

Designing a New Tool for E-learning in Syllogistics

Peter Øhrstrøm and Ulrik Sandborg-Petersen

Department of Communication and Psychology, Aalborg University,
Rendsburgade 14, 9000 Aalborg, Denmark
{poe,ulrikp}@hum.aau.dk

Abstract. This paper is a continuation of earlier studies involving practical experiments with students using various versions of the e-learning system, Syllog. It has been investigated to what extent such tools can be helpful for the students in e-learning and in the context of logic teaching in order to obtain a better understanding of syllogistic reasoning. The aim of the present paper is to discuss how we can design a new and better e-learning tool based on the experiences with Syllog so far.

Keywords: Syllogistics, logic teaching, e-learning tools.

1. Introduction

In this paper we present the design of a new tool for e-learning in Aristotelian and modern syllogistics. This tool should make it easier for the students to obtain a better understanding of syllogistic reasoning and argumentation in general. The new tool is designed using findings from earlier studies involving practical experiments with students of Communication using various versions the Syllog tools (Øhrstrøm et al. 2010, 2013a, 2013 b, 2014, 2015, 2016; Sandborg-Petersen et al. 2013, 2015).

The new tool should of course support the introduction of Aristotelian and modern syllogistics (see Parry et al. 1991). In particular the use of the tool should support a deeper understanding of logical validity. This means that the students should get more familiar with the notion of validity through the use of the tool. The students should thereby become able to answer what it means that an argument is valid and what it means that an argument is invalid. Such answers also involve the ability to explain why an argument is valid or why it is invalid.

Regarding the design of the new tool we shall concentrate on the following functionalities:

- a) Generation of arbitrary syllogisms
- b) Classical evaluation of the validity of a syllogism
- c) Explaining the validity of a syllogism

These functionalities of the new tool will be discussed below in the sections 2, 3, 4 and 5. The overall design of the new system will be presented in the conclusion.

2. Generation of arbitrary syllogisms

In earlier versions of Syllog we have developed tools that offer generation of arbitrary syllogisms that the user is supposed to evaluate (Øhrstrøm et al. 2013 a, 2013b, 2014). So far we have found it useful to generate valid and invalid syllogisms with equal frequency. This means that it is randomly decided whether the next syllogism to be generated by the system should be valid or invalid, and then a random syllogism is picked with the chosen validity. This algorithm will also be used in the new tool.

In (Sandborg-Petersen et al. 2015) and in (Øhrstrøm et al. 2015) we have discussed the use of meaningful versus meaningless terms in the syllogisms offered randomly. It seems interesting to have both options. Furthermore, it may be useful to have a flexible system that will allow several kinds of terms in the system. The most attractive version of the new tool will allow the teacher to choose the set of terms which is going to be active in the system.

3. The representation of Aristotelian syllogisms

Classical Aristotelian syllogistics can be presented as a fragment of first order predicate calculus. A classical syllogism may be represented as a simple implication in the following manner; see (Øhrstrøm et al. 2013 a, 2013b, 2014)¹:

$$(p \wedge q) \supset r$$

where each of the propositions p , q , and r matches one of the following four forms

$$\begin{array}{ll} a(U, V) & \text{(read: "All U are V")} \\ i(U, V) & \text{(read: "Some U are V")} \end{array} \quad \begin{array}{ll} e(U, V) & \text{(read: "No U are V")} \\ o(U, V) & \text{(read: "Some U are not V")} \end{array}$$

We may express these four functors in terms of first order predicate calculus in the following way:

$$\begin{array}{ll} a(U, V) & \leftrightarrow \forall x: (U(x) \supset V(x)) \\ i(U, V) & \leftrightarrow \exists x: (U(x) \wedge V(x)) \end{array} \quad \begin{array}{ll} e(U, V) & \leftrightarrow \forall x: (U(x) \supset \sim V(x)) \\ o(U, V) & \leftrightarrow \exists x: (U(x) \wedge \sim V(x)) \end{array}$$

The four basic propositions can be related in terms of negation:

$$i(U, V) \leftrightarrow \sim e(U, V) \quad o(U, V) \leftrightarrow \sim a(U, V)$$

The classical syllogisms occur in four different figures:

$$\begin{array}{ll} (u(M, P) \wedge v(S, M)) \supset w(S, P) & \text{(1st figure)} \\ (u(P, M) \wedge v(S, M)) \supset w(S, P) & \text{(2nd figure)} \\ (u(M, P) \wedge v(M, S)) \supset w(S, P) & \text{(3rd figure)} \\ (u(P, M) \wedge v(M, S)) \supset w(S, P) & \text{(4th figure)} \end{array}$$

¹ A more elaborate version of classical syllogistics can be found in these earlier papers.

where $u, v, w \in \{a, e, i, o\}$ and where M, S, P are variables corresponding to “the middle term”, “the subject” and “the predicate” (of the conclusion).

In this way, 256 different syllogisms can be constructed. According to classical (Aristotelian) syllogistics, however, only 24 of them are valid. The medieval logicians named the valid syllogisms according to the vowels, $\{a, e, i, o\}$, involved. In this way the following artificial names of the valid syllogisms were constructed (see Parry & Hacker 1991):

- 1st figure: barbara, celarent, darii, ferio, barbarix, feraxo
- 2nd figure: cesare, camestres, festino, baroco, camestrop, cesarox
- 3rd figure: darapti, disamis, datisi, felapton, bocardo, ferison
- 4th figure: bramantip, camenes, dimaris, fesapo, fresison, camenop

In the various versions of Syllog so far we have implemented an evaluation of the syllogisms following this classical approach. This means that a syllogism is valid if and only if it has a form that belongs to the above list of 24 valid syllogistic forms. It also means that the system gives the medieval name in case the syllogism in question is valid (see Sandborg-Petersen et al. 2015). This facility is kept in the new tool. However, in this new implementation it will also be explained why the syllogism is valid, or why it is invalid. This further explanation is seen as essential in e-learning if the student is supposed to obtain a deeper understanding of syllogistic validity through the use of the new tool. In the two next sections, it will be shown how these explanations should be offered with the new tool.

4. Explaining the validity of valid syllogisms

Let us assume that the system has randomly generated the following syllogism:

Some students are fathers
 All students are females
 Ergo: Some females are fathers

We also assume that the user has correctly evaluated it as valid. The user of the new tool should then explain why it valid. This explanation will be given in terms of a proof, i.e., a series of applications of the following five rules of inference starting from the two premises of the syllogism (Øhrstrøm et al. 2014 and 2016)²:

- | | | | |
|----------|--|---------|--|
| (TRANS) | All Y are Z
All X are Y
Therefore: All X are Z | (SUBST) | All Y are Z
Some X are Y
Therefore: Some X are Z |
| (CONTRA) | All X are Y
Therefore: All non-Y are non-X | (MUT) | Some X are Y
Therefore: Some Y are X |
| (EX) | All X are Y
Therefore: Some X are Y | | |

² A more elaborate version of this presentation of proof theory in classical syllogistics can be found in (Øhrstrøm et al. 2016).

This means that in case of a valid syllogism, it will be possible to produce a deduction of the conclusion from the two premises using these five rules of inference.

It should be noted that this only works if we allow for negations of terms. The term non-X simply stands for all elements in the universe that are not instants of X. Clearly, non-non-X (i.e. a double negation) would be equivalent with X. It should also be noted that the e-proposition, “No X are Y”, can be reformulated as “All X are non-Y”. Similarly, the o-proposition, “Some X are not Y” can be reformulated as “Some X are non-Y”. This means that in terms of the controlled natural language the number of types of propositions in syllogistic reasoning can be reduced from four to two, namely the universal propositions (i.e. “All ... are ...”), and the particular propositions (i.e. “Some ... are ...”). In combination with the option of term negation and the above inference rules we have everything that we need in order to evaluate all possible syllogisms in classical syllogistics.

The use of the inference rule (EX) has sometimes been seen as controversial, and the 9 syllogisms which depend on this rule have consequently been seen as “questioned”. Clearly (EX) has to be rejected, if we hold that the statement “all S are P” is true given that S is the empty set. Therefore, if this is accepted it should obviously not be permitted to deduce “some” from “all”. If the (EX) rule is excluded, the number of valid syllogisms is reduced from 24 to 15.

It can be demonstrated that all valid syllogisms can be proved using the five rules of inference. This proof system for classical syllogism has been implemented in an earlier Syllog tool (Sandborg-Petersen et al. 2015). In this system the user can click on buttons to activate the various inference rules in order to prove the conclusion from the premises. Coming back to the example mentioned above this idea can be illustrated in the following manner using the existing Syllog tool:

New				
Trans	Subst	Contra	Mut	Ex
Invalid	Valid			

Some students are fathers.
 All students are females.
 Ergo:
 Some females are fathers.

Here you may check if this conclusion is provable from the two premises!

Some females are fathers
 Proof:
 Some students are fathers
 All students are females
 Some fathers are students
 Some fathers are females
 Some females are fathers

Correct! This syllogism is valid.
 Your score is 1 of 1

Click on New to continue!

Fig. 1: An example of using the five inference rules in order to prove a syllogism. The system is available online (see <http://syllog.emergence.dk/test8/>).

One weakness in the system illustrated in Fig. 1 is that the uses of the rules of inference are not shown explicitly on the screen. We prefer the following format:

Some females are fathers

Proof:

1. Some students are fathers (premise)
2. All students are females (premise)
3. Some fathers are students (from 1 by MUT)
4. Some fathers are females (from 2 and 3 by SUBST)
5. Some females are fathers (from 4 by MUT; QED)

Here it is clearly stated how the proof is carried out, i.e., how the rules of inference are used step by step.

With the new tool the user will be led to the proof system if the randomly generated syllogism is valid – as soon as the user has been given the opportunity to give an immediate evaluation of the validity of the syllogism. When the user starts proving the syllogism, the system will keep track of the uses of inference rules and also mark the end of the proof with a “QED” as shown above.

5. Explaining the invalidity of invalid syllogisms

Let us assume that the system has randomly generated the following syllogism:

Some parents are doctors.

All doctors are adults.

Ergo: Some adults are not parents.

We also assume that the user has evaluated it, and that it has been concluded that it is invalid. The user of the new tool should then explain why it is invalid. In order to give such an explanation we have to present a universe, a state of affairs, according to which both premises of the above argument are true, whereas the conclusion of the argument is false. This universe or state of affairs may be presented in terms of a modern version of a so-called Venn diagram. This idea goes back to the logician John Venn (1834-1923), who suggested that the meaning of the propositions in syllogistic reasoning can be represented geometrically.

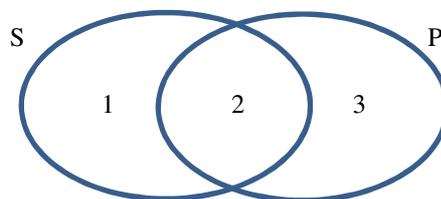


Fig. 2. John Venn’s representation of basic propositions in syllogistics, a: All S are P (“1” empty), i: Some S are P (“2” empty), e: No S are P (“2” non-empty), o: Some S are not P (“1” non-empty).

Venn himself used shading in order to indicate the status of a subset in the diagram

(see Shin et al. 2014, p. 9). We prefer the use of “-” and “+” to indicate that a subset is empty or non-empty, respectively. If the status is unknown it will be indicated by a “?” in the subset in question. This means that the Venn-diagram corresponding to a classical syllogism with three concepts looks as shown in Fig. 3 before the user’s analysis.

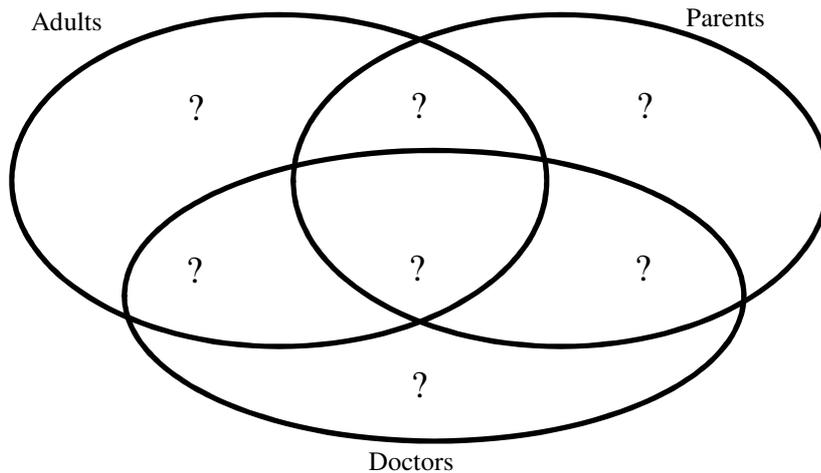


Fig. 3. A version of a Venn-diagram included in the new tool. The user may add some indications (“+” or “-”) in order to demonstrate that a certain syllogism is invalid since it may have true premises but false conclusion.

John Venn originally constructed his diagrams based on a criticism of the so-called Euler Circles. He summarized his criticism in the following way:

The weak point in this [Euler diagrams], and in all similar schemes, consists in the fact that they only illustrate in strictness the actual relation of classes to each other, rather than the imperfect knowledge of these relations which we may possess, or may wish to convey by means of the proposition. (John Venn: “Symbolic Logic”, London: Macmillan 1881, p. 510, quoted from Shin et al., p. 9)

Apparently, John Venn point was that in many cases we only have “imperfect knowledge”, and the question we have to answer is whether or not this knowledge is sufficient to draw the conclusion we consider. It seems that our use of “?” in the diagrams reflects John Venn’s ideas regarding the imperfect knowledge in a nice manner.

We may use the present example as an illustration. In this case the user has to substitute some of the seven “?” in Fig. 3 above with either “+” or “-” to indicate the existence or the non-existence elements in the subsets in question. In Fig. 4 below the user has correctly indicated how the diagram may be marked in order to make sure

that the two premises are both true, whereas the conclusion is false. This shows that the syllogism is invalid – even we don't assume a perfect knowledge but leave some occurrences of “?” in the diagram.

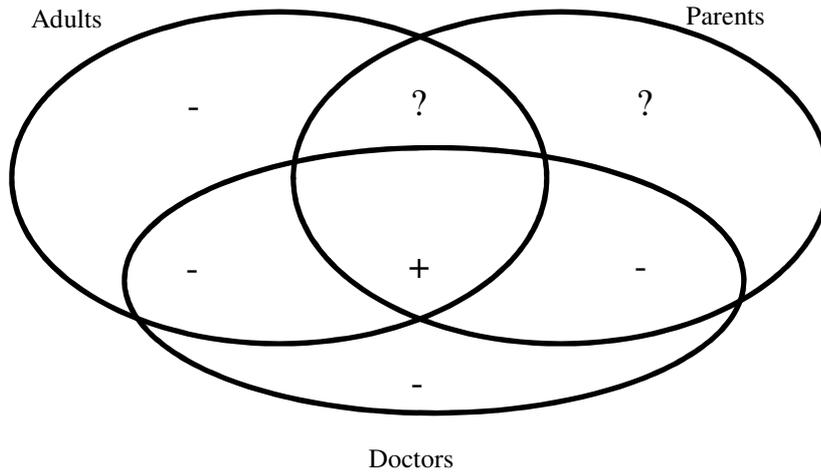


Fig. 4. A version of a Venn-diagram included in the new Syllog tool.

6. The overall design of the new Syllog tool

The overall design of the new Syllog tool can be illustrated in the following manner:

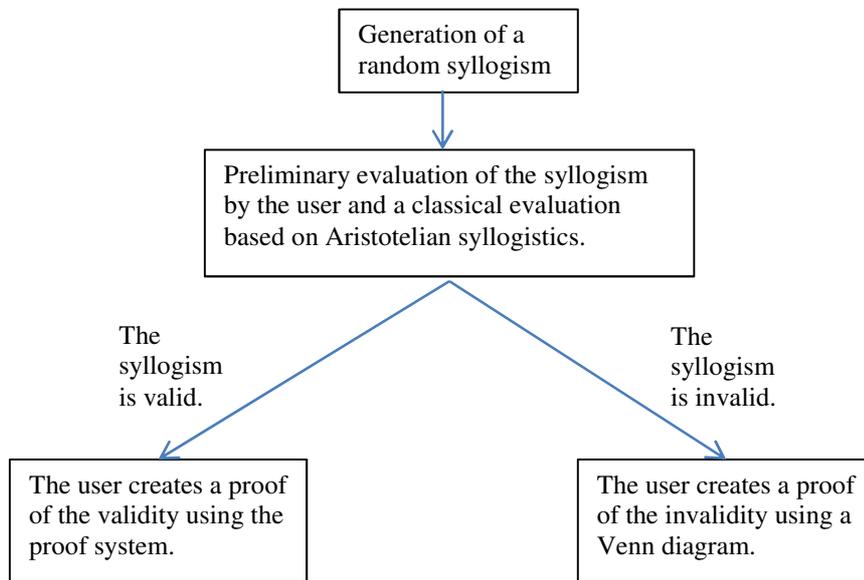


Fig. 5. The overall design of the new Syllog tool.

It seems obvious to implement the new tool in PROLOG+CG (see Kabbaj et al. 2000 & 2001, Petersen 2006 and Sandborg-Petersen 2013-2015). Thereby we may take advantage of the log-functionality of PROLOG+CG in order to establish relevant analytics regarding the practical use of the new tool.

The new tool should of course be evaluated. This may be done using the earlier Syllog tools by which the ability to do syllogistic reasoning can be measured. One group of students should be taught using the new tool presented in this paper, whereas a control group should be taught without the use of this new tool. In both cases the abilities to do syllogistic reasoning should be measured before and after the course.

References

- Aristotle: Prior analytics (1994-2000), translated by A.J. Jenkinson. The Internet Classics Archive, <http://classics.mit.edu/Aristotle/prior.html>
- Kabbaj, A., Janta-Polczynski, M.: From Prolog++ to Prolog+CG: A CG object-oriented logic programming language. In: Ganter, B., Mineau, G. (eds.) Proceedings of ICCS 2000. Lecture Notes in Artificial Intelligence (LNAI), vol. 1867, pp. 540–54. Springer Verlag (2000)
- Kabbaj, A., Moulin, B., Gancet, J., Nadeau, D., Rouleau, O.: Uses, improvements, and extensions of Prolog+CG: Case studies. In: Delugach, H., Stumme, G. (eds.) Conceptual Structures: 9th International Conference on Conceptual Structures. Lecture Notes in Artificial Intelligence (LNAI), vol. 2120, pp. 346–59. Springer Verlag (2001)
- Øhrstrøm, P., Sandborg-Petersen, U., Ploug, T.: Syllog – a tool for logic teaching. In: Proceedings of Artificial Intelligence Workshops 2010 (AIW 2010). pp. 42–55. Mimos Berhad (2010)
- Øhrstrøm, P., Sandborg-Petersen, U., Thorvaldsen, S., Ploug, T.: Classical syllogisms in logic teaching. In: Proceedings of ICCS 2013. Lecture Notes in Artificial Intelligence (LNAI), vol. 7735, pp. 31–43 (2013)
- Øhrstrøm, P., Sandborg-Petersen, U., Thorvaldsen, S., Ploug, T.: Teaching logic through web-based and gamified quizzing of formal arguments. In: Scaling up Learning for Sustained Impact, EC-TEL 2013. LNCS, vol. 8095, pp. 410–23. Springer Publishing Company (2013)
- Øhrstrøm, P., Sandborg-Petersen, U., Thorvaldsen, S., Ploug, T.: Teaching syllogistics using conceptual graphs. In: Hernandez, N.e.a. (ed.) Proceedings of ICCS 2014. Lecture Notes in Artificial Intelligence (LNAI), vol. 8577, pp. 217–230. Springer (2014)
- Øhrstrøm, P., Sandborg-Petersen, U., Thorvaldsen, S., Ploug, T.: Teaching syllogistics through gamification and interactive proofs. In: Conole, G., Klobučar, T., Rensing, C., Konert, J., Lavoué, E. (eds.) Design for Teaching and Learning in a Networked World: Proceedings of EC-TEL 2015. Lecture Notes in Computer Science (LNCS), vol. 9307, pp. 609–612 (2015).
- Øhrstrøm, P., Sandborg-Petersen, U., Thorvaldsen, S., Ploug, T.: Teaching Syllogistics Using E-learning Tools (ed.) Proceedings of Eleot 2016. Forthcoming.
- Parry, W., Hacker, E.: Aristotelian Logic. State University of New York Press (1991)
- Petersen, U., Schärfe, H., Øhrstrøm, P.: Online course in knowledge representation using conceptual graphs (2001-2005), Aalborg University, <http://cg.huminf.aau.dk/>
- Petersen, U., Prolog+CG: a maintainer’s perspective. In: de Moor, A., Polovina, S., Delugach, H. (eds.) First Conceptual Structures Interoperability Workshop (CS-TIW 2006). Proceedings. Aalborg University Press (2006)

- Sandborg-Petersen, U., Øhrstrøm, P.: The syllog code (2013-2015), Open Source Software available from <http://syllog.sourceforge.net/>
- Sandborg-Petersen, U., Øhrstrøm, P., Ploug, T., Thorvaldsen, S.: Two experiments using syllog, <http://syllog.emergence.dk/2015/>
- Shin, Sun-Joo; Lemon, Oliver and Mumm, John, Diagrams, Stanford Encyclopedia of Philosophy, <http://plato.stanford.edu/archives/win2014/entries/diagrams>, 2014.