TALEX

Antonio F. G. Sevilla
`antonio@garciasevilla.com`

February 13, 2015

1 About

TALEX (Temporal Alignment Extractor) is an application for extracting and visualizing temporal synchronisation information in ANVIL[1] files. It is a web tool, but can be downloaded for offline use. Its current home on the Internet is at `https://garciasevilla.com/talex/`

This application is the result of the final assignment for the subject Multimodal Interaction and Communication, code LIN3502 at the University of Malta[2][3], academic year 2014-2015. The lecturer for this course was Prof. Patrizia Paggio[4].

2 Usage

To load the application, navigate to its url on the internet or the local filesystem. If downloaded as HTML app, click on the index.html file in the TALEX folder, or drag and drop it into a web browser. If downloaded as a standalone application, just execute the program.

---

[1]`http://anvil-software.org/`
[2]`http://www.um.edu.mt/`
[3]`http://lct-master.org/`
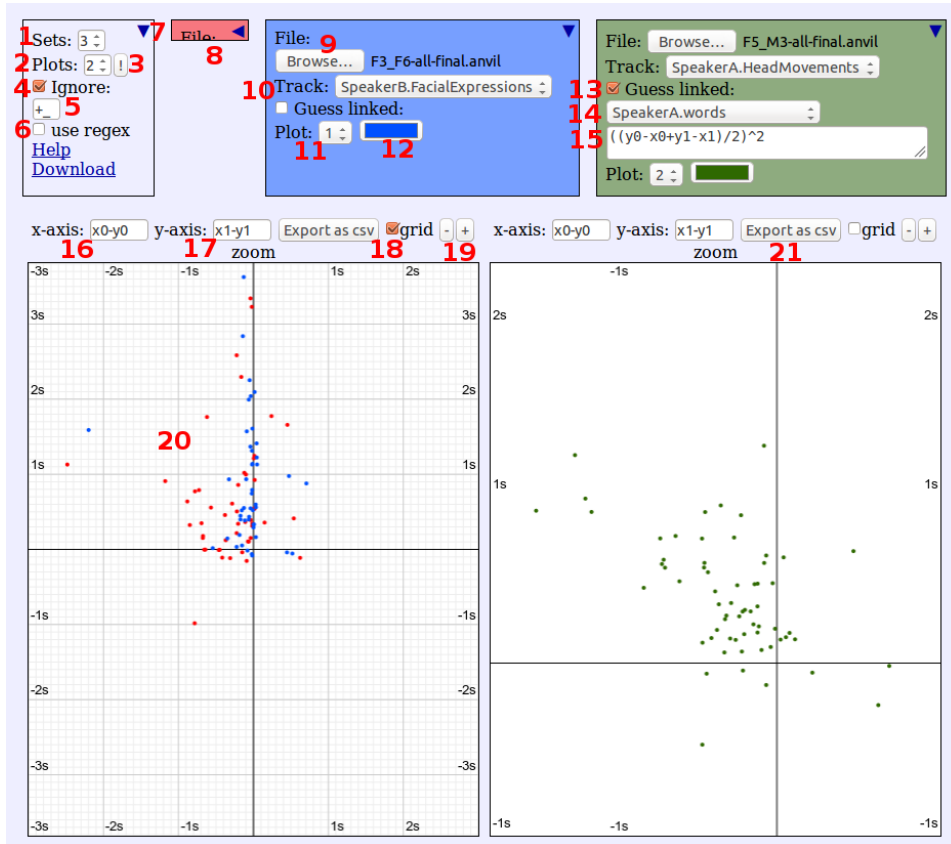[4]`http://cst.dk/patrizia/paggio/uk/index.html`

Figure 1: The TALEX screen

2.1 Option box (1-7)

- **Sets (1)**: How many datasets to plot. Each set will open a new Dataset box (8) to the right of the option box.

- **Plots (2)**: How many Graph boxes (20) to show. They appear in the bottom half of the screen. By clicking on the exclamation mark button (3) the arrangement of plots is changed.

- **Ignore (4)**: What tokens to ignore. If selected, the tool will act as if some of the tokens in the ANVIL file didn't exist. The omitted tokens will be those whose `textContent` matches the value in (5). If (6) is selected, this value will be interpreted as a regular expression.

- **Collapse (7)**: This arrows toggles the state of the option box between

2

shown and minimized. The Dataset boxes also have a collapse arrow.

2.2 Dataset box (8-15)

- **File (9)**: Select the ANVIL file to analyze. By default, when a new Dataset box is created the file in the last Dataset box will already be selected. This can make it easy to open more than one track in the same file: first select the file in the last open Dataset box, then increase the number of sets in (1).

- **Track (10)**: What ANVIL track are the gesture tokens in. Note that the tool doesn't care if the tokens are gestures or not, it can be used with any kind of ANVIL token that has start and end points.

- **Target plot (11)**: What plotbox to draw the points from this dataset in.

- **Colour (12)**: What colour to use for the points. The background of the Dataset box also has this colour, so it is easy to know from what Dataset box each set of points come.

- **Guess linked (13)**: TALEX can try to guess what words are associated to each token. If (13) is selected, the attribute `MMRelationSelf` of the tokens will be ignored, and the associated tokens will try to be automatically detected. The track to look for candidates is selected in (14), and the distance function to use can be input in (15). Mathematical function such as `sin` or `abs` can be used. For a conditional function, use the ternary operator: `condition ? value if true : value if false`. For details see section 5.

2.3 Plot box (16-21)

- **Plot area (20)**: The values are plotted in this area. It can be moved around by clicking and dragging with the mouse. The scale can be changed with the mouse wheel, or with the buttons plus and minus (19).

- **Axis (16, 17)**: Here the value to be plotted should be input. The data from the Dataset box is x0, y0, x1 and y1 as described in the first chapter. Read section 3 for more details.

- **Grid (18)**: If selected, a gray grid will be drawn as background.

- **Export (21)**: If this button is clicked, all the points currently plotted in the graph will be output as a csv file. The format of this csv file will be x-coordinate, comma, y-coordinate for points in the first dataset, x and y for the second dataset, etc. That means, if more than one dataset is plotted in this graph, each will be two columns of the output csv, the first one being the x-coordinate and the second one the y-coordinate. Note that the columns will not have the same length. Null values will just be represented as nothing between the commas. The last row in the file will be all commas.

3 Axis values

Changing the formulas in (16) and (17) allows us to graph (and export) a number of different measurements based on the data timepoints. By default, the axes are set to start delay $(x_0 - y_0)$ and end delay $(x_1 - y_1)$. Another measurement that can be displayed is that of duration, $x_1 - x_0$ and $y_1 - y_0$. See section 4 for the available values.

A useful trick is, if we want to only observe one variable, to choose a dummy independent variable for the x-axis. We don't have access to a unique identifier, but the start time can serve as one. For example, if we want to observe the start delays just by themselves, we can use as x-value $x_0/50$ and as y-value $x_0 - y_0$. We divide the x-value by 50 because the tool is made to display small time scales, and doesn't allow enough zoom. This might change in a future version. For now, we scale the values so they fit in the graph. See figure 2 for an example.
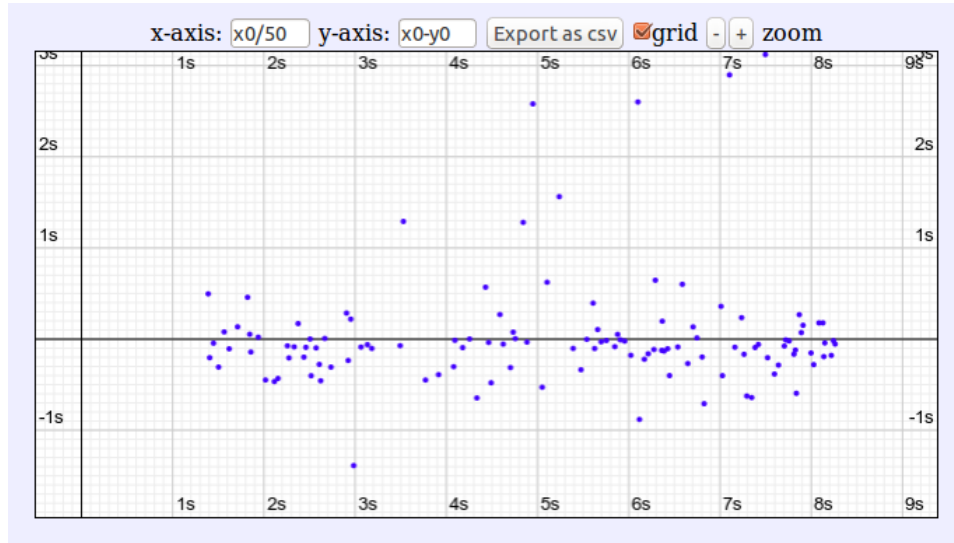

Figure 2: Plotting different values on the axis

4 Timepoints

To study the time relations, first we must convert the annotations into numerical data we can analyze. For each token in a gesture track, we extract four different values, the timepoints corresponding to the start of the token, the end of it, the start of the associated word or word group, and their end.

The associated or linked words are those that the manual annotation marks as such. Each gesture token may have a number of related word tokens. These are encoded under the attribute "MMRelationSelf". For example, if a subject nods while saying "yes", the head gesture corresponding to the nod is annotated as relating to the word token for "yes". More than one word can correspond to each gesture, for example a phrase or a whole sentence, but it need not be a syntactically correct structure. These relations are derived by the human annotator, and TALEX is agnostic towards their meaning, it just sees a gesture token and possibly one or many linked words.

The four points extracted contain all the information that we have available at this level. The measures that we will analyze are all functions of these values. For example, we analyze the time delay of a gesture token with respect to its associated word by computing two values, the "start lag" $(x_0 - y_0)$ and the "end lag" $(x_1 - y_1)$. The notation used in TALEX is that of assigning to the gesture token the name x, and to the linked word (or phrase) the name y. The subscript 0 will denote the starting timepoint of the element, and 1 the ending timepoint. Thus, y_0 means the start time of the (first) associated word, and y_1 the end time of the (last) associated word.

5   Automatically detecting links

TALEX can try to guess what words are associated to each token. Thus, is an ANVIL file is not annotated with "MMRelationSelf" attributes, linking gesture tokens with word ones, this linking can try to be automatically deduced. Note that this function is very experimental and has not been proven to give (semantically) reliable results.

The proposed way to associate token pairs is by means of a distance function. The algorithm looks at every gesture and every word token, measures the distance between them, and decides that the related words for the gesture are those with the minimal distance. While this might seem simple at first sight, the key is in the distance function. Any kind of mathematical relation can be encoded into it, so long as this function ranks potential candidates in a convenient order. An equivalent way to look at it is by thinking the function assigns a likelihood of pairing, and the algorithm selects the maximal likelihood.

Of course, we want to allow the linking to occur between one gesture and more than one word. That is why the distace function need not be one-to-one. If more than one possible candidates give the minimum distance, all of them are used as the associated tokens. Then, the first of them will be used to compute the start delay, and the last one for the end delay.

The values available are x0, x1, y0 and y1 as per section 4. Example distance functions to try are:

1. Euclidean middle points distance: `((y0-x0+y1-x1)/2)ˆ2`

$$dist(x, y) = \left(\frac{y_0 - x_0 + y_1 - x_1}{2} \right)^2$$

2. Edges distance: This distance measures the gap between the closest edges of the tokens. If the tokens overlap, the distance is assigned 0. In interval algebra terms, if $A < B$, then the distance is from the end of A to the start of B, if $A > B$, from the start of A to the end of B. If AoB, then the distance is 0. Arithmetically:

$$dist(x, y) = \begin{cases} 0 \text{ if } (y_1 > x_0 \text{ and } (y_1 < x_1) \text{ or } (y_0 > x_0) \text{ and } (y_0 < x > 1) \\ min((x_0 - y_1)^2, (y_0 - x_1)^2) \text{ otherwise} \end{cases}$$

The function: `(y1>x0 and y1<x1) or (y0>x0 and y0<x1) ?  0 :` `min( (x0-y1)ˆ2, (y0-x1)ˆ2 )`