webAKT Introductory Tutorial

Robert Muetzelfeldt
23 Jan 2023

Glossary

AKT: The Agroecological Knowledge Toolkit project.
AKT5: The software originally developed in the late 1990s within the AKT project to allow knowledge bases to be created, inspected and analysed. This was a desktop application (meaning it had to be installed on your computer) written in LPA's WinProlog.
webAKT: The replacement for AKT5, written as a web application (using HTML/CSS/JavaScript), and designed to be accessed using any web browser without having to install any software. This is currently under development, so please follow the instructions carefully, and be patient if some things do not work perfectly.

HOW TO ACCESS webAKT

- Open a browser – Chrome is recommended, others should work.
- Go to **http://www.webakt.org/msc3/webakt.html**

Maximise the window, so that the workspace area in webAKT is easier to work with. You can also choose the Full screen option of the Zoom command in the Chrome settings menu to get an even bigger workspace.

You will see a simple window with a menubar. We will be using the following three menus in this tutorial:

- **KB**: allows you to switch between knowledge bases, and present basic information about the main components of the knowledge base in an interactive panel.
- **Diagram**: displays a diagram of the causal relationships within the knowledge base and allows you to manipulate the diagram.
- **Tools**: gives access to various additional ways of displaying and analysing the contents of a knowledge base.

<u>You will cover the following sections:</u>

1. **Metadata –** finding out information about the knowledge base.
2. **Statements –** looking at the collection of statements in the knowledge base.
3. **Formal terms –** exploring the words provided by the knowledge base creator that make up the statements, including words for objects.
4. **Object hierarchies –** seeing how objects are organised into hierarchies.
5. **Topics –** seeing how statements can be grouped into clusters depending on the formal terms they reference.
6. **Topic hierarchies –** seeing how topics can be organised into hierarchies.
7. **Sources –** looking at the local people who provided the individual statements.
8. **Boolean search –** learning how to search for statements that using formal terms combined by 'and' and 'or'.
9. **Diagram –** displaying the network of causal statements for a particular topic.
10. **Tools –** looking at additional methods for analysing the knowledge base.

THE KNOWLEDGE BASES

webAKT is pre-loaded with two knowledge bases: **Atwima** and **Ego**.   Atwima is the currently-selected one, meaning that the various actions you will be performing in this tutorial will automatically operate on this knowledge base, until you explicitly change it once you have worked through this tutorial.

1. METADATA

Each knowledge base has information about the knowledge base (rather than in it) – who created it, when it was created, at what locality, etc. This is known as metadata.

▶ Enter the menu command: **KB > Metadata**
I.e. select the **KB** *menu, then select the* **Metadata** *command in this menu.*

This displays the metadata for the current knowledge base. You can close the window by clicking on the X in the top-right corner if you want to.

2. STATEMENTS

The heart of an AKT knowledge base is the statements it contains – statements which were obtained by interviews with local farmers. There are 4 different types of statements:

1. **Attribute-value** (att_value) statements tell you about the properties (attributes) of something – they are descriptive.
2. **Causal** statements give you information about cause and effect relationships.
3. **Comparison** statements compare the properties of two objects.
4. **Link** statements represent the connections between objects that cannot easily be represented using the other statement types (these are rarely used, and are not in the knowledge bases you will be using).

Statements are typed into the knowledge base as formal language statements using a formal grammar specific to AKT. These are then translated by the webAKT program into stylised natural language equivalents that are easier to understand by the user.

▶ Refresh the browser window, and enter the menu command: **KB > Statements**

Initially you see a listbox containing all 445 statements in the Atwima knowledge base. The statements are rendered in a stylised form of English – they are automatically converted from the underlying formal notation. The listbox shows both the formal and the English-like form of each statement.

We will now use the filtering options available at the top of the page.
Note that the checkboxes for the Include… filters indicate which types of statement are included, while the checkboxes for the Restrict to… filters only include statements which have the specified value for the associated dropdown menu.

▶ Uncheck the **causal** and **comparison** checkboxes, leaving **att_value** checked. You will see 31 attribute-value statements.

▶ Now uncheck the **non-conditional** checkbox – you will see just two conditional attribute-value statements, both containing a condition (signalled with the word ‘if’).

▶ Now re-check these checkboxes, and check the **source** checkbox. You will see 26 statements which have **Oliver_Gogoikrom_2000a** (the current option in the dropdown menu) identified as the source.

You notice that there are also filters for formal terms and topics – you can return to these once you have finished the tutorial and understand more about the formal term and topic categories.

Question: How many causal statements are there? How many of these are conditional? How many attribute-value statements had James_Kyereyase_2000a as their source?

▶ Now, unclick the **Source** checkbox, and click on the statement **s6**, then click on the **View** button. This shows the statement in both their formal and English forms; the source(s) for the statement; and other buttons for working with the statement.

Note that the formal terms in the statement are colour coded. Each colour show the type of the formal term – its grammatical role in the statement. If you mouse over a word, you will get a small popup telling you the formal term type of that word – and all the words of that colour.

3. FORMAL TERMS

Formal terms are the terms entered by the knowledge base creator when entering statements, excluding built-on terms such as "att_value", "process" or "and". The names of objects are one type of formal term – words that refer to material or conceptual things e.g. pests, soil, cows, policy, and household. Other types of formal terms are actions – activities with a human agent e.g. harvesting, and processes – activities without a human agent e.g. germination. You will notice that underscores e.g. asase_tuntum, are used instead of spaces in webAKT.

▶ Refresh the browser window, and enter the menu command: **KB > Formal Terms**

▶ Click on the dropdown **Formal term type** menu to see the different types of formal terms. Select **object**. The names of all the objects in the knowledge base are now listed. Scroll down and get an idea of the objects in the knowledge base.

▶ Click on **esre**, then click on **View**. This opens up a new panel, and shows that there are two synonyms for esre: *Panicum maximum* and grass. Now click on **Statements** in the formal_term_details panel. The Statements panel (which you have already seen) appears, showing the 39 statements that mention esre. Note that the **formal term** checkbox has been ticked automatically, and **esre** is the selected formal term.

Questions:

1. How many formal terms are there for actions?
2. How many statements are there for the action of cutting?

4. OBJECT HIERARCHIES

Object hierarchies are a method for sorting knowledge by arranging specific objects under more general objects e.g. esa, funtum and onyina are all types of trees. Therefore, esa, onyina and funtum are all subobjects of the object 'trees', and 'trees' is a superobject of the objects esa, onyina and funtum.

Plant and animal taxonomies are familiar examples of object hierarchies, but the same object can exist in other hierarchies: for example, 'maize' can be both in a plant-taxonomy hierarchy and in a farm-produce hierarchy. Similarly, in AKT the same object can exist in more than one hierarchy, created by the knowledge base creator, which is why we talk about "object hierarch**ies**" (plural).

▶ Refresh the browser window, and enter the menu command: **KB > Object Hierarchies**

You see a listbox showing the object hierarchies in the knowledge base. Select **weeds**, then click on the **View** button. This displays a panel containing a list of all the first-level subobjects in the weeds hierarchy.

▶ Click on this triangle next to the subobject **mwura_bone**, which should display its own subobjects.

▶ Click on **esre**, then click on the **Details** button: [[** Need to bring this in line with the View/Edit/New pattern **]] this opens up the same panel you have already seen, giving details about esre. So getting down to an individual object can be done either by starting off with the alphabetical list of objects (as you did in the previous section), or by moving down through the object hierarchy,

▶ Now, refresh the browser window, select **Formal terms** from the **KB** menu, select **object** in the Formal term type dropdown, select **esre**, click on the **Details** button, and click on the **In hierarchies** button. The Object hierarchies panel opens up again, showing esre in its hierarchy under weeds/mwura_bone. So, you can either start off with the object hierarchy and work your way down; or you can start off with an object and see where it occurs in the object hierarchy!

Question:

5. TOPICS

Topics are ways of organising information for particular subject areas e.g. ‘Weed management’.

▶ Refresh the browser window, and enter the menu command: **KB > Topics**

This gives you a list of all the topics in the knowledge base.

▶ Click on **Management actions**, then click on the **Details** button. In the panel that appears you will see in the ‘Boolean Search String’ how the topic was created – it is a search for any of the following action words – ‘burning’, ‘clearing’, ‘cutting’, ‘harvesting’, ‘planting’, ‘uprooting’, ‘use’, ‘work’.

▶ Click on **Statements** at the bottom of the panel and a list of all the statements on the topic “Management actions” will appear. There are 107 statements in all. Close the list of statements and the topic details. Try the same thing with the topic **Weed control**.

Question: How many statements are there on weed control?

6. TOPIC HIERARCHIES

Just as we saw with objects, topics can be arranged in hierarchies: specific subject areas are arranged under a more general topic heading e.g. the topic hierarchy ‘Fallow length’ has subtopics ‘Fallow management’ and ‘Fallows and soil types’. And, as before, the same topic can be in more than one topic hierarchy.

▶ Refresh the browser window, and enter the menu command: **KB > Topic Hierarchies**

You see a listbox showing the topic hierarchies in the knowledge base.

▶ Select **Soil types**, then click on the **View** button. This displays a panel containing a list of all the first-level subtopics in the Soil types hierarchy.

▶ Click on the grey triangle next to the subtopic **Red soil**, which should display its own subtopics.

▶ Select the subtopic **Red soil** by clicking on the word itself. Click on the Display button to open up a panel just for the Red soil topic.

▶ Now, just as we did for objects, refresh the browser window, select **Topics** from the **KB** menu, select **Red soil** from the list, click on the **View** button, and click on the **In hierarchies** button.

This opens up the Topic hierarchy panel showing where Red soil comes in the Soil types hierarchy.

Question: What topics are shown to be in the topic hierarchy ‘Weeds’?

7. SOURCES

Sources tell you the origin of a statement. All statements have a source, which can be of 2 types: an interview with a person e.g. a farmer or scientist, or a reference e.g. a journal reference.

▶ Refresh the browser window, and enter the menu command: **KB > Sources**

On the left is a list of all the sources interviewed for the knowledge base. Let us look at one of them.

▶ Highlight the name **Antwi,_K_Kyereyase_2000a** and click the **View** button. A panel appears giving you the name of the person interviewed, the interviewer and date of interview. You are also given the gender, age and ethnic origin of the interviewees and a location which is their town of residence.

▶ Click on the **Statements** button in the **Source details** panel to see all the statements by this source.

Question:

8. BOOLEAN SEARCH

The Boolean Search facility allows you to select statements according to the formal term(s) the statement contains and/or the source(s) of the statement. The search expression may be very simple - e.g. a single formal term, such as "planting" or 'onyina', or a complex combination of formal terms joined by 'and' and/or 'or, such as "(planting or harvesting) and onyina".

▶ Refresh the browser window, and enter the menu command: **KB > Boolean Search**

▶ Go to the **Formal term type** listbox and click on the dropdown menu to select **object**.

▶ Select **asase_kokoo** from the listbox. Now click on the **Select** button and asase_kokoo will appear in the Boolean Search String at the bottom of the panel.

▶ Then click on the **and** button.

▶ Click on **asase_tuntum** in the listbox, and click on **Select** once more. The search expression should now read "asase_kokoo and asase_tuntum" (without the quotes).

▶ Now click on the **Search** button. Two statements will appear in the Statements panel. These are the only statements in the knowledge base which include both 'asase_kokoo' and 'asase_tuntum'.

▶ Close the Statements panel by clicking on the **X** in the top right corner.

▶ In the Boolean Search panel click on the **Clear** button.

▶ Now repeat what you did before, selecting **asase_kokoo** and **asase_tuntum**, only this time click on the **or** button instead of 'and', so the search expression now reads "asase_kokoo or asase_tuntum" (without the quotes).

▶ Click on the **Search** button again. Now you have 30 statements. This is because you have selected all the statements that include either 'asase_kokoo' or 'asase_tuntum'.

Question:

9. DIAGRAMMING

One of the main types of statement in an AKT knowledge base is the causal statement: "A causes B". These naturally lend themselves to being displayed as a node-and-link network diagram: A and B are nodes, with a link between them. In computer science, such network structures are called a "graph" (or, to be precise, a "directed graph", since each link has a direction), not to be confused with the common use of the term to mean a plot of data.

AKT5 allows a diagram to be generated from a subset of the causal statements in a knowledge base. Typically, the subset is defined by a topic. webAKT provides the same facility, and uses an algorithm (a "graph layout" method) to make the diagram look nice. Once generated, the nodes can be moved around to further improve the diagram's appearance.

AKT5 provides a number of tools for navigating around the diagram. Currently, in webAKT, the only thing you can do is to mouse over a link to see the causal statement that was used to generate that link.

AKT5 also allows the diagramming tool to be used in reverse; the user can draw a diagram with nodes and links, with each node-link-node structure being used to add a causal statement to the knowledge base. This has been prototyped in webAKT, but is not yet ready for release.

▶ Refresh the browser window, and enter the menu command: **Diagram** > **Diagram from Topic**

This opens up a panel with a blank canvas. Click on the **Topics** dropdown menu to open it up, Then move the mouse over the various topics. Notice that, when you mouse-over and pause over one, it displays the Boolean search term which defines that topic.

▶ Select the topic **Black soil**, then click the **Display** button. You should see a simple diagram, with 3 orphan causal links, and a causal chain with 4 links in it.

▶ Mouse over a link. You will see its two nodes and its link highlighted, and the original causal statement for that link pop up in a little window. Note that the label for the two relevant nodes in the diagram is made up from the start and end parts of the causal statement, except for the final term (the value for an attribute or property). This mirrors the approach used in AKT5.

▶ Now try changing the appearance of the diagram. You can drag nodes around using the mouse. You can also drag a link curve to make it more or less curved. (At the moment, these changes are not saved, but they will be in future.)

▶ Click the **Display** button again, still for **Black soil**. You get a new diagram, with a different layout. So you can repeatedly generate a diagram for a particular topic until you get one you like, before editing it as above.

For a larger causal network, try (for example) **Chromolaena**. This will take longer to display, since there are more nodes and so the layout algorithm has more work to do.

Try a number of other topics. Some topics have no causal statements, so you will get a message telling you this. (An obvious improvement to this tool is to display the number of casual statements next to each topic in the dropdown menu, so it's not trial-and-error – it's on my TODO list.)

10. TOOLS

AKT5 had a major distinction between the ways used to present information under the KB menu, that you have worked with so far, and "Tools". The latter were intended to allow new types of analysis and display to be created in response to requests from users. In principle, any user could learn the special language developed for this purpose (the AKT "Task Language") and build their own tools. In practice, few users did this, and most tools were built by the person maintaining AKT5.

webAKT completely removes this distinction, and uses a standard programming pattern (called a "widget") for anything that involves getting information from the KB and displaying it in a panel. The 'Statements' panel and the 'Statement detail' panel were two such widgets. However, for legacy reasons, the separation between the KB menu (with the emphasis on exploring a KB) and Tools (with the emphasis on KB analysis) is maintained in webAKT, by having the two separate menus. Moreover, as in AKT5, anyone with (or with access to someone with) skills in HTML and JavaScript programming can add new Tools to their own local copy of webAKT, without having to go to some central body.

Today you can look at three such Tools, each one being a direct replica of the corresponding System Tool in AKT5, to give you an idea of the sorts of Tools that AKT users want. They are presented in order of complexity, the first being very simple. The last one, species_summary, was, as an experiment, deliberately designed to have a line-for-line correspondence with the Task Language used to define the same tool in AKT5.

▶ Refresh the browser window, and enter the menu command: **Tools > Use a tool > statements_summary**
This simply tabulates the number of statements in the knowledge base, divided into attribute/causal/comparison and non-conditional/conditional categories.

▶ Menu command: **Tools > Use a tool >hierarchic_objects_usage**
Produces a table showing the occurrence of the same object (formal term) within the various object hierarchies in the knowledge base.

▶ Menu command: **Tools** > **Use a tool** > **species_summary**
The aim of this tool is to bring together in one place all the information on a particular species (in practice: a formal term object). For this exercise, I suggest restricting your choice of species to **nyanya**, **cocoyam** and **esre**, though you can try others if you want. You need to type in the species name, which you can get by looking at the list of objects in the Formal Terms panel. Note the use of colour-coding of terms to indicate their grammatical role; and the ability to include an image of the species.

[[Postscript: If you have found this exercise – and the wider topic of trying to capture farmer's knowledge – interesting, and would like to get involved as we continue to work on webAKT, please contact me at robertmuetz@mailfence.com. You don't need any programming expertise to help with ideas and testing. If you do know some HTML/JavaScript, then there is plenty of scope for help in debugging and developing new tools.]]