

Open Research Online

The Open University's repository of research publications and other research outputs

Using similarity metrics for matching lifelong learners

Conference or Workshop Item

How to cite:

Van Labeke, Nicolas; Poulouvassilis, Alexandra and Magoulas, George (2008). Using similarity metrics for matching lifelong learners. In: ITS'08 - 9th International Conference on Intelligent Tutoring Systems, 23-27 Jun 2008, Montreal, Canada, Springer, pp. 142-151.

For guidance on citations see [FAQs](#).

© 2008 Springer

Version: Accepted Manuscript

Link(s) to article on publisher's website:

http://dx.doi.org/doi:10.1007/978-3-540-69132-7_19

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's data [policy](#) on reuse of materials please consult the policies page.

oro.open.ac.uk

Using Similarity Metrics for Matching Lifelong Learners

Nicolas Van Labeke, Alexandra Poulouvassilis, and George Magoulas

London Knowledge Lab,
Birkbeck, University of London,
23-29 Emerald Street
London WC1N 3QS - United Kingdom
email: {nicolas, ap, gmagoulas}@dcs.bbk.ac.uk
<http://www.lkl.ac.uk/research/myplan/>

Abstract. The L4All system provides an environment for the lifelong learner to access information about courses, personal development plans, recommendation of learning pathways, personalised support for planning of learning, and reflecting on learning. Designed as a web-based application, it offers lifelong learners the possibility to define and share their own timeline (a chronological record of their relevant life episodes) in order to foster collaborative elaboration of future goals and aspirations. A keystone for delivering such functionalities is the possibility for learner to search for ‘people like me’. Addressing the fact that such a definition of ‘people like me’ is ambiguous and subjective, this paper explores the use of similarity metrics as a flexible mechanism for comparing and ranking lifelong learners’ timelines.

1 Introduction

Supporting the demands of lifelong learners is increasingly considered at the core of the learning and teaching strategy of HE and FE institutions and poses new challenges, such as enabling better support for lifelong learners and facilities for accessing cross-institutional resources. To address these challenges, it is important to exploit further the advantages of Information and Communications Technology networks to enable better support for planning lifelong learning and ubiquitous access to lifelong learning facilities from home, the workplace and educational organisations. This new trend to educational services has led to research and development that involves the provision of new learner-centred models of organising and delivering educational resources (see for example the integrated framework proposed in [1, 2]).

The L4All system [3, 4] provides an environment for the lifelong learner to access information about courses, personal development plans, recommendation of learning pathways, personalised support for planning of learning, and reflecting on learning. The MyPlan project follows on from the initial L4All pilot project and aims to develop, deploy and evaluate personalised functionalities for the creation, searching and recommendation of learning pathways. This will enhance

individual learners' engagement with the lifelong learning process by offering personalised levels of learner control over their learning pathways, personalised support in the reflection of where their learning activities may take them, and management of their personal record of progress and attainment. It will also support building communities of learners with similar interests, and information sharing with other members of the community, other users of the L4All system, and HE/FE institutions. Figure 1 shows the main page of the L4All system.

At the core of L4All is the specification of a User Model that addresses the specificities of lifelong learners and is based on the notion of learning 'trails' [5]. In the context of L4All, a 'trail' is a *timeline*-based representation of learners' work, learning and other life experiences that provides a holistic approach and continuity between their learning episodes and work experiences.

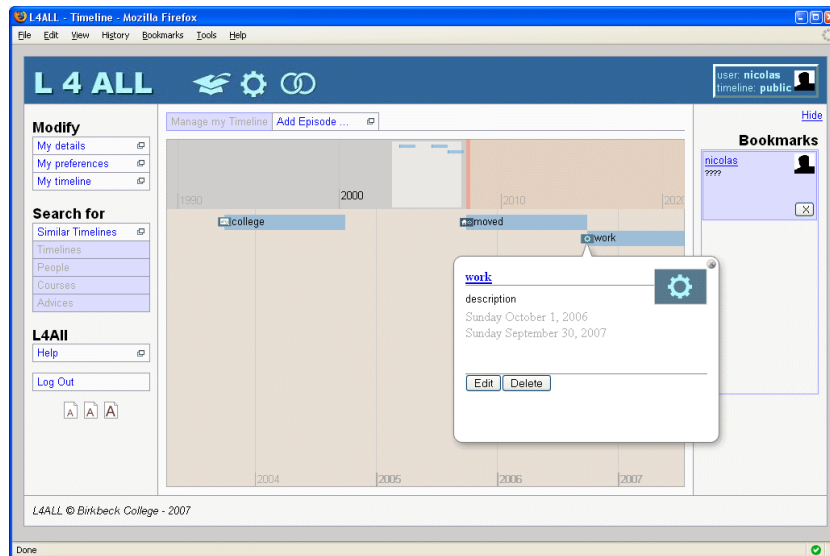


Fig. 1. The main page of L4All with the user's timeline (*centre*), access to the various functionalities (*left*) and a bookmarks section for networking (*right*).

One requirement for offering such personalised services is to provide learners with the possibility to search for 'people like me', i.e. to exploit the full structure and content of their profiles and timelines in order to find similar matches that will foster collaborative elaboration of future goals and aspirations. This poses some interesting challenges that this paper addresses. First, the structure of the timeline, as a sequence of temporal records, is potentially of such complexity that it does not immediately suggest a natural way of enabling comparisons between timelines. Second the notion of similarity of timelines is vague and subjective, and it is not clear which aspects of that complex structure should be considered and how they should be compared. Third, assuming that a personalised search

and similarity-ranking of timelines can be designed and developed, supporting learners in exploiting such functionalities is an open problem.

This paper presents an investigation of these challenges and is organised as follows. First, we review the User Model underlying the L4All system, in particular the way timelines are represented. Second, a flexible mechanism for encoding the timeline in a form suitable for similarity matching is presented. Third, several algorithms for similarity measures of timelines are compared and analysed from the point of view of their behaviour in identifying key aspects of timeline comparisons. Fourth, we describe the user interface for the personalised construction of a new ‘people like me’ search, and for the visualisation and exploration of the timelines returned. The paper concludes by addressing some of the issues arising from our work and proposes some future developments.

2 User Model and Timelines

The L4All User Model [6] is comprised of three parts:

1. The *User Profile* contains personal information about learners such as their name, gender, year of birth, email, login name and password.
2. The *Learning Profile* contains information about the educational and professional background of learners (such as current occupation, highest qualification and skills) and information about their learning needs (such as willingness to travel, current learning goal, preferred mode of learning – part-time, full-time – and preferred learning methods – in groups, alone, online).
3. The *Timeline* is the novel part of the User Model, specifically addressing the particularities of lifelong learners. It represents the learning – and, more generally, life – pathway of the learners to date and contains a chronological record of those episodes of their life that they deem significant to their personal development.

Episodes in a timeline are identified by their category, selected from 20+ categories currently supported by the system. They include personal episodes, e.g. relocation, travel abroad, illness, marriage, death in family, etc., occupational episodes, e.g. started work, set up business, retired, did voluntary work, etc. and educational episodes, e.g. attended college, university or school, attended courses, etc. Each episode is specified by a start date and a duration (if applicable), title, description, keywords and an optional URL.

In order to extend the descriptive power of the timeline, some of the most significant episodes are also further elaborated by one or two further attributes, referred to as primary and secondary classifications: educational episodes by a subject and a qualification level; work episodes by an industry sector and a position; and business episodes by an industry sector. These additional attributes are populated by a specific tree-like taxonomy of values selected from relevant British standards¹. The structure and identifiers of these taxonomies have been

¹ The Standard Industrial Classification (SIC), the Standard Occupational Classification (SOC), the National Qualification Framework (NQF) and the Labour

maintained but, for usability purposes, their depth is limited to four levels. Elements in each taxonomy can therefore be represented by four-digit identifier, each digit uniquely identifying its precise position in the tree.

3 Similarity Measures of Timelines

The initial prototype of the L4All system supported several search functionalities over users and their timelines. Two limitations of this approach were identified during the first piloting phase [4]. First, all the search functionalities were keyword-based, targeting the various fields of the User Profile, Learning Profile and Timelines, and therefore limited in their scope. In particular, searching on timelines returns matches based solely on the occurrence of the keywords present in one or several episodes but cannot exploit the overall structure of the timeline. Second, the results of any search were not personalised according to the particular user performing the search. An alternative approach was needed, that could take into account both these issues: in other words, some form of comparison or similarity measure between a user’s timeline and the rest of the timelines in the L4All repository.

String metrics offer such a possibility. String metrics (also referred to as similarity metrics) have been widely used in information integration and in several fields of applied computer science [7, 8]. In the context of Intelligent Tutoring Systems, similarity metrics have been used in the REDEEM system [9] to compare alternative sequences of instructional activities as produced by authors. In the context for the L4All timelines, the main requirement for using similarity metrics is to encode a time-based sequence of records into a token-based string. For this purpose, we have made four simplifying assumptions at the outset (the implications of these assumptions for users will be explored in our forthcoming evaluation activities):

The precise duration and dates of an episode have no particular significance. This may seem strange for a time-dependent data structure but the relevance and usage of such information for searching for ‘people like me’ is ambiguous. Should we consider two learners having done the same university degree but at different dates similar or not? Should we consider them more different if one of them has taken twice as long as the other (being part-time for example)? Or is it enough, at some level, to consider them similar since both of them have done this particular degree? In the absence of evidence supporting one point of view against the other, we decided, initially, to ignore this information. Only each episode’s relative time-stamp (i.e. its position in time compared to the other episodes in the timeline) is used in order to ‘linearise’ the timeline by ordering the episodes in chronological order.

Gaps between episodes have no particular significance unless explicitly expressed as an episode. The problem posed by gaps in timelines is the lack of

Force Survey’s Subject of Degree (SBJ). See the Labour Force Survey User Guide http://www.statistics.gov.uk/downloads/theme_labour/Vol15.pdf.

explicit explanation for their occurrence and therefore for their significance for the timeline. Again, in the absence of such information, they are ignored.

Some categories of episode may have no role to play in defining ‘people like me’. The purpose of a timeline is for learners to record every episode of their background that may have an impact on their learning pathways. For example, personal episodes such as marriage, illness, relocation, etc. are important as they may have a clear influence on the decisions made for personal development (e.g. a course at a particular learning institution may have been followed because of a relocation to a particular city). However, this does not necessarily mean that such episodes are a prerequisite or a necessary condition for reaching a particular stage in someone else’s development. Their importance while searching for role models, inspiration, or ‘people like me’ are therefore ambiguous and subjective. Therefore, whether to include or not particular categories of episode in the similarity matching should be left to the user to specify.

The exact classification of an episode may not be significant in defining ‘people like me’. As described earlier, some of the most important episodes in the timeline (educational and work-related episodes) use specific attributes to precisely describe their instance, e.g. working as a researcher in computer science. However, taking such a fine-grained description of episode may not be useful in searching for ‘people like me’, as it may make more sense to consider that a researcher (without a precise field) is someone to consider ‘like me’. Therefore the level of specialisation of episodes should also be left to the user to specify.

Using these assumptions, it is now relatively straightforward to generate a token-based string representing the timeline. Each episode of the timeline is encoded as a string token composed of a two-letter unique identifier of the category of the episode (e.g. C1 for a College episode, Wk for a Work episode) and two four-digit codes classifying the exact instance of this episode (as described in the previous section). Note that, in order to maintain a consistent pattern for the token’s encoding, nonexistent or unspecified classifications are encoded as 0.0.0.0.

Combining the two first assumptions above means that no time information is used to encode episodes, only their relative position matters². Filters are then applied to the string of tokens to remove the episodes that should not be considered in the current similarity search, as well as for limiting the depth of their classification. In the latter case, the use of the coding system for the classification facilitates the process: digits below the specified depth are replaced by 0, replacing the specific classification by a more general parent.

4 Comparison of Similarity Measures

The metrics used in this study are part of the `SimMetrics`³ JAVA package, an open source extensible library of metrics that provides real number-based similarity measures between strings, allowing both normalised and un-normalised

² With an arbitrary decision as to their ordering if multiple episodes coincide in time.

³ `SimMetrics`, see <http://www.dcs.shef.ac.uk/~sam/stringmetrics.html>.

output. The `SimMetrics` package contains about 20 different metrics, some of them customisable by using user-defined cost functions and tokenisers. Not all metrics can be used in our context, since some are tailored for working on a particular application domain (linguistic for example) and require strings that are incompatible with our encoding of timelines. We refer the reader to the package documentation for descriptions of each metric.

Table 1 shows a set of synthetic timelines used in our comparison study. They are deliberately simplistic in their structure, as the purpose of this comparison is to identify general trends arising from the various similarity metrics, rather than evaluating their intrinsic power of discrimination.

Table 1. List of encoded timelines used for the metrics comparison

Ref.	Description	Encoding
<i>Source</i>	Timeline used as the source for the similarity measure	C100 Un00 Mv00 Wk00
<i>ID</i>	Timeline identical to <i>Source</i> .	C100 Un00 Mv00 Wk00
<i>RE</i>	Timeline containing the same episodes as <i>Source</i> but in a totally different order.	Un00 Wk00 C100 Mv00
<i>AD_w</i>	New episode (similar to an existing one) added to <i>Source</i> .	C100 Un00 Mv00 Wk00 Wk00
<i>AD_e</i>	New episode (different from all existing) added to <i>Source</i> .	C100 Un00 Mv00 Wk00 Bs00
<i>RM_w</i>	Last episode removed from <i>Source</i> .	C100 Un00 Mv00
<i>RM_u</i>	One episode removed from <i>Source</i> .	C100 Mv00 Wk00
<i>SB_e</i>	One episode of <i>Source</i> substituted by a new one (different from all existing ones).	C100 Un00 Mv00 Bs00
<i>SB_u</i>	One episode of <i>Source</i> substituted by an existing episode.	C100 Un00 Mv00 Un00
<i>SB_w</i>	One episode of <i>Source</i> substituted by a variant of an existing episode (a different classification).	C100 Un00 Mv00 Wk10

The *Source* timeline is a string of four episodes of different type: college (C100), university (Un00), move (Mv00) and work (Wk00). Each episode has been encoded as a token, using the scheme described in the previous section. For the sake of clarity, and since this comparison does not rely on the full power of discrimination of the scheme, the episode classifications have been reduced to a single digit each (i.e. representing 0.0.0.0 as 0).

The target timelines represent a variety of alterations of the *Source* timeline that could occur in real-life situations: a totally similar timeline (i.e. the same sequence of episodes), a reordered timeline (i.e. the same episodes but totally reordered), adding an extra episode, removing an existing episode, substituting an episode by another one. Note that the set of target timelines listed in the table only represent the most representative of each group. In order to test the behaviour and consistency of the metrics, all possible combinations were generated for each group (e.g. timelines representing the addition of a new episode were generated considering every possible position in the *Source* timeline).

Table 2 summarises the results of the different similarity measures applied to every target timeline. The values shown in the table do not represent the distance between the two strings but their normalised similarity, i.e. the ratio between the calculated distance and the maximum distance. As mentioned earlier, the main aim of this comparison is not to focus on individual measures for assessing their accuracy but to extract general conclusions regarding their behaviour when confronted with particular configurations. From these results, several conclusion can be drawn. First, all the similarity measures are indeed able to recognise complete similarity between timelines (as indicated by all 1 in the *ID* column). More interestingly, three groups of metrics emerge, as listed in Table 2.

Table 2. Normalised similarity between the source and the target timelines.

	<i>ID</i>	<i>RE</i>	<i>AD_w</i>	<i>AD_e</i>	<i>RM_w</i>	<i>RM_u</i>	<i>SB_e</i>	<i>SB_u</i>	<i>SB_w</i>
Levenshtein	1	0	0.8	0.8	0.75	0.75	0.75	0.75	0.75
Needleman - Wunsch	1	0	0.8	0.8	0.75	0.75	0.75	0.75	0.88
Jaro	1	0.72	0.93	0.93	0.92	0.92	0.83	0.83	0.83
Matching Coefficient	1	1	0.8	0.8	0.75	0.75	0.75	0.75	0.75
Euclidean Distance	1	1	0.84	0.84	0.8	0.8	0.75	0.75	0.75
Block Distance	1	1	0.89	0.89	0.86	0.86	0.75	0.75	0.75
Jaccard Similarity	1	1	1	0.8	0.75	0.75	0.6	0.75	0.6
Cosine Similarity	1	1	1	0.89	0.87	0.87	0.75	0.87	0.75
Dice Similarity	1	1	1	0.89	0.86	0.86	0.75	0.86	0.75
Overlap Coefficient	1	1	1	1	1	1	0.75	1	0.75

The first group includes transformation-based metrics like Levenshtein, Jaro and Needleman-Wunsch that are able to discriminate between the basic operations of string manipulation (copy, substitution, addition, deletion). The non-zero result for the Jaro distance in the *RE* column can be explained by a threshold used for determining matching tokens (see the documentation of this metric); our test strings are not long enough (only four tokens) to allow proper discrimination. All these metrics do not take into consideration the position of the token involved in one of the string manipulations (whatever the location of the added or substituted episode, the scores are the same). The only exception is the Needleman - Wunsch distance, which gives a different score when a variant of the initial episode (i.e. same category but different classification) is substituted (score of 0.88 in *SB_w*, instead of 0.75 in *SB_e* and *SB_u*). This is due to the use of specific gap cost and distance functions that can be tailored to the particular nature of the data involved in the similarity measure and therefore could be adjusted for our particular use of the timelines (see Section 6).

The second group of metrics includes vector-based metrics such as Block Distance, Euclidean Distance and Matching Coefficient that are not able to discriminate between re-ordered strings, as indicated by 1 in the *RE* column. Whatever the order of the tokens in the string, both source and target are considered to be identical since they contain the same set of tokens. As with the

metrics in the previous group, the results for addition, substitution and removal of tokens are position-independent.

The third group of metrics includes the rest of the vector-based metrics (Jaccard, Cosine, Dice Similarities and Overlap Coefficient) which, as with the previous group, do not discriminate between reordering of tokens. Moreover, this group also fails to take into account the duplication of tokens in the string, as exemplified by the scores of 1 in the AD_w column (i.e. adding an episode that is already existing in the timeline) or the different scores for the SB_u column (i.e. substituting an episode with one that is already existing, resulting in fact in the deletion of this episode). Once again, this is because of the set-based algorithms used for these metrics, in particular the use of intersection/union procedures rather than summation as in the previous group. This is also reflected by the fact that substitution also depends on the nature of the episode substituted (the SB_u column give scores different from the other substitutions). In this group, the Overlap Coefficient is an extreme case, as it basically measures whether the source string is a subset of the target one (or the converse).

5 Searching for ‘People Like Me’

What the comparison above shows is that different similarity metrics offer different degrees of support for the basic operations of string manipulation: copy, substitution, addition or deletion of a token. The important point here is that the comparison does not highlight one particular metric as being more useful or accurate for our purpose, precisely because our purpose (or, rather, the user’s) is unknown. The assumptions made in Section 3 encompass a wide range of users’ behaviour regarding the way they understand a ‘people like me’ functionality.

In order to validate these assumptions, a dedicated interface for such searches was therefore designed and implemented. It provides users with a three-step process for specifying their own definition of ‘people like me’. The first step of a user’s query specifies those attributes of the user’s profile that should be matched with other users’ profiles (age, qualification, location, etc.) and act as a filtering of the possible candidates before application of the similarity comparison on the timelines. The second step of the query specifies which part(s) of the timeline should be taken into account for the similarity comparison (currently by selecting the appropriate categories of episode). The final step specifies the nature of the similarity measure to be used (i.e depth of episode classification and metric). Once a definition of ‘people like me’ has been specified by the user, the search returns a list of all candidate timelines, ranked by relevance (i.e. their normalised similarity measure). The user now have the possibility to access any returned timelines and explore them.

This first approach to offering a ‘people like me’ functionality has given us the possibility to accumulate information about usage and expectations from users. It has offered us some insight into the context and relevance of particular configurations and how specific aims – such as looking for aspirational timelines or learning recommendations – could be supported. These issues and propos-

als for personalised support for the variety of activities they highlight will be investigated further in future work.

6 Discussion and Future Work

Lifelong learning requires technology to be used effectively to support learners in becoming more aware of their own learning and help them with planning of their learning throughout life under varying circumstance and settings. In this context, it is important to support user engagement and participation in lifelong learning and facilitate collaboration among lifelong learners for community building. In this paper we have discussed how string similarity measures could be used to encode and compare the timelines of lifelong learners. We have shown that existing metrics behave differently in identifying key aspects of timeline comparison, such as addition, substitution or deletion of episodes. Since the precise definition of what is a similar timeline is ambiguous and subjective, we have designed a new user interface for L4All such that learners can specify their own definition of ‘people like me’, offering them the possibility to decide which aspects of a timeline need to be considered or not for the matching. Evaluating the soundness and acceptability of this approach for users – as well as the usability of our user interface – are currently under evaluation. In the first evaluation phase – underway at the time of writing – we will be asking learner participants from three different learning institutions to explore the definition and the results of applying different similarity definitions on a predefined database of synthetic timelines. In the long term, several issues arising from our work will also be explored.

The encoding of timelines and episodes for similarity computation may need to be improved, in particular in determining by how much two episodes are similar. One way of dealing with this issue is by using the depth-adjusting encoding of episodes, where specific classification identifiers can be relaxed to one of their more general parents in the hierarchy, thus increasing the chance for two episodes to be compared as identical. But by doing so we are not only losing the descriptive power of episodes but also uniformly applying the filtering on all episodes in the timeline. An alternative, unfortunately only supported by distance metrics such as Levenshtein or Needleman-Wunsch, is to incorporate user-defined distance and gap cost functions, i.e. specifying a fine-grained analysis of the distance between two given tokens and of the cost of adding or removing a token in a string. Instead of the current binary comparison of episodes (i.e. their encodings are syntactically equal or not), we could adjust the distance between two similar episodes by the distance between their classifications (i.e. the sum of the distances to the closest common ancestor of each classification’s element).

Similarly, our first two assumptions in Section 3 are clearly the most critical. The ongoing piloting of our techniques will certainly provide us with insights about the importance or not of taking temporal information into account. Extensions of our token-based encoding of timelines or even a specific similarity mechanism that maintains temporal tags will have to be considered.

Finally, a further important issue we still need to address is the question of providing lifelong learners with support for exploiting the results of a similarity search. Currently, we are relying on a pure visualisation approach, by displaying both the learner's own timeline and similar timelines returned by the search. A specifically designed dynamic widget is used to allow the learner to scroll back and forward across each timeline, to access individual episodes, etc. Such an interactive visualisation of timelines is certainly helping learners to explore alternative timelines and is supporting them in elaborating future goals and aspirations, but more proactive supports will also be investigated. To enable the provision of feedback and on-demand support necessitates the ability to identify the reasons for a search considering two timelines as being similar. Again, metrics such as Needleman-Wunsch offer the possibility for such an identification by enabling backtracking of the distance computation and determining potential sequence alignments, i.e. the ability to identify alignment between pairs of tokens in matching strings.

Acknowledgments. The MyPlan – Personal Planning for Learning Through Life project (<http://www.lkl.ac.uk/research/myplan/>) is funded by the e-Learning Capital Programme of the Joint Information System Committee, UK.

References

1. Koper, R., Tattersall, C.: New directions for lifelong learning using network technologies. *British Journal of Educational Technology* **35**(6) (2004) 689–700
2. Koper, R., Giesbers, B., van Rosmalen, P., Sloep, P., van Bruggen, J., Tattersall, C., Vogten, H., Brouns, F.: A design model for lifelong learning networks. *Interactive Learning Environments* **13**(1–2) (2005) 71–92
3. de Freitas, S., Magoulas, G., Oliver, M., Papamarkos, G., Harrison, A.P.I., Mee, A.: L4all - a web-service based system for lifelong learners. In: *Proceedings of eChallenges'2006 (Workshop on Next Generation in Technology Enhanced Learning)*. (2006) 1477–1484
4. de Freitas, S., Harrison, I., Magoulas, G., Mee, A., Mohamad, F., Oliver, M., Papamarkos, G., Poulouvasilis, A.: The development of a system for supporting the lifelong learner. *British Journal of Educational Technology* **37**(6) (2006) 867–880
5. Peterson, D., Levene, M.: Trail records and navigational learning. *London Review of Education* **1**(3) (2003) 207–216
6. Baajour, H., Magoulas, G., Poulouvasilis, A.: Modelling the lifelong learner in a services-based environment. In: *Proceedings of ITA'07 – 2nd International Conference on Internet Technologies and Applications*. (2007) 181–190
7. Gusfield, D.: *Algorithms on Strings, Trees, and Sequences - Computer Science and Computational Biology*. Cambridge University Press (1997)
8. Cohen, W.W., Ravikumar, P., Fienberg, S.E.: A comparison of string distance metrics for name-matching tasks. In: *Proceedings of IIWeb'03 – IJCAI Workshop on Information Integration on the Web*. (2003) 73–78
9. Ainsworth, S., Clarke, D., Gaizauskas, R.J.: Using edit distance algorithms to compare alternative approaches to its authoring. In: *Proceedings of ITS'02 – 6th International Conference on Intelligent Tutoring Systems*. Volume 2363 of *Lecture Notes in Computer Science*, Springer (2002) 873–882