весов есть целевая функция быть максимизированной; эта сумма есть сырое сходство документов.

*Подход максимального выравнивания*

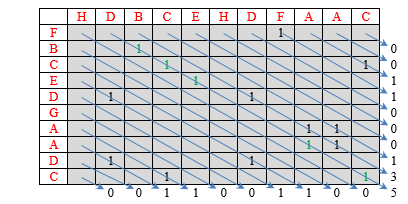
METHOD=MXTRACE. Пусть последовательность D1 есть FBCEDGAADC, последовательность D2 есть HDBCEHDFAAC. Если протягивать одну последовательность вдоль другой, смещая каждый раз на позицию, в конце концов найдем такое соположение («выравнивание»), при котором число одинаковых слов среди противолежащих слов – максимально для этой пары последовательностей. Вот это выравнивание:



Оно дает 5 совпадений (показаны зеленым). Это и есть величина сходства двух последовательностей по идее максимального выравнивания. Обратите внимание на отличие от идеи максимального паросочетания: в последовательностях есть и другие совстречные слова, например D или F. Паросочетание учло бы их: сходство было бы численно выше (оно достигло бы 9). Выравнивание же заботится максимизацией совмещенных, противолежащих совпадений, а не общим количеством совпадений. Другими словами, максимальное выравнивание выбирает самое весомое, одинаковое в последовательностях, распределение слов вдоль них[[5]](#footnote-5).

Имея матрицу совстреч **M**, величину максимального выравнивания легко установить. Она есть наибольшая из всех диагональных (косых) сумм. Так, в рассматриваемом примере максимальная диагональная сумма – 5:

**Fig. 14**



Как бы ни выглядела матрица **M** ко времени этапа начисления сходства, макрос действует одинаково – он просто калькулирует все диагональные суммы и выбирает из них наибольшую в качестве сырого сходства. Например, если **M** в этот момент выглядит как на **Fig. 6**, сходство будет равно 6. А если **M** подошла к этапу 3 в виде таком, как на **Fig. 10**, то сходство будет равно 4.45. В методе выравнивания как начислении сходства все ненулевые совстречи, лежащие на диагонали, где сумма максимальна, можно назвать «полезными»: из них складывается сходство.

*Резюме.* Максимальное выравнивание (простой, фиксированный вариант) – принцип установления сходства, который сводится к выбору максимальной из всех диагональных (косых) сумм в матрице совстреч. «Полезные» совстречи, суммирующиеся в сырое сходство, лежат на одной диагонали в матрице совстреч.

*Подход максимальной общей цепочки*

METHOD=MXCHW. Возможен, если был подэтап 2a (награждение цепочек совпадения), т.е. CHMAXW задано не как 1). Пусть *wmax* есть максимальный вес в **M** в момент окончания пометки цепочек[[6]](#footnote-6) (и *до* умножения на исходную **M** с дробными весами, если задано TRV/TRSM – см. выше, «Взвешивание длиной цепочки в условиях задания подкоманд TRV или TRSM»). Пусть *lmax* есть длина длиннейшей общей цепочки, существующей между двумя документами. Возможны следующие случаи:

**Tab. 1**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | CHMINW ≤ *lmax* | CHMINW > *lmax* |
| CHMAXW=NOLIM | | *wmax* = *lmax*  только цепочки длиной *lmax* награждены весом *wmax* | *wmax* = 1 (если кл. слово INI) или 0  поэтому METHOD=MXCHW упрощается в METHOD=MXW для данной пары документов |
| CHMAXW=*n* | *lmax* < *n* |
| *lmax* ≥ *n* | *wmax* = *n*  все цепочки с длинами [*n*, *lmax*] награждены весом *wmax* | невозможно: CHMINW не может превышать *n* |

*Примечание*. Если было кл. слово SQUARE в п/к CHMAXW, то *wmax* возведен в квадрат.

Назовем общие цепочки, элементы которых были помечены весом *wmax*, «длинными цепочками», множество *L*. В таблице видно, что это есть цепочки максимальной длины *lmax* или – иногда при CHMAXW=*n* – с длинами от *n* и выше.

По METHOD=MXCHW, сырое сходство равно сумме весов на входе в этап 3 (начисление сырого сходства) той цепочки из *L*, у которой эта сумма наибольшая.

Так, если **M** на входе в этап 3 выглядит как на **Fig. 14**, то *S* = 3: это сумма весов в цепочке BCE. Для **M** на **Fig. 12** S = 3+3+3 = 9, а для **M** на **Fig. 13** S = 3+2.4+3 = 8.4. *S* это вес самой весомой из «длинных цепочек».

Отметим частный случай: если подкоманды TRV/TRSM и DIAG не употреблялись, то веса в **M** не менялись с момента пометки цепочек в подэтапе 2a, и тогда наибольшей суммой в этапе 3 будет обладать всякая длиннейшая, длиной *lmax*, цепочка: *S* будет равно *wmax*∙*lmax*, т.е. сходство *S* тогда пропорционально длине длиннейшей цепочки совпадения.

*Резюме.* Максимальная общая цепочка – принцип установления сходства, который заключается в наборе «длинных общих цепочек» и затем во взятии самой «весомой» из них в качестве меры сходства. Как частный случай: сходство попросту отражает длину самой длинной общей цепочки. «Полезные» совстречи, суммирующиеся в сырое сходство, составляют одну цепочку совпадения.

*Другие подходы в макросе*

Прочие существующие в макросе подходы к начислению сходства – более тривиальны и не требуют особых объяснений. Можно взять в качестве сырого сходства среднюю положительных весов в **M** (METHOD=MEANPW), максимальный вес в **M** (METHOD=MXW) или суммарный вес в **M** (METHOD=SUM).

1. **Нормирование в коэффициент сходства**

Опция нормирования существует в макросе для методов (подходов) начисления сходства: паросочетательных, выравнивания, общей цепочки. Это методы, спаривающие слова, т.е. суммирующие «полезные совстречи».

Если DLEN=NONE, сырое сходство *S*, полученное на этапе 3, выдается как итоговое: *Sim*(D1, D2) = *S*. Если *S* = 0, то *Sim*(D1, D2) = 0. В иных случаях *S* нормируется в коэффициент сходства:

где *DIVISOR* объясним ниже, а *DLEN* задано п/к DLEN и по умолчанию равно длине меньшего из двух документов: . (Имеется в виду длина чистая, без учета разрывов.)

Идея нормирования заключается вообще в том, что знаменатель равен максимально возможному для данной пары документов D1, D2. Сырое сходство *S* максимально для данной пары, когда меньший из двух документов без остатка складывается из цепочек, помеченных весом *DIVISOR* на выходе из этапа 3. Формально, *Sim*(D1, D2) = 1 (максимально), если:

Повторим, вы можете ожидать *Sim*(D1, D2) = 1 только тогда, когда *все* полезные (спаривающие, суммирующиеся в сырое сходство) элементы имеют вес, равный *DIVISOR*.

*Эффект диагональности*

Формула коэффициента сходства *Sim*(D1, D2) не компенсирует отступление от диагональной структуры матрицы совстреч, поэтому *Sim*(D1, D2) может достичь 1, только если матрица совстреч диагональна. Вы можете опустить подкоманду DIAG, чтобы проверки диагональности не было.

*Эффект релевантностей (и схожестей)*

Формула коэффициента сходства *Sim*(D1, D2) не нормирует релевантностями/схожестями терминов (п/к TRV/TRSM), поэтому *Sim*(D1, D2) может достичь 1, только если все слова, образующие полезные совстречи в этапе 3, имеют полную релевантность/схожесть 1. Если некоторые термины недорелевантны (но вы не исключили их из данных), то это должно уменьшать *Sim*, и нет нужды компенсировать это.

*Эффект DLEN и выбор DLEN*

*Sim*(D1, D2) = 1 может быть только при либо при (документы равной длины). Как сказано выше, по умолчанию . Подкоманда DLEN позволяет выбрать иное значение для *DLEN* (нормирующая длина документа), нежели дефолтное. Например, вы можете предпочесть длину большего из двух документов, или усредненную длину двух документов. Вы можете отказаться от нормирования: DLEN=NONE. Также, вы можете сохранить информацию о длинах документов и о максимальных весах, наблюдаемых в **M** после этапа 1 или 2a (см. п/к SAVEMXAW), чтобы использовать ее для придумывания собственного знаменателя, нормирующего *S* в *Sim*.

*Выбор значения DIVISOR*

Если пометки цепочек не было (CHMAXW=1, подэтап 2a пропущен), *DIVISOR* = 1. Если пометка цепочек была, нужно решить, каким принять этот параметр. *DIVISOR* при нормировании убирает *эффект награды цепочек* повышенным весом – снимает инфляцию весов. Это нужно, чтобы теоретический верхний предел у *Sim* был 1, а не выше.

Пусть *wmax* есть максимальный вес в **M** в момент окончания пометки цепочек (и *до* умножения на исходную **M** с дробными весами, если задано TRV/TRSM – см. выше, «Взвешивание длиной цепочки в условиях задания подкоманд TRV или TRSM»)[[7]](#footnote-7). Как вытекает из описания подэтапа 2a «Взвешивание длиной цепочки совпадения»,

где *lmax* есть длина длиннейшей общей цепочки совпадения, существующей между данными двумя документами.

*Замечание*. В присутствии кл. слова SQUARE в CHMAXW, *wmax* – в квадрате.

Когда документов в анализе только два, естественный выбор *DIVISOR = wmax* не приносит с собой проблем. Но когда документов в анализе более двух, применение *DIVISOR = wmax* имеет последствие, которое пользователя может не устроить. Вот для примера несколько матриц **M** (на момент завершения пометки цепочек), выбранных из одного анализа с несколькими документами:

**Fig. 15**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  |  | 1 |  |  |  |
|  | 5 |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  | 1 |  |  |  |  |  |
|  |  | 5 |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  | 1 |  |  |
|  |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  |
|  |  |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |

**a b c d**

*S* = 25 9 9 5

*DLEN* = 5 5 3 5

*wmax* = 5 3 3 1

Допустим, что мы используем максимальное паросочетание как метод начисления сходства. Тогда сырое сходство *S* будет такое, как указано под матрицами. Делитель *DLEN* (длина меньшего документа) везде 5, кроме случая **c**, где он 3.

Если *DIVISOR =* *wmax*, то *Sim* равно: 25/(5∙5) = 1 в **a**, 9/(3∙5) = 0.6 в **b**, 9/(3∙3) = 1 в **c**, 5/(1∙5) = 1 в **d**. Хотя пользователю может нравиться, что в обоих случаях **a** и **c** – где *S* является максимально возможным – *Sim* = 1, он увидит, что в **d** *Sim* тоже получилось 1. Это может не понравиться пользователю, который затевал пометку цепочек ровно для того, чтобы получить сходство в случаях **a** и **c** выше, чем в случае **d**. Это преимущество над **d** было зафиксировано в величине *S*, но теперь оно потерялось при нормировании в *Sim*. Выбор *DIVISOR = wmax* – это *решение, забывающее взвешивание цепочек*; оно превращает при нормировании награжденные повышенным весом совстречи обратно в вес 1. Эта опция существует в макросе как DIVISOR=DIVISOR1.

Посмотрим на другой выбор *DIVISOR*. Пусть *DIVISOR = Wmax*:

*Wmax =* max(*wmax\_1*, *wmax\_2, …, wmax\_q*) – наибольшее из всех *q* *wmax* (*q* это число пар сравниваемых документов в текущем анализе).

*Wmax* есть константа для всего текущего анализа. Положим, что в нашем примере *Wmax =* 5. Тогда *Sim* равно: 25/(5∙5) = 1 в **a**, 9/(5∙5) = 0.36 в **b**, 9/(5∙3) = 0.6 в **c**, 5/(5∙5) = 0.2 в **d**. Пользователя, который употреблял награждение цепочек, удовлетворит этот результат в случае **d**: сходство мало́, как и ожидалось. Зато теперь сходство понизилось и в случае **c**, – это может не понравиться тому пользователю, который считает, что в **a** и **c** нормированное сходство одинаково и максимально. Выбор *DIVISOR = Wmax* – это *решение, утверждающее взвешивание цепочек*; причем это *суровое* решение. Объясним, почему. Когда мы делим *S* на *Wmax*, мы «наказываем» совстречи за то, что они весят меньше, чем *Wmax*. Мы говорим, обращаясь к случаю **c**: «вот есть матрица **a**, где веса достигли даже *Wmax* (5), а у вас, **c**, они показали более скромный успех (3)». Но справедлив ли такой «упрек» к случаю **c**? Нет, потому что в случае **c** вес и не мог бы при пометке цепочек достить 5: не позволяет длина меньшего документа, равная там 3. В случае **b** – мог бы, т.к. длина меньшего документа там 5. Поэтому вышеозначенный «упрек» справедлив в случае **b**, но не в случае **c**. Вот почему *DIVISOR = Wmax* это «суровое» решение. Эта опция существует в макросе как DIVISOR=DIVISOR2.

Из предыдущего абзаца следует, что *справедливым решением, утверждающим взвешивание цепочек*, является *DIVISOR =* min(*Wmax*, *wulimit*), где *wulimit* есть верхний предел для *wmax* в данной матрице **M**. Как сказано, этот предел задается длиной меньшего из двух документов; но он может быть и меньше в случае CHMAXW=*n*:

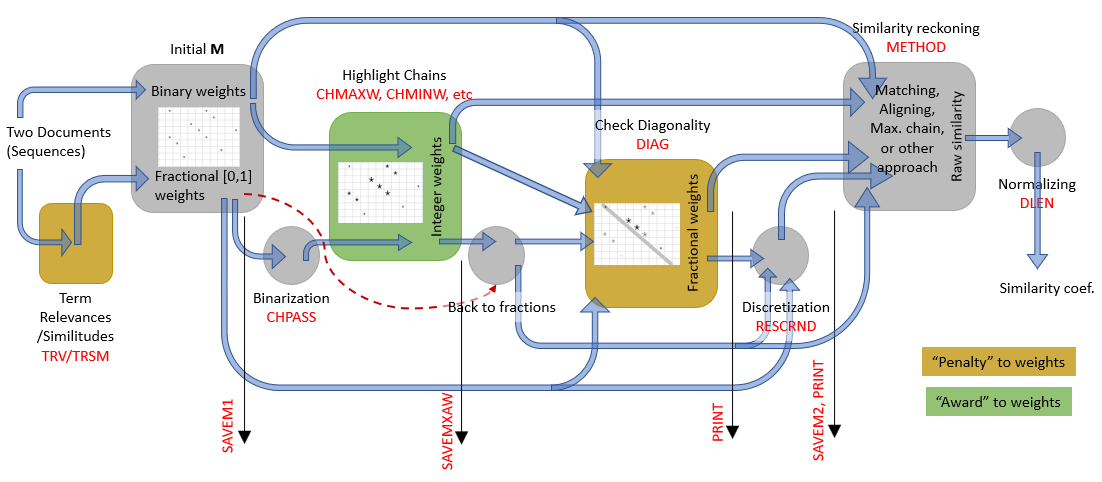
*Замечание*. В присутствии кл. слова SQUARE в CHMAXW дли́ны документов |D|, как и *n* – в квадрате.

Если в данной паре документов *wulimit* меньше, чем *Wmax*, тогда эта *wulimit* используется вместо *Wmax* в данной паре в качестве *DIVISOR*. И тогда *Sim* в случае **c** снова оказывается максимальным, 9/(3∙3) = 1, в других же трех случаях (**a**, **b**, **d**) *Sim* остается таким же, как было при *DIVISOR = Wmax*. Опция *DIVISOR =* min(*Wmax*, *wulimit*) существует в макросе как DIVISOR=DIVISOR3, и она используется по умолчанию.

Еще один вариант: *компромиссное решение, утверждающее взвешивание цепочек*. Это «компромисс» между «суровым» и «справедливым» вариантами: *DIVISOR =* harmonic\_mean(*Wmax*, *wulimit*). Между двумя значениями, *Wmax* и *wulimit*, вычисляется гармоническая средняя. Тогда *Sim* в случае **c** будет незначительно сниженным, 9/(3.75∙3) = 0.8 (это между 1 «справедливого» варианта и 0.6 «сурового» варианта), а в других трех случаях (**a**, **b**, **d**) *Sim* остается таким же – в нашем примере, – как было при *DIVISOR = Wmax*. Опция *DIVISOR =* harmonic\_mean(*Wmax*, *wulimit*) существует в макросе как DIVISOR=DIVISOR4.

*Замечание*. Если документов в анализе всего два, DIVISOR1 = DIVISOR2 = DIVISOR3 = *wmax*.

*Замечание*. Есть опция (подкоманда WMAX) установить *Wmax* на желаемую пользователем константу, не зависящую от набора данных, т.е. от значений *wmax*.



**Fig. 16**. Блок-схема алгоритма макроса.

**Быстродействие**

Сравнение последовательностей – время- и ресурсо-затратно (в отличие от частотного сравнения документов). Макрос вычисляет сырое сходство *S*(D1, D2), беря документы попарно, – стало быть, время сильно зависит от количества документов. И, само собой разумеется, от их размера (длины). Все операции с текущей матрицей совстреч **M** (см. «Алгоритм») делаются в RAM-памяти. Будучи написан на высокоуровневом интерпретируемом языке – каким является синтаксис SPSS – макрос неминуемо ограничен в своих скоростных достоинствах; в связи с чем он рассчитан в основном для задач малого или среднего масштаба.

* METHOD=HUNGAR намного медленнее жадного метода паросочетания и не рекомендован, если у вас документы в тысячи слов. METHOD= GREEDY REORD несколько медленнее, чем METHOD=GREEDY.
* METHOD=MXTRACE (выравнивание) и METHOD=MXCHW (максимальная цепочка) обладают средней скоростью. Методы SUM, MEANPW, MXW – очень быстры.
* Задание DIAG – существенный источник замедления. Замедлением является и задание TRSM/TRV. В указанных случаях важная причина замедления процесса паросочетания (это METHOD=GREEDY/HUNGAR) – большое разнообразие весов в матрице совстреч. Для борьбы с ним используйте п/к RESCRND.
* PRINT= MIDDLE или LONG дают замедление. Не используйте LONG, когда документы длинные.

***Подкоманды***

**DOCVAR**

Укажите короткую текстовую (до 8 байтов шириной) переменную, значения которой – имена документов (последовательностей). Значения должны быть уже сгруппированы вместе: наблюдения массива, относящиеся к одному документу, должны быть рядом. Значения должны начинаться с буквы или символа @ и являться единым словом – эти имена документов станут именами переменных в массиве, который выдаст макрос. В DOCVAR-переменной не может быть пустых ячей и пропущенных значений.

**WORDVAR**

Укажите текстовую или числовую переменную, значения которой – это элементы последовательностей («слова» документов). Каждое наблюдение массива это элемент, или слово. WORDVAR-переменная на входе будет подвергнута макросом автоперекодировке SPSS-командой AUTORECODE в целочисленные коды 1, 2, 3, ..., и анализ будет проводиться уже с ними. Автоперекодированная переменная – ее имя *WORD\_.#$* – сохранится во входящем массиве. Повторный пуск макроса перезаписывает ее. Используйте п/к PRINT (см.), чтобы видеть в Viewer протокол автоперекодировки.

*Разрыв vs удаление слова*. Пропущенное значение или пробел в WORDVAR-переменной трактуются как «разрыв», а не слово. Это означает: слова документа, разъединенные разрывом, не смогут быть рассмотрены как смежные, как члены цепочки. Разрыв это как бы слово, разделяющее соседние слова, само не участвующее никак в анализе, оно не входит даже в длину документа. Присутствие или отсутствие разрывов имеет значение, только когда параметр CHMAXW не 1, т.е. когда вопрос о череде смежных слов поднят. Если же CHMAXW=1 (что есть дефолт), то разрыв эквивалентен удалению (или фильтрованию прочь) сло́ва из входящих данных. Удаление слова смыкает окружающие его слова, делая их смежными, однако при CHMAXW=1 вопрос о смежности, о цепочках, просто не встает. О придании термину нулевой релевантности (еще одна возможность) – см. в п/к TROTR.

**COMPARE**

Укажите работу для макроса:

ALLPAIRS - (тж. по умолчанию/незаданию) выдать квадратную симметричную матрицу сходств между всеми документами попарно.

FIRST - выдать вектор сходств между 1-м документом и каждым другим документом.

FIRSTEXT - экстрагировать длиннейшую цепочку (непрерывная подпоследовательность элементов) совпадающих между документами слов. Своя цепочка извлекается из сравнения 1-го документа с каждым другим документом. Эта опция игнорирует подкоманды CHMAXW/CHMINW, CHBWARD, TRV/TRSM/TROTR, DIAG/DIAGLS, RESCRND, METHOD, DLEN/DIVISOR/WMAX, PRINT, поскольку она не вычисляет меру сходства. COMPARE=FIRSTEXT требует задания п/к DATASET (см.).

COMPARE=FIRSTEXT извлекает (показывает) только одну длиннейшую цепочку совпадения из каждого сравнения. Если длиннейших цепочек одинаковой длины существует несколько, макрос предупреждает об этом. Вы можете увидеть все цепочки совпадения, существующие между двумя документами, воспользовавшись п/к SAVEM1.

ПРИМЕР 2. Поисковый запрос.

data list free /content (a8).

begin data

A B C D E F G H

C B M A D B B M L J A D C J

C B L N B C D E F K D M M C D

A K A A B C D E F G H J J A A C D L M K A B C D E F G H H

E A B C D E F G J C D E F G H A K F

end data.

string seq (a8).

do if $casenum<9.

compute seq= 'seq0'.

else if $casenum<23.

compute seq= 'seq1'.

else if $casenum<38.

compute seq= 'seq2'.

else if $casenum<67.

compute seq= 'seq3'.

else.

compute seq= 'seq4'.

end if.

execute.

dataset name data.

* Данные введены.

!KO\_seqsim docvar= seq /wordvar= content /compare= FIRST /chmaxw= 8 /chminw= 8 /dlen= NONE

/method= SUM.

* COMPARE=FIRST сравнивает 1-й документ с каждым другим. Это подходит для задачи «поисковый запрос».
* Тут первая последовательность из *SEQ* (ее имя seq0) сравнивается с каждой прочей. Исследователь сделал первую последовательность своим поисковым запросом: она вся есть искомая фраза ‘A B C D E F G H’ из 8 слов в нужном порядке, и исследователь хочет знать, встречается ли целиком эта последовательность внутри остальных, более длинных документов.
* CHMAXW=8 значит, что встреча (в остальных документах) этой цепочки из 8 слов будет восьмикратно награждена, а CHMINW=8 значит, что любые частичные совпадения, т.е. короче 8 слов, во внимание не принимаются. Следовательно, сырое (не нормированное длинами документов) сходство между поисковой последовательностью seq0 и всяким прочим документом может быть, по методу METHOD=SUM (т.е. просто сумма элементов в матрице совстреч): или *k*∙64 (найдено), или 0 (не найдено). *k* это число экземпляров найденных последовательностей.
* В нашем примере сходство между seq0 (запрос) и seq3 равно 128 – т.е. в документе seq3 присутствуют две штуки последовательности ‘A B C D E F G H’.

!KO\_seqsim docvar= seq /wordvar= content /compare= FIRST /chmaxw= 6 /chminw= 6 /dlen= NONE

/method= SUM /print= LONG.

* В этих пусках с теми же данными и тем же поисковым запросом (seq0) исследователя заинтересовали любые совпадения длиной не меньше 6 подряд идущих слов из тех 8 слов. Поэтому он снизил требования, параметры CHMAXW и CHMINW, до 6. Теперь сходство величиной 48 (т.е. 6∙8) будет признаком наличия в документе всей цепочки из 8 слов, сходство 42 (6∙7) обнаруживает совпадение по некоторым 7 подряд идущим словам из 8, а сходство 36 (6∙6) – совпадение по некоторым 6 подряд идущим словам из 8. Так будет, если цепочка найдена в одном экземпляре.
* Взглянем, что получилось в нашем примере. Для лучшей ориентации в результатах, п/к PRINT=LONG распечатала нам матрицы совстреч. Между seq0 (запрос) и seq3 сходство METHOD=SUM равно 96. Это потому, что в seq3 есть два экземпляра последовательности ‘A B C D E F G H’, каждая из них «весит» 6∙8. Между seq0 и seq4 сходство равно 78. Эта сумма весов есть 6∙7 + 6∙6: ибо в seq4 найдены два экземпляра неполного совпадения с поисковым запросом: один длиной из 7 слов (именно, ‘A B C D E F G’), а второй из 6 слов (‘C D E F G H’).

**TRV**

Вам понадобятся подкоманды TRV/TRSM и TROTR тогда, когда вы хотите придать разным словам разные важности в деле сравнения документов и вычисления сходства между ними. TRSM разрешает заодно обозначить и синонимию слов. TRV/TRSM действуют на этапе создания начальной матрицы совстреч (см. «Алгоритм») и сказываются на дальнейших этапах.

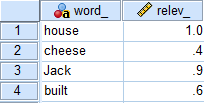
Опциональная подкоманда TRV (“term relevance vector”) позволяет придать различным словам – терминам – разные релевантности (важности) в деле сравнения документов. Релевантность это число в диапазоне [0,1], где 0 значит «термин совсем не важен для сравнения документов» и 1 значит «термин полностью важен для сравнения документов». По умолчанию/незаданию подкоманды релевантность всякого термина принята за 1, что значит: все слова входящих документов одинаково и максимально важны для их сравнения. Вы придаете терминам разные релевантности подкомандой TRV либо подкомандой TRSM (вы вправе задать только одну из этих двух подкоманд).

В TRV, укажите одно из двух:

AUTO - релевантность всякого термина макрос вычислит как , где *d* – число документов (из всего введенного в макрос корпуса документов согласно DOCVAR-переменной), в которых встречается данный термин. Таким образом, термин, встречающийся лишь в немногих документах из всех анализируемых сейчас макросом документов, становится важнее для попарного сравнения документов, нежели термин, встречающийся во многих из этих документов. Термин же, встречающийся только в одном документе, получит важность (релевантность) 0.

*файл/массив* - релевантность терминов задается пользователем, путем указания на внешний .SAV файл или открытый массив, являющийся справкой о важности терминов. *Путь/имя* файла в кавычках или апострофах или *имя* массива. Терминам, существующим в данных, но отсутствующим в этой справке, будет назначена релевантность TROTR (см. подкоманду, по умолчанию это 1).

Справочный файл/массив должен состоять из переменных *WORD\_* и *RELEV\_*. Первая – текстовая или числовая, соответстует входящей в макрос WORDVAR-переменной и содержит список слов (терминов), которым вы хотите придать индивидуальную релевантность; вторая – это сами релевантности, числа от 0 до 1. Если WORDVAR-переменная – текстовая, то *WORD\_*, тоже текстовая, должна иметь ту же ширину, что и она. Пропуски в переменной *RELEV\_* равнозначны отсутствию: таким терминам будет назначена релевантность TROTR. Пример справочного файла для ПРИМЕРА 1:



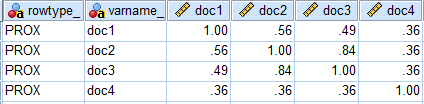
*Придание термину нулевой релевантности*. См. обсуждение этого вопроса в п/к TROTR.

Т.к. релевантности меньшие, чем 1, т.е. неполная важность, есть коэффициенты, понижающие величину сходства, то бессмысленно сравнивать напрямую сходства, полученные при заданной п/к TRV с результатами, полученными без нее.

TRV создает во входящем массиве переменную *RELEV\_.#$*, показывающую релевантности слов. TRV=AUTO кроме того создает переменную *NSEQW\_.#*, показывающую число документов, в которых слово встречается.

ПРИМЕР 3. Используйте данные ПРИМЕРА 1.

!KO\_seqsim docvar= doc /wordvar= word /trv= relev /method= GREEDY.



* *RELEV* это открытый неактивный массив, содержащий релевантности терминам **house**, **cheese**, **Jack**, **build** (см. изображение его выше); для остальных терминов релевантность по умолчанию =1. Остальное задание то же, что было в 1-м пуске в ПРИМЕРЕ 1. Релевантность ниже 1 понижает величину сходства.

**TRSM**

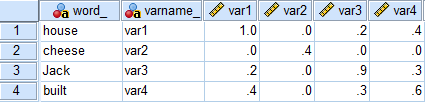
Опциональная подкоманда TRSM (“term relevance & similitude [=similarity] matrix”) является расширенной версией п/к TRV. TRSM позволяет придать различным словам – терминам – разные релевантности (важности) в деле сравнения документов, а также указать степень схожести между разными терминами. Для пользователя высокая схожесть двух терминов может носить субъективный смысл синонимии (термины взаимозаменяемы) или контекстного сродства (термины часто совстречаются в однотематических документах).

И релевантность, и схожесть – это числа в диапазоне [0,1], где 0 значит нулевая важность (или схожесть) и 1 значит полная важность (или схожесть). Релевантность и схожесть – понятия одного порядка, они рядоположны: релевантность – это важность совстречи термина с самим собой, а схожесть – это важность совстречи двух разных терминов. В обоих случаях важность совстречи играет за подобие сравниваемых документов, если она близка к 1, и играет против их подобия, если она близка к 0.

Подкоманда TRV позволяет ввести релевантности – в виде вектора. Подкоманда TRSM позволяет ввести как релевантности, как и схожести, - это квадратная симметричная матрица. Вы вправе использовать только одну из двух подкоманд. По умолчанию/незаданию этих подкоманд релевантность всякого термина принята за 1, что значит: все слова входящих документов одинаково и максимально важны для их сравнения; и схожесть всех терминов принята за 0, что значит, что разные термины априори несвязаны: они *разные*.

В TRSM укажите внешний справочный .SAV файл – *путь/имя* в кавычках или апострофах. Терминам, существующим в данных, но отсутствующим в этой справке, будет назначена релевантность TROTR (см. подкоманду, по умолчанию это 1). Что касается схожести для терминов, отсутствующих в этой справке, то она фиксированно принимается за 0.

Справочный файл должен начинаться с переменной *WORD\_*. Она – текстовая или числовая, соответстует входящей в макрос WORDVAR-переменной и содержит список слов (терминов), которым вы хотите придать индивидуальную релевантность/схожесть. Если WORDVAR-переменная – текстовая, то *WORD\_*, тоже текстовая, должна иметь ту же ширину, что и она. Второй должна идти короткая (до 8 байтов шириной) текстовая переменная *VARNAME*\_ с именами переменных, образующих тело матрицы. Далее должно идти тело квадратной симметричной матрицы. На ее диагонали находятся релевантности терминов. Схожести между разными терминами – ее внедиагональные элементы. Пропущенных значений в файле должно не быть. Посторонних переменных должно не быть. Пример справочного файла для ПРИМЕРА 1:



Обычно схожести (внедиагональные значения) в вашей матрице будут меньше релевантностей (диагональных значений), потому что как правило наличие ‘A’ и ‘A’ в обоих документах сильнее действует за сходство, чем присутствие ‘A’ в одном и ‘B’ в другом. Но, вообще говоря, так не обязательно.

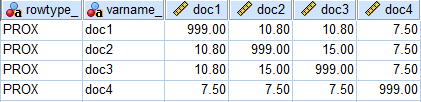
Если все внедиагональные элементы нулевые, то эффект будет тот же, что ввести только диагональные элементы как вектор релевантностей подкомандой TRV (последнее будет выгоднее с точки зрения скорости исполнения).

TRSM требует задания п/к DATASET (см.).

ПРИМЕР 4. Используйте данные ПРИМЕРА 1.

!KO\_seqsim docvar= doc /wordvar= word /trsm= 'd:\exercise\trsm.sav' /trotr= .7 /chpass= .3 .3

/chmaxw= 3 /method= HUNGAR /dlen= NONE /dataset= data.



* *TRSM* это файл, матрица, содержащая релевантности и схожести терминам **house**, **cheese**, **Jack**, **build** (см. изображение матрицы выше); для остальных терминов релевантность задана 0.7 (п/к TROTR). Поскольку заказано награждение общих подпоследовательностей (CHMAXW=3), необходимо указать проходной балл для пропуска в процедуру награждения: это п/к CHPASS, 1-е значение касается релевантностей, 2-е – схожестей. В данном случае оба порога одинаковы, 0.3. Поскольку схожесть между **house** и **cheese** (0.2) ниже этого порога, схожесть между этими двумя словами примется за 0 и не сможет участвовать в начислении сходства документов.
* Остальное задание то же, что было во 2-м пуске в ПРИМЕРЕ 1.

**TROTR**

Эта подкоманда (“term relevance, other”) действует, если задана TRSM, или TRV задана как файл/массив. Она назначает релевантность всякому «другому» термину – термину, не упомянутому в файле-справке. Укажите число в диапазоне [0,1]. По умолчанию это 1, т.е. полноценная важность в деле сравнения. TROTR касается только релевантностей, т.е. весов совстреч терминов с собой.

*Придание термину нулевой релевантности*. Если вы зададите релевантность 0 каким-то терминам – либо через TROTR, либо в векторе TRV или в матрице TRSM, – то встает вопрос, равнозначно ли это тому, что удалить или перекодировать в пропуски («разрывы») эти слова в данных перед пуском макроса? (При задании TRSM, положим сейчас: все схожести =0, т.е. эта TRSM эквивалентна TRV.) Ответ таков. Для начисления *сырого* сходства задание термину релевантности 0 эквивалентно его перекодировке в разрыв (а если CHMAXW=1, то эквивалентно и удалению термина из данных). Однако для получения *нормированного* сходства (см. п/к DLEN) это не вполне эквивалентные вещи, поскольку слово с релевантностью 0 входит в длину документа, а разрыв, как и удаленное слово, не входит в длину документа. О разнице между разрывом и удалением – см. в п/к WORDVAR.

**CHMAXW**

Вам понадобятся подкоманды CHMAXW, CHMINW, CHBWARD и CHPASS, если вы хотите сравнивать документы как последовательности смежных слов, а не просто как наборы слов.

CHMAXW (“chain maximal weight”) – основная из этих 4-х подкоманд. CHMAXW назначает максимальный наградительный вес совстречам, образующим сплошные подпоследовательности (цепочки смежных слов), совпадающие у двух сравниваемых последовательностей (документов). Укажите целое *число* от 1 и выше или кл. слово NOLIM (нет предела). См. иллюстрацию в «Алгоритм: Взвешивание длиной цепочки совпадения».

По умолчанию/незаданию подкоманды, CHMAXW=1, что значит, что цепочки совпадений не будут цениться больше совпадений отдельных слов. Это отвечает задаче сравнить документы только по словесному *содержанию*. CHMAXW же больше 1 есть уже сравнение по *содержанию и порядку смежных* слов. Чем больше CHMAXW, тем более дифференцированно ценятся цепочки совпадений разной длины. Этот параметр прямо связан с длиной цепочки. Так, при CHMAXW=4 цепочки длиной 4+ слов, одинаковые между двумя сравниваемыми документами, получат вес 4 (каждое слово такой цепочки); цепочки длиной 3, одинаковые между документами, получат вес 3; цепочки длиной 2 получат вес 2. Вес 1 останется у совпадающих между двумя документами слов, которые не входят в цепочки совпадения. При CHMAXW=2 цепочки длиной 2+ получат вес 2. Вес 1 останется у совпадающих слов, которые не входят в цепочки совпадения. Таким образом, CHMAXW=*число* – это предельный наградительный вес для элемента цепочки совпадения.

При CHMAXW=NOLIM предел не кладется, и вес элемента цепочки будет всегда равен наблюдаемой длине цепочки, какой бы длинной она ни оказалась.

Опциональное добавленное кл. слово SQUARE делает веса квадратными, нежели линейными. Например, CHMAXW= 4 SQUARE создает веса 1, 4, 9, 16 вместо весов 1, 2, 3, 4, что ускоренно поднимает цену более длинных цепочек совпадений. Слово SQUARE можно употребить и после CHMAXW=NOLIM.

Вы можете использовать п/к SAVEMXAW (см.) для сохранения максимального наблюдаемого, после пометки (награждения) цепочек, веса в каждой паре сравнененных документов.

*Награждение цепочек совпадений в условиях заданной TRV или TRSM*. Когда вы не задали TRV/TRSM, начальная матрица совстреч является двоичной: вес 1 = совстреча (совпадение), вес 0 = несовпадение. Эффект награды (пометки) цепочек совпадения подкомандой CHMAXW (когда она установлена не на 1) есть дифференциация веса 1: на месте весов 1 появятся веса 1, 2, 3,… Когда же вы задали TRV/TRSM, начальная матрица совстреч скорей всего не будет двоичной, а будет состоять из разных дробных значений от 0 до 1. Чтобы подкоманда CHMAXW могла действовать, т.е. наградить цепочки из единиц, необходимо сначала дихотомизовать дробные значения матрицы в двоичные: 1 и 0. Этим заведует подкоманда CHPASS (см.).

ПРИМЕР 5.

set rng= mc seed= 4056748. /\*You may set another seed

input program.

string seq (a8).

leave seq.

loop #seq= 1 to 10. /\*Let there be, say, 10 sequences (documents)

compute seq= concat('seq',ltrim(string(#seq,f3))).

loop #case= 1 to rnd(rv.uniform(50,100)). /\*Each containing, say, 50 to 100 words (cases)

compute word= rnd(rv.uniform(1,8)). /\*Let a "word" be an integer from, say, 1-8 range

end case.

end loop.

end loop.

end file.

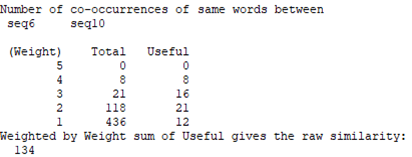
end input program.

execute.

dataset name data.

* Порождены некоторые данные.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 5 /method= GREEDY PRINT.



* Последовательности сравниваются попарно и выдается матрица сходств.
* Исследователя интересует не только словеснык состав, но и порядок смежных слов, поскольку CHMAXW задано и не равно 1. CHMAXW=5 награждает совпадения длиной 5+ смежных слов весом 5, совпадения длиной 4 смежных слов весом 4, совпадения длиной 3 смежных слов весом 3, и т.д.
* METHOD=GREEDY, делая спаривание, подсчитывает совпадения, начиная с более весомых. Т.к. возможные веса в данном примере целочисленны (1, 2, 3, 4, 5), вы можете использовать кл. слово PRINT для отчета о весах совпадений.
* В данном случае видим, что, например, между документами seq6 и seq10 нет ни одной общей цепочки длиной 5, есть две общие цепочки длиной 4 (поэтому сумма весов этих совстреч 8), есть 7 общих цепочек длиной 3 (поэтому сумма весов 21), и так далее. При спаривании были учтены (т.е. составили сходство) все 8 самых весомых совстреч, 16 совстреч весом 3, 21 совстреч весом 2, и т.д.

!KO\_seqsim docvar= seq /wordvar= word /trv= AUTO /chpass= .1 /chmaxw= 5 /method= GREEDY.

* В этом пуске, в отличие от предыдущего, у слов (терминов) неодинаковая релевантность (важность в деле сравнения документов): задано TRV и CHPASS. Величина сходства между всеми документами снизится, потому что релевантности окажутся меньше 1 (в данном примере они все одинаковы и равны 0.301, хотя обычно релевантности интересны, когда они разные).
* Веса совстреч (из-за п/к TRV) – дробные, поэтому кл. слово PRINT убрано.

**CHMINW**

Эта подкоманда (“chain minimal weight”) назначает минимальный наградительный вес совстречам, образующим сплошные подпоследовательности (цепочки смежных слов), совпадающие у двух сравниваемых последовательностей (документов). Укажите целое число не меньше 1. Эффект подкоманды следующий. *После* того, как CHMAXW наградит повышенным весом цепочки совпадений, CHMINW сотрет (занулит) все совстречи, вес которых оказался меньше, чем значение CHMINW. Добавление кл. слова INI после числа заставит подкоманду не стереть все такие совстречи, а оставить им начальный (до-наградительный) вес, т.е. 1. Таким образом, INI не выводит из обращения короткие цепочки, а удерживает за ними тот же вес, что у совстреч вне цепочек.

Например, CHMINW=4 уничтожит все совстречи, образующие цепочки совпадения длиной 3, 2 и 1 (1 это значит совстречи, не образующие цепочек). Но CHMINW= 4 INI придаст всем этим совстречам вес 1 (т.е. вернет им их исходный вес). CHMINW=2 уничтожит все совстречи с весом 1, т.е. не образующие цепочек. Но CHMINW= 2 INI сбережет их, т.к. вернет им их исходный вес 1 (они его и не меняли). Заметьте, что CHMINW= 2 INI эквивалентно по результату CHMINW=1.

По умолчанию, CHMINW=1. Это значит, что отдельные одинаковые в двух документах слова – учитываются при установлении сходства документов. Поднимите CHMINW выше единицы, если вас интересует сходство только за счет цепочек длиной не меньше столько-то, а не за счет отдельных слов. Поднимите CHMINW выше единицы и добавьте кл. слово INI, если вас интересует сходство преимущественно за счет цепочек длиной не меньше столько-то.

Значение в CHMINW указывается в линейной шкале, как в CHMAXW. Если в CHMAXW добавлено кл. слово SQUARE, оно автоматически распространяется на параметр CHMINW. Так, CHMINW=3 будет в этом случае пониматься как минимальный вес 9.

Когда CHMAXW=NOLIM, а заданное вами значение CHMINW оказалось больше длины длиннейшей цепочки, существующей в данной паре документов, то все цепочки окажутся занулены (если вы не добавите кл. слово INI) и сходство будет 0.

Когда CHMAXW=*число*, то CHMINW не может быть задано вами больше, чем это число CHMAXW. Если длиннейшая цепочка, существующая в данной паре документов, короче не только чем CHMAXW, но и короче чем CHMINW, то все цепочки окажутся занулены (если вы не добавите кл. слово INI) и сходство будет 0.

ПРИМЕР 6. Используя данные ПРИМЕРА 5.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 5 /chminw= 3 INI /method= GREEDY.

* В отличие от 1-го пуска в ПРИМЕРЕ 5, тут добавлено CHMINW= 3 INI. Это значит, что макрос наградит совпадения длиной 5+ смежных слов весом 5, совпадения длиной 4 смежных слов весом 4, совпадения длиной 3 смежных слов весом 3, а вот совпадения длиной 2 он наградит только весом 1, как и совпадение единичные (длиной 1).

**CHBWARD**

Эта подкоманда (“backward chains”) играет роль, если CHMAXW не 1. По умолчанию/незаданию и при CHBWARD=NO цепочки совпадений только с одинаковым, прямым порядком слов награждаются повышенным весом. При CHBWARD=YES цепочки и с обратным порядком тоже считаются совпадающими и награждаются повышенным весом. Например, череда слов ‘A B C’ в документе посчитается цепочкой совпадения не только с такой же чередой слов в другом документе, но и с чередой ‘C B A’ в нем. Иначе говоря, подкоманда CHBWARD заставляет считать зеркально-идентичные подпоследовательности тоже одинаковыми, как и со-идентичные.

**CHPASS**

Эта подкоманда (“pass score to chains”) действует и необходима, если вы задали TRV или TRSM, в то время как CHMAXW не 1 – т.е. когда должно быть награждение общих цепочек. Когда вы не задали TRV/TRSM, начальная матрица совстреч является двоичной: вес 1 = совстреча (совпадение), вес 0 = несовпадение. Эффект награждения цепочек совпадения подкомандой CHMAXW есть дифференциация веса 1: на месте весов 1 появятся веса 1, 2, 3,… Когда же вы задали TRV/TRSM, то начальная матрица совстреч скорей всего не будет двоичной, а будет состоять из разных дробных значений от 0 до 1. Чтобы подкоманда CHMAXW могла действовать, т.е. наградить цепочки из единиц, необходимо сначала дихотомизовать дробные значения матрицы в двоичные: 1 и 0. Этим заведует подкоманда CHPASS, назначающая проходной балл для участия в пометке цепочек.

Если задана п/к TRV, то начальная матрица совстреч содержит, кроме нулей, релевантности – они выступают в качестве весов совстреч терминов с самими собой. Укажите в CHPASS одно значение в диапазоне (0,1]. Совстречи терминов с собой, имеющие вес не ниже этого значения, будут обращены в 1, а имеющие вес ниже – обращены в 0. После такой дихотомизации матрица подается на награждение цепочек.

Если задана п/к TRSM, то начальная матрица совстреч содержит, кроме нулей, релевантности – они выступают в качестве весов совстреч терминов с самими собой, и содержит также схожести – они выступают в качестве весов совстреч между некоторыми разными терминами. Укажите в CHPASS два значения в диапазоне (0,1]. Первое значение касается совстреч терминов с самими собой. Такие совстречи, имеющие вес не ниже этого значения, будут обращены в 1, а имеющие вес ниже – обращены в 0. Второе значение касается совстреч между разными терминами. Такие совстречи, имеющие вес не ниже этого значения, будут обращены в 1, а имеющие вес ниже – обращены в 0. После такой дихотомизации матрица подается на награждение цепочек. Таким образом, схожесть, обращенная в 1, может стать звеном цепочки совпадения. Так, ‘A B C D’ в документе и ‘A B Y D’ в другом документе сочтутся совпадающими цепочками, если ‘C’ и ‘Y’ имеют между собой схожесть ≥ проходной балл для схожестей.

Если вы задали TRV и два значения в CHPASS, второе значение молчаливо игнорируется.

После того, как награждение (пометка) цепочек сделано (т.е. сыграют все заданные из подкоманд CHMAXW, CHMINW, CHBWARD), CHPASS «отыграет назад» сделанную ею дихотомизацию, просто умножив поэлементно полученную целочисленную матрицу совстреч на начальную матрицу совстреч. Например, некоторая совстреча, элемент матрицы совстреч, имела сначала вес 0.6. Это значение оказалось выше пропускного балла, заданного в CHPASS, значит, оно обращается в 1 (а не в 0). Далее пометка цепочек выяснила, что эта совстреча принадлежит цепочке совпадения такой длины, что ее элементы достойны награды весом 3. В итоге: 3∙0.6=1.8 – такой вес будет иметься у данной совстречи по выходе из блока «награждение подпоследовательностей». А если начальный вес совстречи, допустим, был 0.2 и оказался ниже пропускного балла, то он обращается в 0 и не участвует в награждении цепочек. Т.к. 0∙0.2=0, такая совстреча отныне не сможет подать голос за сходство двух текущих документов.

Подчеркнем: пропускной режим CHPASS действует, только если CHMAXW задана и не есть 1.

ПРИМЕР 7. Используя данные ПРИМЕРА 5. Вам потребуется еще подготовить файл с релевантностями *RELEV.SAV*, файл с релевантностями/схожестями *TRSM.SAV*.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 5 /trv= 'd:\exercise\relev.sav' /chpass= .1

/method= GREEDY.

* Во 2-м пуске ПРИМЕРА 5 использовалась TRV=AUTO. В нынешнем примере релевантности заданы как внешний файл. А т.к. CHMAXW не равно 1, необходима CHPASS. Всякое слово с релевантностью от 0.1 и выше будет участвовать в пометке цепочек подкомандой CHMAXW. После пометки полученные целочисленные веса будут умножены на соответствующие исходные веса, взятые из TRV-файла.
* Подкоманда TROTR (сопровождающая TRV или TRSM) опущена. По ее умолчанию, релевантность для всякого неупомянутого в файле термина =1.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 5 /trsm= 'd:\exercise\trsm.sav' /chpass= .1 .8

/method= GREEDY /dataset= data.

* В этом пуске, в отличие от предыдущего, мы задаем не только релевантности, но и схожести слов. CHPASS требует два параметра – первый для релевантностей, второй для схожестей. Термины, между которыми степень синонимии от 0.8 и выше, будет участвовать в пометке цепочек подкомандой CHMAXW: они будут рассматриваться на этом подэтапе как полные синонимы. После пометки полученные целочисленные веса будут умножены на соответствующие исходные веса, взятые из TRSM-файла.
* Подкоманда TROTR (сопровождающая TRV или TRSM) опущена. По ее умолчанию, релевантность для всякого неупомянутого в файле термина =1 (а схожести для неупомянутых терминов всегда =0).

**DIAG**

Независимо от того, важны вам в определении сходства документов последовательности смежных слов или важен только словесный состав, вас может волновать, одинаковое или разное положение в документах занимают фрагменты (цепочки, отдельные слова), между ними совпадающие. Иначе говоря, близка или далека матрица совстреч от диагонального паттерна.

Опциональная подкоманда DIAG существует для учета степени диагональности матрицы совстреч. Матрица совстреч максимально близка к диагональности, когда документы близки к тому, чтобы быть копиями друг друга. Подкоманда DIAG накажет снижением веса совстречи в тех рядах и столбцах матрицы, где размещение совстреч нарушает диагональный идеал. Это наказание понижает сходство между документами. См. «Алгоритм: Проверка диагональности».

Вы можете захотеть применить DIAG в случае, если для вас документы сходны только тогда, когда одинаковые у них слова или подпоследовательности («фразы»), занимают примерно одинаковые позиции в них. Т.е., например, ‘A B C D X Y Z’ и ‘X Y Z A B C D’ вас уже не устроит, потому что ‘A B C D’ и ‘X Y Z’, хотя и общие у двух последовательностей, занимают разные позиции, их размещение разное. В отличие от подкоманды CHMAXW, п/к DIAG не занимается порядком именно *смежных* слов, но рассматривает со-упорядоченность на «верхнем уровне» последовательностей.

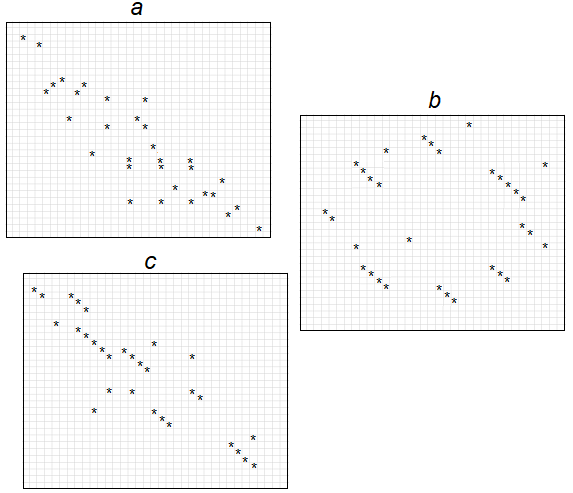
Задайте DIAG= MAIN, MIDDLE или BAND. Разница между этими вариантами проявляется, когда матрица совстреч неквадратна, т.е. документы имеют разную длину. MAIN принимает за диагональ отсчета только главную диагональ, MIDDLE – срединную диагональ, а BAND – всю полосу диагоналей. См. иллюстрацию в «Алгоритм: Проверка диагональности».

Задание DIAG несовместимо с CHBWARD=YES.

ПРИМЕР 8.

На **Fig. 17** три разные матрицы совстреч, каждая между некими двумя документами. Точками показаны совстречи одинаковых слов между одним и другим документом. (Данные, соответствующие этим рисункам, вы найдете в «Приложении».) В случае (**a**) паттерн довольно диагональный, но цепочки совпадения почти отсутствуют. Мы можем сказать, что документы близки к тому, чтобы быть «расслабленными», без фиксации порядка смежных слов, копиями друг друга. В случае (**b**) напротив, мы видим много цепочек совпадения, т.е. одинаковых «фраз», однако они занимают разные положения в двух документах, отчего паттерн очень недиагональный. В случае (**c**) в наличии как диагональность, так и цепочки смежных слов. Тут два документа – почти копии друг друга на обоих уровнях: на «макроуровне» расположения частей, и на «микроуровне» соприкасающихся слов.

**Fig. 17**



!KO\_seqsim docvar= seq /wordvar= word /diag= MIDDLE /method= GREEDY.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 4 /method= GREEDY.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 4 /diag= MIDDLE /method= GREEDY.

* Первый пуск «накажет» снижением сходства случай (b) – из-за наличия п/к DIAG; случаи (a) и (c) покажут более высокое и примерно одинаковое сходство.
* Второй пуск даст довольно высокое и примерно одинаковое сходство в случаях (b) и (c), и существенно меньшее сходство в случае (a).
* Третий пуск совмещает CHMAXW и DIAG, и даст высокое сходство в случае (c), оба других случая будут «наказаны», каждый по своей причине. В данном примере случай (b) покажет сходство, грубо, посредине между (c) и (b). Вы можете усилить (или ослабить) наказание за недиагональность, добавив подкоманду DIAGLS, являющуюся настройкой. Так, если вы зададите /DIAGLS= 0.35 1 (тогда как дефолт соответствует /DIAGLS= 0.5 1), случай (b) будет наказан сильней и покажет сходство примерно на уровне случая (a).

**DIAGLS**

Эта опциональная подкоманда обслуживает п/к DIAG и существует для тонкой настройки проверки диагональности. По умолчанию, зависимость «наказания» за величину отклоненности от диагональной позиции – линейная. Вы можете сделать ее криволинейной, усилить или ослабить ее «резкость». Укажите два числа: первое это параметр положения – число в диапазоне (0,1); второе это параметр формы, или крутизны – число ≥ 1. См. «Алгоритм: Проверка диагональности».

**RESCRND**

Эта опциональная подкоманда действует, если задано DIAG или TRV/TRSM. В указанных случаях значения в матрице совстреч подходят к основному этапу – этапу начисления сходства – как правило в дробном виде, и разнообразие значений в матрице может быть велико. Когда разнообразие велико, т.е. данные довольно континуальны, то на процедуру максиматьного паросочетания (METHOD= GREEDY или HUNGAR) уходит больше времени. RESCRND дискретизует данные способом перешкалирования-округления, это ускорит процедуру паросочетания. См. «Алгоритм: Округление веса совстречи». Разумеется, дискретизация несколько огрубляет, делает менее точным вычисление итогового сходства, но в большинстве случаев это терпимая потеря, в отличие от медленной работы макроса.

Укажите число 1, 2 или 3. Это есть точность, число оставляемых десятичных цифр, при округлении после перешкалирования в диапазон 0-1. RESCRND=1 это самый грубый и быстрый вариант. Незадание/умолчание подкоманды RESCRND – это самый медленный и точный вариант. RESCRND=2 – часто вариант выбора, если скорость вам важна: возможно ускорение в полтора-два раза (при METHOD=HUNGAR) и четыре-пять раз (при METHOD=GREEDY), с терпимой потерей точности.

Методы другие, чем GREEDY или HUNGAR, никогда не нуждаются в применении п/к RESCRND, т.к. их скорость не зависит от разнообразия весов в матрице. Однако результаты, конечно, все равно зависят от того, применили вы RESCRND или нет.

Используйте PRINT=MEDIUM (см.), чтобы знать, как RESCRND уменьшило разнообразие весов совстреч.

ПРИМЕР 9. Используя данные ПРИМЕРА 5. Вам потребуется еще подготовить файл с релевантностями *RELEV.SAV*, файл с релевантностями/схожестями *TRSM.SAV*.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 5 /trsm= 'd:\exercise\trsm.sav' /diag= MAIN

/rescrnd= 2 /method= GREEDY /dataset= data.

* Без RESCRND задание в этом пуске будет неблагоприятно для быстродействия, особенно если документов много или они длинные. Т.к. присутствуют CHMAXW, TRSM, DIAG. Каждая из этих подкоманд разнообразит веса в матрице совстреч, тем более – все три подкоманды вместе. RESCRND=2 является снадобьем в такой ситуации. Не большая потеря точности вычислений компенсируется хорошей экономией времени.
* RESCRND уменьшает дробящий эффект DIAG и TRSM (или TRV). В отсутствие этих подкоманд RESCRND игнорируется.

**METHOD**

Метод получения величины (сырого) сходства двух документов – из матрицы совстреч, прошедшей заказанную вами предваряющую обработку. Предлагается два варианта метода максимального паросочетания, метод максимального выравнивания (в базовой форме), метод максимальной общей цепочки, и некоторые простейшие методы. Подробнее – см. «Алгоритм: Начисление сырого сходства». См. тж. **Fig. 2**.

GREEDY - простой жадный алгоритм максимального паросочетания. Он пошаговый и (если веса у элементов, т.е. совстреч, разные) начинает спаривание за счет более весомых совстреч. Он всегда выберет спарить более весомый элемент. Этот метод преследует, но не гарантирует достижение глобального максимума паросочетания, т.е. выявления величины сырого сходства до ее выявляемого предела.

HUNGAR - венгерский (Kuhn–Munkres) алгоритм максимального паросочетания. Он итеративный и максимизирует паросочетание до глобального максимума, т.е. выявляет величину сырого сходства до ее выявляемого предела. Он не всегда выберет спарить более весомый элемент менее весомому элементу. Это более медленный метод.

MXTRACE - максимально-выравнивающий (aligning) подход. В матрице совстреч вычисляется сумма в каждой ее диагонали (индивидуальная диагональ начинается с каждой позиции 1-го ряда либо с каждой позиции 1-го столбца). Наибольшая такая сумма и есть сырое сходство. В отличие от паросочетания, тут нет цели включить, по возможности, в сходство как можно больше совстреч (ненулевых весов), содержащихся в матрице. А в отличие от метода MXCHW, тут могут суммироваться веса более чем одной цепочки.

MXCHW - максимальная общая цепочка: величина сходства это вес одной, самой весомой среди длинных общих цепочек. Если п/к TRV/TRSM/DIAG не употреблялись, то это есть вес, прямо отражающий длину длиннейшей общей цепочки. Если что-то из п/к TRV/TRSM/DIAG употреблялось, а CHMAXW=*число*, то максимально весомой может оказаться и не длиннейшей цепочка. В отличие от всех предыдущих методов, в методе MXCHW в величину сходства суммируются веса совстреч, составляющих только одну цепочку. Подробнее – см. «Алгоритм: Подход максимальной общей цепочки». Метод MXCHW разрешен только если вы заказали взвешивание цепочек совпадения, т.е. задали CHMAXW не как 1. MXCHW несовместим с CHBWARD=YES.

MXW - величина сходства это максимальный вес в матрице совстреч.

MEANPW - величина сходства это усредненный ненулевой (положительный) вес в матрице совстреч.

SUM - величина сходства это сумма всех весов в матрице совстреч.

Из двух максимально-паросочетательных методов метод GREEDY можно рассматривать как на практике предпочтительный. Во-первых, он гораздо быстрее, чем венгерский. Во-вторых, в условиях разреженной матрицы элементов совстреч (что как правило имеет место) локальный оптимум, достигаемый жадным подходом, часто отличается мало или даже совпадает с глобальным оптимумом венгерского подхода. В-третьих, форсируемое жадным методом предпочтение более весомого элемента менее весомому может быть для пользователя понятнее и важнее, нежели интерес венгерского алгоритма «исчерпать в итоге» наличное сходство до своего алгебраического предела.

После METHOD=GREEDY возможны добавочные ключевые слова REORD и/или PRINT.

Кл. слово REORD меняет перед паросочетанием порядок рядов/столбцов в матрице совстреч так, чтобы увеличить шансы жадного алгоритма приблизиться к глобальному максимуму паросочетания: METHOD= GREEDY REORD максимизирует паросочетание обычно сильнее (ближе к HUNGAR) чем METHOD=GREEDY. Это преимущество максимально тогда, когда разнообразие весов в матрице совстреч мало, что реализуется, когда параметр CHMAXW низок (но выше 1) и когда многообразие слов во входящих документах не очень мало. Сходство от GREEDY REORD может тогда быть приблизительно на 2-3% выше, чем от GREEDY. Что касается CHMAXW=1 в отсутствие TRV/TRSM и DIAG, то матрица тогда дихотомична: есть только один вес у совстреч; в этих условиях REORD не дает никакого преимущества. На REORD-переупорядочивание рядов/столбцов, конечно, тратится время.

Кл. слово PRINT (см. ПРИМЕР 5) распечатывает в окно результатов сводку о количестве совстреч с разными весами – перед жадным паросочетанием (графа “Total”) и после него (графа “Useful”). Вычисленное сырое сходство есть взвешенная весами сумма в “Useful”. См. «Алгоритм: Подход максимального паросочетания». Кл. слово PRINT не разрешено в следующих случаях: задано DIAG, задано TRV/TRSM, CHMAXW=NOLIM, CHMINW содержит INI.

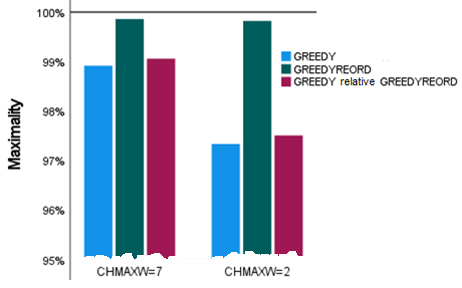
ПРИМЕР 10. Используя данные ПРИМЕРА 5.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 7 /method= GREEDY /dlen= NONE /print= MEDIUM.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 7 /method= GREEDY REORD /dlen= NONE /print= MEDIUM.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 7 /method= HUNGAR /dlen= NONE /print= MEDIUM.

* PRINT= MEDIUM информирует о разнообразии весов в матрице совстреч перед начислением сходства. От всех трех пусков макроса эта информация одинакова. DLEN=NONE заказывает выдать сырое сходство, какое оно выдано методом METHOD.
* GREEDY REORD имеет тенденцию давать иногда сходство выше, чем GREEDY, хотя порой и ниже, чем HUNGAR. Впрочем, различия обычно не превосходят 1-2 процентов.
* Если вы зададите CHMAXW меньше (но не 1) – например 2, отрыв GREEDY REORD от GREEDY будет более выраженным (**Fig. 18**).



**Fig. 18**. Оптимальность для рассмотренного примера. Максимально возможное при паросочетании начисленное сходство, 100%, соответствует результату HUNGAR.

ПРИМЕР 11. Метод максимального выравнивания.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 7 /method= MXTRACE /dlen= NONE.

* Этот пуск использует подход выравнивания, а не паросочетания. Величина сходства будет естественно ниже, чем в ПРИМЕРЕ 10, поскольку выравнивание не старается максимизировать количество совстреч, учтенных в суммировании сходства.

ПРИМЕР 12. Метод максимальной общей цепочки.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= NOLIM /method= MXCHW /dlen= NONE.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 3 /method= MXCHW /dlen= NONE.

* Эти два пуска эквивалентны. **Tab. 1** в разделе «Алгоритм» проясняет, почему. Допустим, что длиннейшая существующая в некоторой паре документов X и Y общая цепочка имеет длину 6. Тогда CHMAXW=NOLIM пометит эту цепочку весом 6. Сырое сходство, являясь суммой весов в длиннейшей цепочке, будет равно, следовательно, 6∙6=62. Со своей стороны, CHMAXW=3 пометит ту же цепочку весом 3 (предельной разрешенной пользователем наградой). Следовательно, сырое сходство будет равно 6∙3. Эти результаты эквивалентны, потому что мы знаем, как сделать их одинаковыми: в первом случае взять корень из сырого сходства, а во втором случае поделить сырое сходство на 3.
* И действительно, если вы затребуете нормирование под DIVISOR1: вместо /DLEN=NONE напишете /DIVISOR=DIVISOR1, то нормированное сходство получится в обоих случаях одинаково. В ходе нормирования происходит деление на параметр DIVISOR. В первом случае этот параметр (DIVISOR1 это *wmax*, см. формулу нормирования в разделе «Алгоритм») в паре X и Y будет равен 6, а во втором случае – равен 3.
* Если не употреблять п/к DIAG/TRV/TRSM, то METHOD=MXCHW есть метод длиннейшей общей цепочки.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= NOLIM /diag= MAIN /method= MXCHW /dlen= NONE.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 3 /diag= MAIN /method= MXCHW /dlen= NONE.

* Эти два пуска отличаются от двух предыдущих использованием п/к DIAG. Подкоманды DIAG, TRV, TRSM – это подкоманды, которые изменяют веса совстреч относительно тех целочисленных весов, которыми помечаются общие цепочки на подэтапе награждения цепочек. В результате этого изменения имеем то, что:
* (i) если цепочек максимальной длины существует несколько, они могут различаться весами: выиграет (выразит собой сходство) самая весомая из них; (ii) при CHMAXW=*число* «длинными» могут засчитаться и более короткие цепочки, чем длиннейшие цепочки (см. **Tab. 1**), и выиграть может одна из тех более коротких – если она окажется самой весомой из «длинных».
* Вот почему мы говорим в общем случае – т.е. в проспекте п/к DIAG/TRV/TRSM – о методе «максимальной общей цепочке», а не о методе «длиннейшей общей цепочке».

**DLEN**

Эта подкоманда делает опциональное нормирование сырого сходства *S*(D1, D2), полученного подкомандой METHOD, в нормированное сходство *Sim*(D1, D2), коэффициент сходства. Данная опция касается METHOD= GREEDY, HUNGAR, MXTRACE и MXCHW, т.е. методов, где сырое сходство возникает от суммирования «полезных совстреч», и слово участвует не более одного раза в суммировании. Для других методов – MEANPW, MXW, SUM – макрос выдает только сырое сходство *S*(D1, D2) и принуждает поэтому задать DLEN=NONE.

DLEN – нормирующая длина документа, она входит в знаменатель формулы коэффициента сходства *Sim*(D1, D2) между двумя документами (см. «Алгоритм: Нормирование в коэффициент сходства»). Коэффициент может теоретически достичь верхнего предела 1, только если DLEN=MIN или если два документа одинаковой длины. Длина документа |D| это число слов в нем (разрывы не учитываются). Параметр DLEN выбирается пользователем исходя из своих задач. Вы можете также оставить сырое, ненормированное сходство. Укажите одно из следующего:

MIN - (тж. по умолчанию/незаданию) длина более короткого из двух документов: min(|D1|, |D2|)

MAX - длина более длинного из двух документов: max(|D1|, |D2|)

MEAN - среднеарифметическая длина двух документов: (|D1|+|D2|)/2

GMEAN - среднегеометрическая длина двух документов: sqrt(|D1|∙|D2|)

HMEAN - среднегармоническая длина двух документов: 2∙|D1|∙|D2|/(|D1|+|D2|)

NONE - не нормировать, выдать числитель (сырое сходство) *S*(D1, D2) как есть.

При DLEN=NONE на диагонали квадратной матрицы сходств будет условное значение 999, а в прочих случаях – единица.

Если вы хотите сохранить, как файл, длины документов, |D|, используйте п/к SAVEMXAW (см.) в условиях COMPARE=ALLPAIRS.

**DIVISOR**

Эта подкоманда допускается, только если DLEN не NONE. И она игнорируется при незаданном CHMAXW или CHMAXW=1, т.е. когда не заказана пометка цепочек. Делитель *DIVISOR* входит в формулу нормирования сырого сходства в коэффициент сходства. Его задача – устранить из числителя (т.е. сырого сходства) инфляцию весов, связанную с награждением цепочек, для того чтобы верхний предел коэффициента сходства не превышал 1. Подробнее ­– см. «Алгоритм: Нормирование в коэффициент сходства: Выбор значения DIVISOR».

DIVISOR1 - «решение, забывающее взвешивание цепочек».

DIVISOR2 - «суровое решение, утверждающее взвешивание цепочек».

DIVISOR3 - (тж. по умолчанию/незаданию) «справедливое решение, утверждающее взвешивание цепочек».

DIVISOR4 - «компромиссное решение, утверждающее взвешивание цепочек».

Если документов в анализе только два (и п/к WMAX не задана), DIVISOR1 = DIVISOR2 = DIVISOR3.

**WMAX**

Эта опциональная, дополнительная к п/к DIVISOR подкоманда позволяет вам задать свое значение для *Wmax* вместо определяемого макросом. Используйте, если вы хотите, чтобы *Wmax* не зависело от текущих данных. *Wmax* употребляется при вариантах DIVISOR= DIVISOR2, DIVISOR3 и DIVISOR4. Подробнее ­– см. «Алгоритм: Нормирование в коэффициент сходства: Выбор значения DIVISOR». Укажите целое положительное число.

**SAVEM1**, **SAVEM2**

Эти опциональные подкоманды действуют, если входящих документов ровно два. Подкоманды сохраняют матрицу совстреч, в которой ряды – это последовательность элементов 1-го документа (кроме 1-го ряда, который показывает слова 2-го документа), и столбцы – это последовательность элементов 2-го документа (и 1-й столбец, *SEQ1*, показывает слова 1-го документа). Значения матрицы – веса совстреч.

Таким образом, 1-й столбец и 1-й ряд – это сами последовательности (документы). Они приведены в кодировке переменной *WORD\_.#$*, т.е. состоят из целых положительных кодов. Отрицательные значения (это случайные отрицательные числа) соответствуют разрывам между словами, если таковые были.

SAVEM1 сохранит начальную (полученную в этапе 1 алгоритма) матрицу совстреч. SAVEM2 сохранит матрицу совстреч, какова она на входе в этап 3 (см. «Алгоритм: Начисление сырого сходства», см. также **Fig. 16**).

Задайте одну или обе подкоманды. В каждой укажите свой внешний .SAV-файл (путь/имя в кавычках или апострофах) или имя заявленного массива (можно в кавычках/апострофах), для сохранения[[8]](#footnote-8).

П/к SAVEM2 несовместима с COMPARE=FIRSTEXT.

ПРИМЕР 13.

dataset declare mat2.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= NOLIM /savem2= mat2 /method= GREEDY.

* Матрица совстреч, на базе которой происходит начисление сходства, сохранена как новый массив *MAT2*.

**SAVEMXAW**

Эта опциональная подкоманда (“save maximal awarded weight”) сохраняет, для каждого сделанного сравнения, наибольший в матрице совстреч вес (значение), наблюдаемый после создания матрицы совстреч и опциональной пометки цепочек – то есть после этапа 1 или после подэтапа 2a, если он был (см. «Алгоритм»). Конкретно:

* Если CHMAXW=1 (= по умолчанию CHMAXW), т.е. пометки, награждения цепочек не было, то сохраняемый максимальный вес это вес из начальной матрицы совстреч. (Он может быть двоичным или дробным, в зависимости от того, задана ли п/к TRV/TRSM.)
* Если CHMAXW не 1, т.е. пометка, награждение цепочек было, то сохраняемый максимальный вес это вес из матрицы совстреч после награждения цепочек, т.е. действий подкоманд CHMAXW и CHMINW. Это целочисленный вес. Причем, если у вас задана п/к TRV/TRSM, то сохраняемый вес – это вес перед умножением его на исходный, дробный вес (см. «Алгоритм: Взвешивание длиной цепочки в условиях задания подкоманд TRV или TRSM»; см. также **Fig. 16**). В «Алгоритме» он упоминается как *wmax*.

Укажите внешний .SAV-файл (путь/имя в кавычках или апострофах) или имя заявленного массива (можно в кавычках/апострофах), для сохранения. Если COMPARE=ALLPAIRS, сохранится квадратная симметричная матрица, а на диагонали ее будут стоять длины документов (без учета слов-разрывов). Если COMPARE=FIRST, сохранится вектор.

П/к SAVEMXAW несовместима с COMPARE=FIRSTEXT.

ПРИМЕР 14.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= NOLIM /savemxaw= 'd:\exercise\mxw.sav' /method= GREEDY.

!KO\_seqsim docvar= seq /wordvar= word /chmaxw= 5 /diag= MAIN /dlen= MEAN

/savemxaw= 'd:\exercise\mxw2.sav' /method= GREEDY.

* В первом, предварительном, пуске CHMAXW=NOLIM, поэтому матрица, сохраняемая SAVEMXAW, будет содержать длину длиннейшей существующей цепочки совпадения. Открыв и осмотрев файл *MXW.SAV*, исследователь нашел, что эти значения в его данных лежат в районе 5-7. Поэтому он решил установить во втором, окончательном пуске CHMAXW на 5. Общим цепочкам длиной 5+ будет присуждаться максимальный вес 5.
* Во втором пуске исследователь ввел некоторые опции, какие планировал. Между прочим, значения в *MXW2.SAV* уже не будут выше 5.

**PRINT**

Опциональная PRINT печатает в окне результатов некоторые сведения, включая отчет об исходной автоперекодировке.

SHORT - вычисленное сырое сходство *S*(D1, D2), а также сообщение, если матрица стала нулевой (пустой) и когда. (Матрица могла быть пустой изначально, стать пустой в ходе пометки цепочек.)

MEDIUM - кроме этого, информация о плотности (% к числу ячей) и разнообразии ненулевых значений, т.е. весов, в матрице совстреч перед начислением сходства. Если задано RESCRND, информация показывается дважды, до округления и после округления перед начислением сходства, что может быть полезно для настройки макроса (см. **Fig. 16**).

LONG - кроме этого, сама матрица совстреч (если задано RESCRND, то дважды). Когда заданы DIAG или TRV/TRSM, т.е. значения в ней могут быть дробными, они показываеются с 4-мя десятичными цифрами. Нет смысла использовать LONG, если документы большие: распечатка тогда малоинформативная и, кроме того, замедляет работу макроса.

**DATASET**

Эта подкоманда работает и требуется при COMPARE=FIRSTEXT или если задано TRSM, и носит подсобный, технический характер. Укажите имя входящего, рабочего массива. Или задайте DATASET=RENAME, тогда макрос просто переименует ваш входящий массив в *KO\_SEQSIM#$.\_*.

**NEWDATA**

Необязательная подкоманда для придания имени выходящему массиву со сходствами (или с длиннейшими цепочками, при COMPARE=FIRSTEXT). Задайте имя. По умолчанию/незаданию массив останется безымянным.

***Особые режимы***

Макрос игнорирует взвешивание и не рассчитан на расщепленное состояние массива данных. Он отключает взвешивание массива. Макрос слушается команд, выбирающих наблюдения (FILTER, USE, SELECT IF, N OF CASES). Он не слушается временных (стоящих под TEMPORARY) операций.

***Приложение***

Данные для **Fig. 17**.

\*Fig. 17a data.

data list free /word (a8).

begin data

A B C D E F G H I J K L M N O L P Q R S T T U V W X T Y Z AA BB CC

DD EE C GG D K J I O K J S HH L II JJ T O L R T KK W KK T MM X X V Z Y NN OO BB PP

end data.

string seq (a8).

do if $casenum<=32.

compute seq= 'seq1'.

else.

compute seq= 'seq2'.

end if.

execute.

dataset name fig17a.

\*Fig. 17b data.

data list free /word (a8).

begin data

A B C D E F G H I J K L M N O P M Q R H S T I J K F U V W X Y Z

AA BB CC O P DD EE H I J K F FF GG R HH D E F U V II B JJ KK I J K L M Q LL H MM NN

end data.

string seq (a8).

do if $casenum<=32.

compute seq= 'seq1'.

else.

compute seq= 'seq2'.

end if.

execute.

dataset name fig17b.

\*Fig. 17c data.

data list free /word (a8).

begin data

A B C D E F G H E F I J K L M N O K P Q I R S T U V W X U Y Z AA

BB C D CC H DD D E F I J K EE J K L M I R S FF GG K P HH II JJ V W X U KK LL MM NN

end data.

string seq (a8).

do if $casenum<=32.

compute seq= 'seq1'.

else.

compute seq= 'seq2'.

end if.

execute.

dataset name fig17c.

1. Не путать с методом *оптимального паросочетания последовательностей* на базе алгоритма Needleman–Wunsch. [↑](#footnote-ref-1)
2. Лишь в условиях подкоманды TRSM совстрече термина не с собой может уделяться внимание. [↑](#footnote-ref-2)
3. Формула функции Φ заимствована у Werner Antweiler (<https://wernerantweiler.ca/blog.php?item=2018-11-03>), который назвал ее sigmoid-logit distribution function. [↑](#footnote-ref-3)
4. или не этого – в условиях п/к TRSM с ненулевыми схожестями. [↑](#footnote-ref-4)
5. Это простейший метод сравнения последовательностей посредством выравнивания; существуют другие методы выравнивания, в частности, вставляющие разрывы и др. редакторские правки там и сям внутрь последовательности, но такие методы настоящий макрос не предоставляет. [↑](#footnote-ref-5)
6. Это тот вес *wmax*, который сохранит подкоманда SAVEMXAW. [↑](#footnote-ref-6)
7. Это тот вес *wmax*, который сохранит подкоманда SAVEMXAW. [↑](#footnote-ref-7)
8. Если желаете начертить рисунок сохраненной матрицы, тепловую карту (heatmap), используйте макрос !KO\_EDPROXMX (коллекция “Various proximities”). [↑](#footnote-ref-8)