Fuzzy-Temporal Gradual Patterns

The T-GRAANK Algorithm

Dickson Owuor[1]

[1] Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier (LIRMM)
Fuzziness, Alignments, Data & Ontologies (FADO)

27 février 2019

Summary

Global overview

Knowledge Discovery in Databases KDD is an area of research in **Computer Science** that deals with the techniques of finding knowledge in data-sets. The data-sets can be in the form of files (various formats) or in the form of data stored in database (various types).

Researchers have formulated different techniques for knowledge discovery in databases. We are interested in these two categories :

- Frequent pattern mining
- Gradual pattern mining

More specifically, our area of focus is gradual pattern mining and this presentation unveils work that extends gradual rules to include fuzzy temporal constraints.

Introduction

Frequent Patterns

- mines the frequent item-sets in a data-set through association rules
- main algorithm is APRIORI
- quality measures :
  1. support : occurrence count in the entire data-set
  2. confidence : probability that a transaction contains pattern
- For instance given the data-set : { {t001, bread, milk}, {t002, bread, eggs, milk}, {t003, cheese, milk,}, {t004, sugar, milk, coffee, bread}}. The pattern `bread implies milk` : *bread* $\rightarrow$ *milk* has a support of $\frac{3}{4}$ and confidence of 100%
- application areas : customer tendencies, image processing (related color bands), gene classification (cancerous) etc.

Introduction

Gradual Patterns

- mines the correlations between attributes through gradual rules such as “the more the A, the more the B”
- main algorithms are GRAANK and GRITE
- quality of patterns is also measured by support count
- For instance given the data-set : { {id, activity, stress-level}, {t001, 1, 4}, {t002, 2, 3}, {t003, 3, 2}, {t004, 5, 4}}. The pattern ‘‘`the more activity, the less stress`’’ : $\{activity \uparrow, stress \downarrow\}$ has a support of $\frac{3}{4}$ and a confidence of 75%
- application areas : customer tendencies, geospatial mapping etc.

NB : unlike frequent patterns, support for gradual patterns is determined by the subsequent count of tuples that respect the pattern and not by isolated occurrence counts.

Proposition

It is true that gradual rules allows for retrieval of correlations between the attributes of a data-set. However, we may further want to know the temporal difference between the correlations e.g. "`the more A, the more B` **almost 6 months later** denoted as : $\{(A, \uparrow), (B, \uparrow)_{+\simeq 6 months}\}$

In this paper we extend the **GRAANK approach :** `gradual ranking technique based on Kendall's` τ in order to allow gradual rules to include the fuzzy temporal tendencies in the correlations between attributes.

In order for us to achieve this we :

1. propose an algorithm for transforming the original data-set with respect to the date-time attribute
2. introduce formula that will enable a fuzzy triangular membership function to estimate the temporal gap between correlations.

Sample Dataset

id	**date** (day/month)	**exercise** (hours)	**stress levels**
r1	01/06	1	4
r2	04/06	2	2
r3	05/06	3	3
r4	10/06	1	2
r5	12/06	3	3

TABLE – A sample data-set $\mathcal{D}$

id	**days *lag*** $(r_n - r_{n+1})$	**exercise** (r_n)	**stress** (r_{n+1})
t1	3	1	2
t2	1	2	3
t3	5	3	2
t4	2	1	3
t5	-	-	-

TABLE – Transformed data-set $\mathcal{D}'$: transformation : r_{n+1}

the gradual pattern $\{(exercise, \uparrow), (stress, \downarrow)\}$ for trans. r_{n+1} using the tuples <t2, t3>, has a support of $\frac{2}{4}$.

Take note of :

- representativity
- time-lag estimation

Time-lag Estimation

From the previous transformation r_{n+1}, we have the following time-lag members : $\{3, 1, 5, 2\}$. Our interest is to choose a fuzzy membership that will include the greatest portion of these members. Therefore we propose two main definitions :

Definition a. *Time Lag. A time lag is the amount of time that elapses before or after the changes in a one gradual item affects the changes in another gradual item. Time lag is denoted as $\alpha\beta t$ where α is an operator $\alpha \in \{+, -\}$ and '+' implies after and '−' implies before ; β is an operator $\beta \in \{=, \simeq\}$ and '=' implies equal to and '$\simeq$' implies almost ; t is the value of time lag* - given by formula in Definition b.

Definition b. *$t = Medial_{M \in m}$, where M is a medial of sequence m, and $m = (c_i r_1 - c_i r_{1+k}), ..., (c_i r_n - c_i r_{n+k})$ where c is the column for time/date, r is a single tuple/row, $i = 1, 2, ..., I$, $n = 1, 2, ..., N$, and $k = 1, 2, ..., K$.*

Fuzzy Logic

Fuzzy logic - a concept that was first introduced by Lofti Zadeh in the 1960s and it allows computing to be based on **"degrees of truth"** rather than **crisp** "true or false" (1 or 0). It enables computer processing to come closer to imitating human reasoning. For instance we can interpret two things as "tall" or "short"... or better "0.38 of tallness"

Fuzzy membership functions enable us to represent a fuzzy set graphically and also to quantify linguistic terms. For instance, let us consider :

- set : $A = \{20, 30, 40, 50\}$, where each element is denoted by X
- membership function of *set A* : $\mu_A(X) \rightarrow [0, 1]$. e.g. a triangular function that starts at 20 centers at 40 and ends at 60.
- if each element X is mapped is mapped to a value between 0 and 1 (via function $\mu_A(X)$) we obtain the membership degree of the element X to the *set/class* defined by the function, *i.e 40 is the set/class defined in the function above* - it can be age for classification of **mature persons**
- now we want to see how many elements of *set A* are members of class 40 - mature by calculating their membership degrees. Therefore the fuzzy set : $A_f = \{(20, 0), (30, 0.5), (40, 1), (50, 0.5)\}$

Building the Fuzzy modality

There are a great deal of fuzzy modalities ranging from (triangular, gaussian, trapezoidal etc.) that can be used to estimate our time-lag ; but it is very difficult to find one that will perfectly fit the data-set.

We recommend the membership triangular function since we are interested in approximating the **medial time-lag** such that if the value is taken as the center, the function should include a majority of the members.

The **TRUE center** is established by the largest proportion of members around it - so we propose a second algorithm that :

1. initially takes median as the center
2. slides the center left or right
3. re-calculates the membership function
4. repeats 1 to 3, until the **TRUE center** is found

Slide and Re-calculate Algorithm

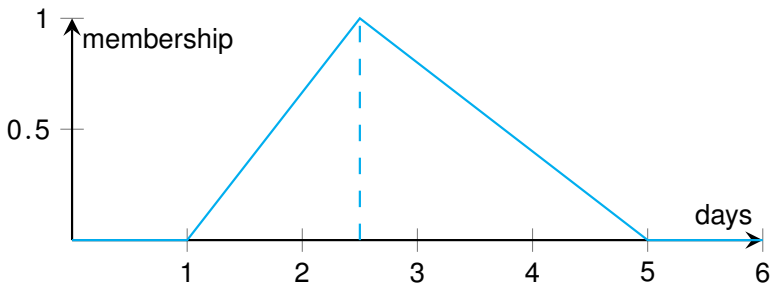


FIGURE – Initial membership function for r_{n+1}

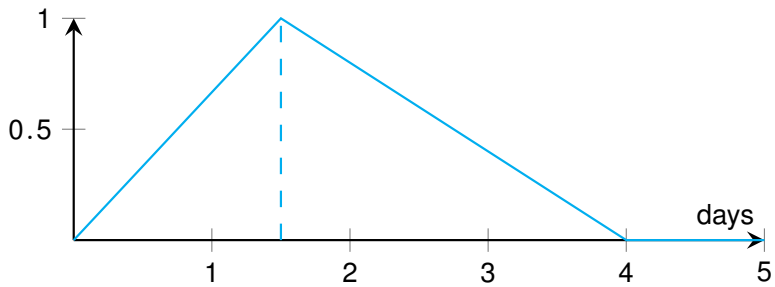


FIGURE – Modified membership function for r_{n+1}

data-set $\{3, 1, 5, 2\}$. We observe that the membership degree support of '$\simeq 2.5$' for path <t2, t3> : $\{1, 5\}$ is $\frac{0}{2}$

if we slide the center to the left, the membership degree support of '$\simeq 1.5$' for path <t2, t3> : $\{1, 5\}$ is $\frac{1}{2}$

conclusion : the fuzzy-temporal gradual pattern $\{(exercise, \uparrow), (stress, \downarrow)_{+\simeq 1.5 days}\}$ has a support of $\frac{2}{4}$, a *representativity* of $\frac{4}{5}$ and the *time lag* : '$\simeq 1.5 days$' has a support of $\frac{1}{2}$.

The T-GRAANK algorithm

The overall algorithm combines 3 algorithms : data-transform, slide & re-calculate and GRAANK. The main steps are :

1. transform data-set and calculate time-lag members
2. use time-lag members to build a fuzzy triangular membership function
3. perform GRAANK to retrieve concordant pairs
4. use concordant pairs and membership function to estimate **TRUE time-lag**
5. repeat step 1 to 4 for each transformation

Results for temporal gradual patterns

In order to test the efficacy of the $T - GRAANK$, we performed two separate tasks related to weather and compared their results. The aim was to confirm the conclusions of {doi :10.1080/01431169308954042}, that the NDVI (Normalized Difference Vegetation Index) is a sensitive indicator of the inter-annual variability of rainfall in the East African region.

In the first task, we retrieved the historical rainfall distribution amounts for 4 towns in Kenya (October-December 2013 and 2015) from Kenya Meteorological Service weather report, shown in the Table. Here, we selected two observable patterns : $\{(MAK, \downarrow), (WAJ, \uparrow)\}_{+=2years}$ and $\{(ELD, \uparrow), (NRB, \uparrow)\}_{+=2years}$, see also `http://www.meteo.go.ke/index.php?q=archive`.

town	**amount**	
	2013	2015
MAK	104	75
WAJ	49	69
ELD	174	200
NRB	44	223

TABLE – Rainfall distribution in Kenya

In the second task, we first generated NDVI data (for year 2013 and 2015) from LANDSAT 7 satellite images over Kenya through a novel tool known as *data-cube*. The *data-cube* is a great tool for the expanded use of satellite data in an Open Source framework, see also `https://www.opendatacube.org`.

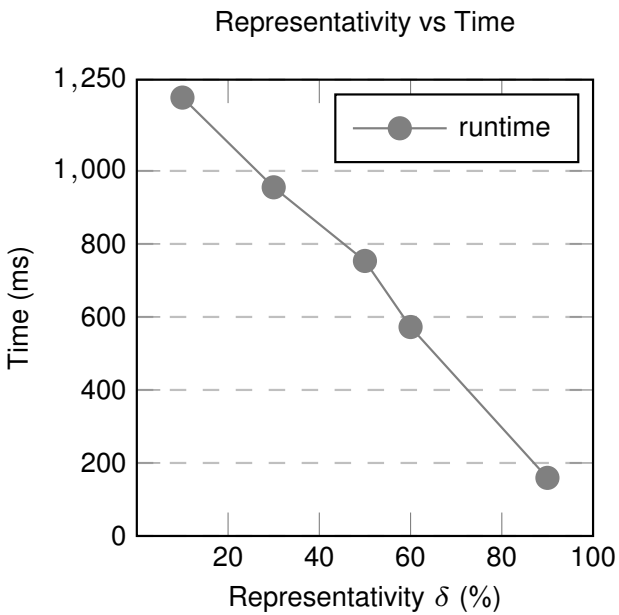
Lastly in the second task, we applied our approach on the NDVI data and we obtained the results shown in the Table. It can be seen that the patterns built by our algorithm match the selected patterns in the Table above ; except, the *time lag* is slightly less for pattern $\{WAJ+, MAK-\}$.

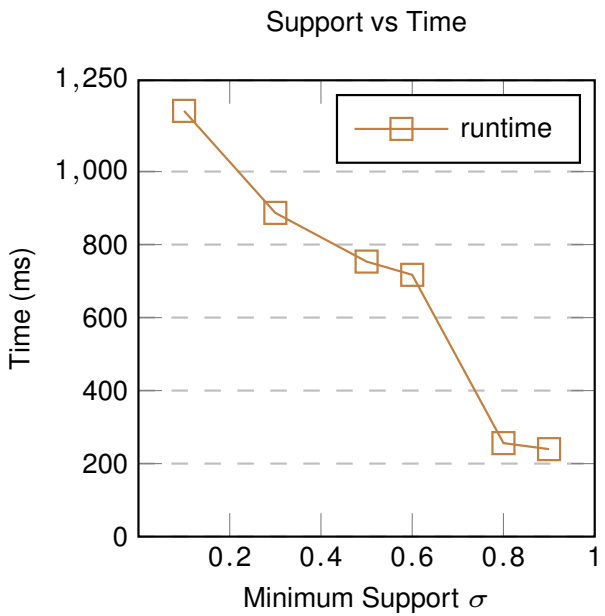
Ref. Item	Pattern : Sup	Time : Sup	Rep
NRB	$\{ELD+, NRB+\} : 0.666$	$\approx +1.999yrs : 1.0$	50%
	$\{WAJ+, NRB+, MAK+\} : 0.666$	$\approx +1.999yrs : 1.0$	50%
WAJ	$\{ELD+, WAJ+\} : 0.600$	$\approx +1.223yrs : 0.5$	62.5%
	$\{WAJ+, MAK-\} : 0.600$	$\approx +1.747yrs : 0.5$	62.5%

TABLE – NDVI Temporal Pattern Results

Short performance analysis

The runtime performances of our algorithm for temporal gradual pattern mining shown here were obtained from the execution of dummy data containing 50 tuples and 2 gradual items. The runtime values were generated by a python code that recorded the start-time and stop-time.





The graank desktop application

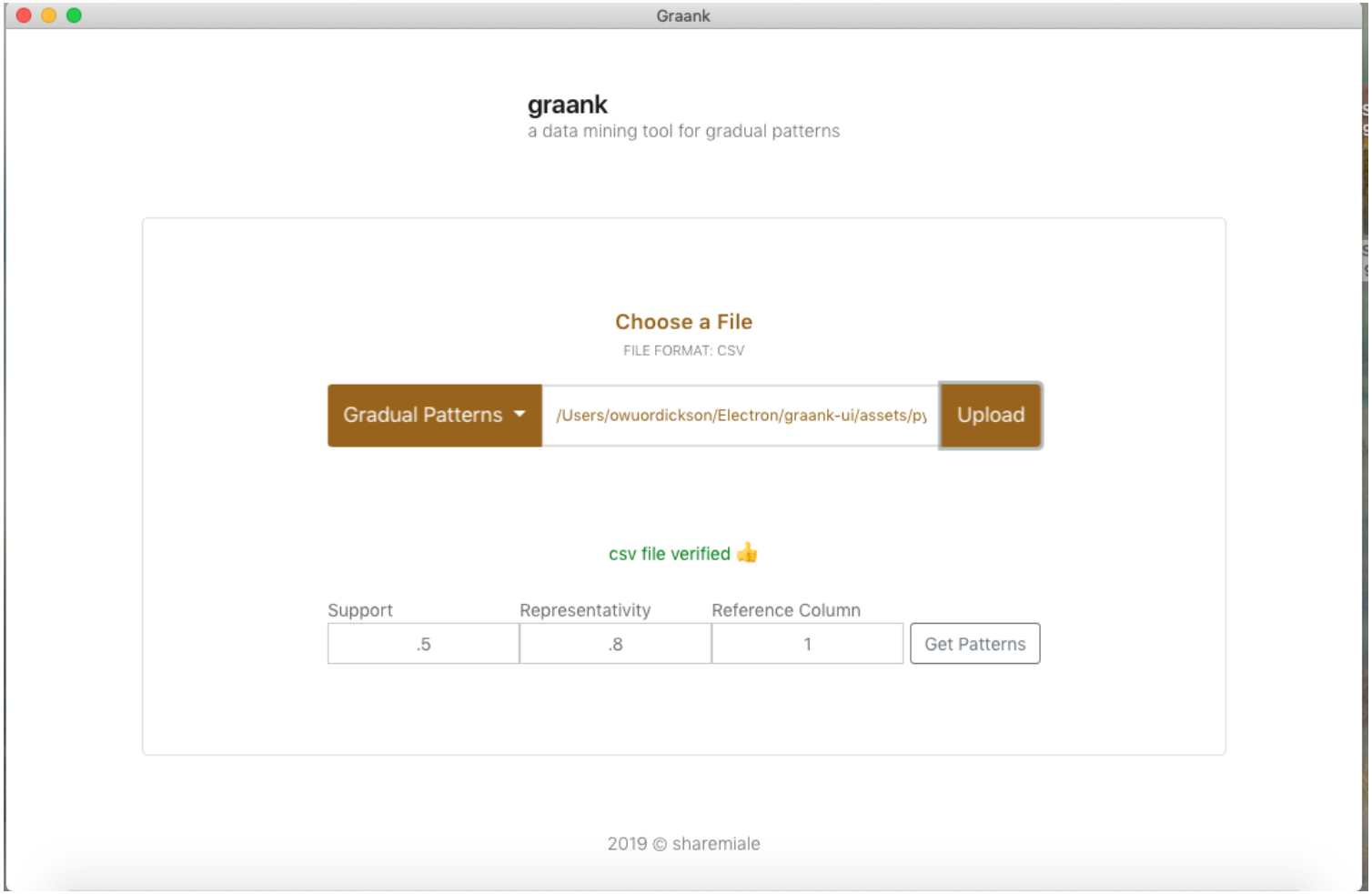


FIGURE – Mine Fuzzy-temporal Gradual Patterns. (github.com)

The graank desktop application

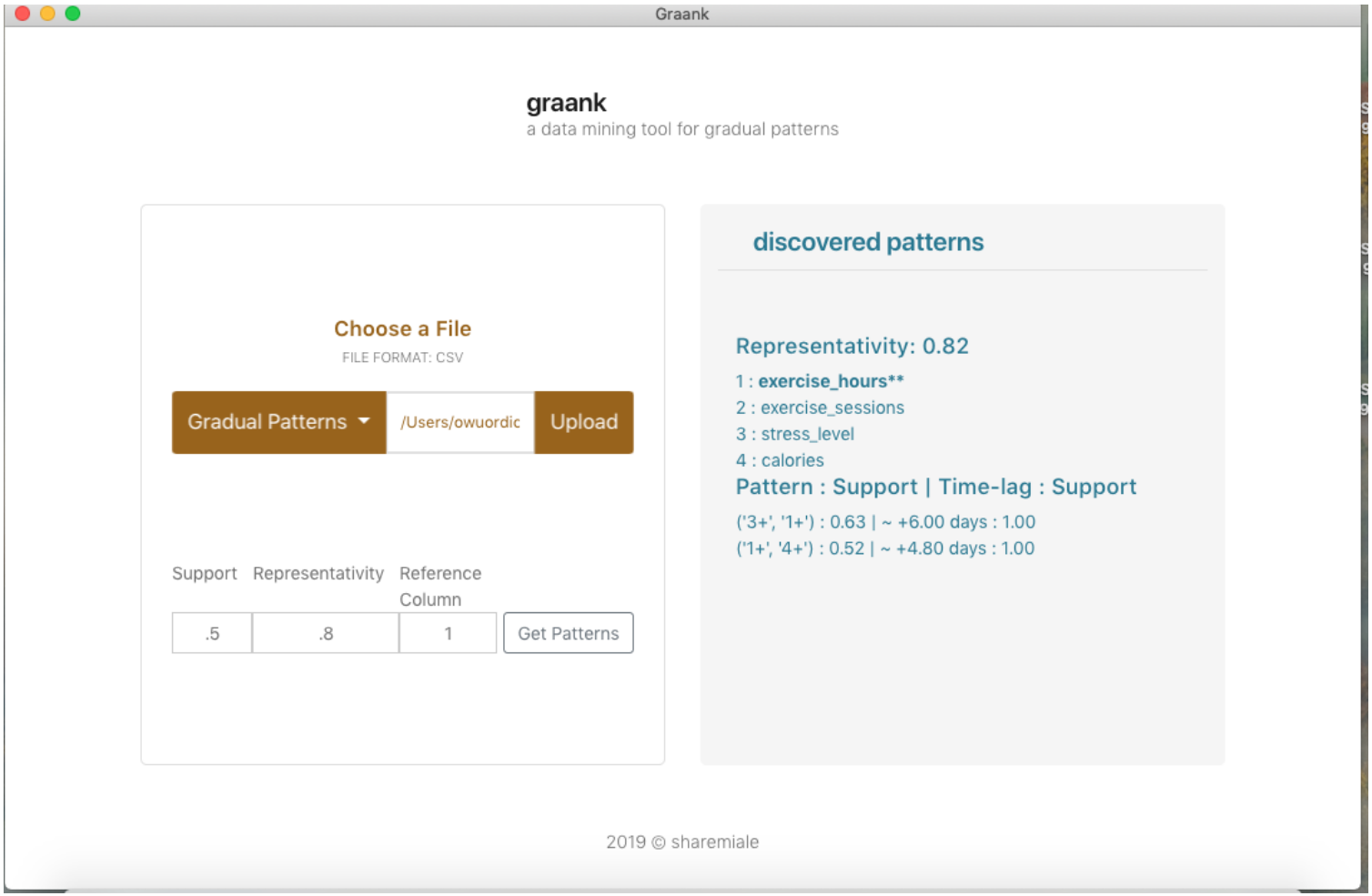


FIGURE – Results from Python Implementation. (github.com)

Conclusions

We have submitted this work to the FuzzIEEE Conference that will be held in July 2019 in New Orleans, USA. We are waiting for the verdict.

For the future, we wish to extend the approach in order to identify Emerging Patterns among the discovered fuzzy-temporal gradual patterns.

The T-GRAANK algorithm is available at our GitHub repository `https://github.com/owuordickson/t-graank.git`.

Thank you...