Studies on Editing Patterns in Large-scale Wikis

Philip Boulain, Nigel Shadbolt and Nicholas Gibbins

1 Introduction

In this chapter, we present a complementary pair of experiments on the way users edit Wikipedia. First, we look at the macro scale, quantitatively categorising the types of edits users made. Then, we look at the micro scale, qualitatively studying the rationale and process of individual editors. These experiments form part of a broader project looking into the potentially beneficial relationships between open hypermedia, the study of interconnected documents; Semantic Web, the study of interconnectable data; and 'wikis', web-based communal editing systems.

Wikipedia is a communally-edited encyclopædia with over two and half million articles in the English version. Each article is a document with prose about an encyclopædic subject, usually supplemented with illustrations. Almost all articles are placed into at least one ad-hoc category, and linking between articles is common.

Hypermedia is a long-standing field of research into the ways in which documents can expand beyond the limitations of paper, generally in terms of greater cross-referencing and composition (reuse) capability. Bush's *As We May Think* [3] introduces the hypothetical early hypertext machine, the 'memex', and defines the "essential feature" of it as "the process of tying two items together". This *linking* between documents is the common feature of hypertext systems, upon which other improvements are built.

As well as simple binary (two endpoint) links, hypertext systems have been developed with features including n-ary links (multiple documents linked to multiple other documents), typed links (links which indicate something about *why* or *how* documents are related), generic links (links whose endpoints are determined by matching criteria of the document content, such as particular words), and composite documents, which are formed by combining a set of other, linked, documents. Open

Philip Boulain · Nigel Shadbolt · Nicholas Gibbins
Intelligence, Agents, Multimedia Group, School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, United Kingdom e-mail: {prb,nrs,nmg}@ecs.soton.ac.uk

Hypermedia extends this with interoperation, both with other hypermedia systems and users, and with non-hypermedia resources. A key concept in open hypermedia is that of the *non-embedded* link—links (and anchors) which are held external to the documents they connect. These allow links to be made to immutable documents, and to be added and removed in sets, often termed 'linkbases'. One of the earliest projects attempting to implement globally-distributed hypertext was Xanadu [9], a distinctive feature of the design of which was *transclusion*: including (sections of) a document into another by reference.

In related work, we are currently investigating the relationship between an exemplar semantic wiki, Semantic MediaWiki [7], and open hypermedia systems, as defined by the Dexter Hypertext Reference Model [6]. Our preliminary results based on a formal description of Semantic MediaWiki in terms of the Dexter model suggest that such semantic wikis can be treated as simple open hypermedia systems. While details are beyond the scope of this paper, some basic parallels are evident: a wiki node is akin to a hypermedia document, and a semantic web resource. Semantic wikis generally treat typed inter-node links as RDF statements relating the nodes, and these links are embedded and binary in hypermedia terms. From this we can see a meaningful similarity between a graph of documents connected by typed links, and a graph of resources connected by RDF statements. We can also see that wikis do not have features covering more advanced hypermedia links: such as those which are not embedded, or have more than two endpoints.

This then suggests that semantic wikis stand to gain from techniques developed within hypermedia, but we must first judge if there is any substantial cost to be reduced, hence these experiments. We found that twice as many edits changed links alone, not affecting the article text, and that edits which maintained manual indexes of pages constituted approximately a tenth of total edits. We also discovered that content re-use was not as desirable as hypermedia research has assumed, but that automatic linking and transclusion could still address problems with current technology.

2 Macro-scale experiment

We carried out an experiment to estimate the proportion of effort expended maintaining the infrastructure around data, rather than the data itself, on a weak hypertext wiki system. We define a 'weak' hypertext system here as one whose feature set is limited to embedded, unidirectional, binary links, as with the World Wide Web. Our hypothesis is that the manual editing of link structure, of a type which richer hypertext features could automate, will show to be a significant overhead versus changes to the text content. If supported, this indicates that further work on stronger hypertext wikis is potentially beneficial.

This experiment also seeks to partially recreate a related, informal experiment, discussed in an essay by Swartz [11].

2.1 Dataset

We chose English Wikipedia[1] as the experimental dataset, because it has both a considerably large and varied set of documents, and a complete history of the editing processes—performed by a wide range of Web users—between their first and current versions[2]. The wiki community keep the dataset fairly well inter-linked and categorised for cross-reference, but they do this via the cumulative efforts of a large body of part-time editors. As well as being statistically significant, demonstrating possible improvement of English Wikipedia is socially significant, as it is a widely-used and active resource.

It is important to stress the size of the English Wikipedia dataset. Wikipedia make available 'dumps' of their database in an ad-hoc XML format; because this study is interested in the progression of page contents across revisions, it was necessary to use the largest of these dumps, containing both page full-text and history (unfortunately, also non-encyclopædic pages, such as discussions and user pages). This dump is provided compressed using the highly space-efficient (although time-complex) bzip2 algorithm; even then, it is 84.6GB. The total size of the XML file is estimated to be in the region of two terabytes.

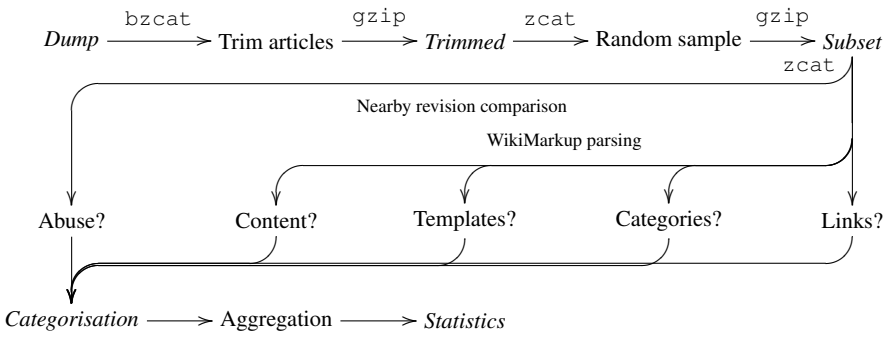

Fig. 1 Data flow of Wikipedia experiment

2.2 Procedure

Figure 1 shows the simplified data flow of the processing of the dump performed for the experiment.

[1] `http://en.wikipedia.org/`

[2] MediaWiki, unlike many wikis, never deletes old revisions of a page.

2.2.1 Reduction

First, we trimmed down the dataset to just those pages which are encyclopædic articles, as these are the pages of greatest significance to the Wikipedia project's goals, and thus the most important to study. Otherwise, the dataset would include a lot of 'noise' in the form of discussion and user pages, which are likely to have different editing patterns, and be less connected to the hyperstructure. The most practical way to do this was to remove any page placed in a namespace. On English Wikipedia, this also has the effect of removing other page types, such as media and image descriptions, help pages copied from MetaWiki, front-page portal components, and templates. As this stage also required decompressing the data, it ran over the course of several days on a multi-processor server.

We took a random subset of the data for processing. Samples of 0.04% and 0.01% of pages (approximately: see the description of the subset tool below; actual page counts 14,215 and 3,589 respectively) were selected, yielding a compressed dataset which would fit on a CD-ROM, and could be processed in a reasonable timeframe. Further iterations of the experiment may study larger subsets of the data.

2.2.2 Parsing

We performed categorisation on the revisions, into several edit types which would be automatically distinguished. In particular, a simple equality comparison between a revision, and the revision two edits previous, can detect the most common (anti-)abuse modification: the rollback, or revert (unfortunately, MediaWiki does not record such operations semantically). A sequence of reverts[3] is usually indicative of an 'edit war', where two users continually undo each-others changes in favour of their own. Page blanking was also easy to detect, but identifying more complicated forms of vandalism (e.g. misinformation, spam) was not feasible—if reliable, automatic detection were possible, they would not be present in the data, as Wikipedia could prevent such changes from being applied. Identifying abuse (and abuse management) of the simpler types is important, as otherwise they would appear as very large changes.

In order to detect changes in the text content, templates used, MediaWiki categories, and links from a page, it was necessary to attempt to parse the MediaWiki markup format. Such 'wikitext', as it is known, is not a formally defined language: there is no grammar for it, and it does not appear likely that an unambiguous grammar actually exists. MediaWiki does not have a parser in the same way as processing tools such as compilers and XML libraries; instead it just has a long and complicated set of text substitution procedures which convert parts of 'wikitext' into display-oriented HTML. These substitutions often interact in a ill-defined manner, generally resulting in either more special-case substitutions, or as being defined as a new, hybrid, feature, which editors then use. Because of these problems, and the

[3] e.g. `http://en.wikipedia.org/w/index.php?title=Anarchism&diff=next&oldid=320139`

lack of abstraction in MediaWiki's 'parser', as much as the programming language boundary, a 'scraping' parser was created which attempted to approximate partial processing of the wikitext format and return *mostly* correct results. This parser is a single-pass state machine (42 states) with a few additional side-effects. This yields excellent performance: testing showed that the time spent parsing is dominated by the time performing decompression.

2.2.3 Text differences

To determine if an edit included a significant ('major') change to the text content, we required a difference metric between the plaintext of the revisions. This metric was then compared to a threshold to classify edits as being content changes or not (in particular, the imperfect parser generates 'noise' from some non-content changes, as it cannot correctly remove all the markup). The default threshold was chosen as 5%: sentences in the English language are generally around twenty words in length, so this considers anything up to changing one word in each sentence as non-major (minor). MediaWiki also allows registered users to explicitly state than an edit is minor; this flag was respected where present.

We chose an approximation of Levenshtein distance[8], as it is a simple measure of insertions, deletions, and substitutions, fitting the kind of edit operations performed on the wiki. However, the algorithm for computing Levenshtein itself was far too time-complex, even with aggressive optimisation, taking two minutes on a tiny test set of just a few thousand revisions of a single page (before trimming away the identical parts at either end of both strings to take advantage of edit locality, this took 45 minutes). The problem was that the matrix-based approach is $O(n \times m)$, where n and m are the string lengths, in all cases: for n and m in the region of 16,000 characters, as found on many revisions, merely iterating through all 256 million matrix cells was prohibitively expensive.

Instead, we developed a new approach to computing such a distance, taking advantage of the domain-specific knowledge that the two strings being compared are likely very similar save for 'local' edits: the difference is likely to be a new paragraph, or a removed sentence, or some changed punctuation. Instead of efficient search within the space of editing operations, as Levenshtein, it is based on the idea of "sliding windows": a pass is made over both strings in parallel; when characters begin to differ, a look-back 'window' is opened between the point at which differences began, and continues until similarity is again found between these windows. At this point, the position through the strings resynchronises, the distance is increased by the offset required, and the windows are again 'closed'. When the end of either string is reached by the far edge of the window, the algorithm can terminate, as any remaining characters in the other string must be unmatched and thus add to the distance. As a result, the algorithm scales with regard to the shorter of the two strings, which is helpful when revisions may add whole paragraphs of new text to the end. To reduce inaccuracy in certain cases, the algorithm maintains a 'processed point' cursor, to avoid double-counting of overlapping insertions and

Algorithm 1 'Sliding window' string distance metric

```
    procedure STRING-DISTANCE(A, B)
        proc ← 0                                   ▷ No. of chars. of string processed
        procstr ← NEITHER                          ▷ Last string aligned upon
        dist ← 0                                   ▷ Difference accumulator
 5:     nearA ← farA ← A                           ▷ Near and far pointers
        nearB ← farB ← B
        Let endA be the beyond-last character of buffer A, and endB beyond B
        procedure SCAN(near, far)
            for scan ← near to before far do
10:             if Chars. at scan and far same then return scan
            return false
        while farA ≠ endA ∧ farB ≠ endB do
            synfarA ← SCAN(nearA, farA)
            synfarB ← SCAN(nearB, farB)
15:         if synfarA ∨ synfarB then              ▷ Missed alignment
                if synfarA is further into A than synfarB is into B then
                    farA ← synfarA
                else
                    farB ← synfarB
20:         else if synfarA then farA ← synfarA
            else if synfarB then farB ← synfarB
            if Chars. at farA and farB same then
                ▷ Aligned; calc. nears after proc. point
                enA ← MIN(nearA, A + proc − 1)
25:             enB ← MIN(nearB, B + proc − 1)
                ▷ Unaligned lengths
                unA = positive dist. from enA to farA
                unB = positive dist. from enB to farB
                procedure ALIGN(un, far, buffer, other)
30:                 distance ← distance + un
                    proc = far's distance into buffer
                    if procstr = other then proc ← proc + 1
                    procstr ← buffer
                if unA > unB then
35:                 ALIGN(unA, farA, A, B)
                else
                    ALIGN(unB, farB, B, A)
                if farA = endA then                ▷ Ending
                    distance ← distance+ distance between farB and endB
40:             else if farA = endA then
                    distance ← distance+ distance between farA and endA
                else                               ▷ Advanced with closed window
                    nearA ← farA ← farA + 1
                    nearB ← farB ← farB + 1
45:                 proc ← proc + 1
            else                                   ▷ Not aligned; widen windows
                if farA ≠ endA then farA ← farA + 1
                if farB ≠ endB then farB ← farB + 1
        return dist
```

deletions. Pseudocode is presented as algorithm 1, which works on a pair of string buffers, and `upstr.c` in the tool source contains a C implementation. This approach is still $O(n \times m)$ worst-case, but is $O(n)$ (where n is the shorter string) for identical strings, and degrades smoothly as *contiguous* differences increase in size: instead of two minutes, the tiny test set was compared in a little over ten seconds.

Unfortunately, changes such as 'ABCF' to 'ADCDBCF' can return overestimates, as the localisation which explicitly prevents full lookback (and keeps computational cost below $O(n^2)$) causes the 'C' in 'BCF' to match with the 'C' in 'DCD': 'ADC' is considered a substitution of 'ABC' before the algorithm can realise that 'BC' is still intact in the string, and 'DCD' is merely an insertion. As a result, the later 'B' is considered an insertion, as it no longer matches anything, and the distance is overestimated by one. Synthetic tests showed this overestimation to be minor; tests against Levenshtein on a tiny subset of Wikipedia data (a node's first few hundred revisions, thus under heavy editing) show it to be larger, with errors in the tens, and a peak error of over two-hundred. The reason for such large errors is unclear, as the resynchronisation approach should also keep *error* localised, but it does not greatly affect the result for the purpose of minor/major determination: the majority of changes were correctly classified.

2.2.4 Grouping

We identified the following non-mutually-exclusive groupings to usefully categorise edits:

Revert	Edit which simply undoes a previous edit.
Content	Major (nontrivial) edit of the page content.
Minor	Minor (trivial) edit of the page content.
Category	Edit to the categories of a page.
List of	Edit to a page which is an index to other pages.
Indexing	Edit to categories or listings, possibly both.
Template	Edit to the templates used by a page.
Page link	Edit to an internal page link.
URL link	Edit to a WWW URL link; usually external.
Links	Edit to page or URL links.
Link only	As 'links', but excluding major edits.
Hyperstructure	Any hypermedia change: indexing, linking, or template.

These categorisations yielded simple information on which kinds of changes were made by each revision, and removed much of the 'bulk' of the dataset (the revision texts); as a result, simple scripts could then handle the data to aggregate it into various groupings in memory, so as to produce graph data and statistics for analysis.

We expand upon the definition and significance of these groups as needed in section 2.4.

2.3 Tools developed

To process the sizable dataset, we created a set of small, robust, stream-based tools in C. Stream-based processing was a necessity, as manipulating the entire data in memory at once was simply infeasible; instead, the tools are intended to be combined arbitrarily using UNIX pipes. We used standard compression tools to de- and re-compress the data for storage on disk, else the verbosity of the XML format caused processing to be heavily I/O-bound.[4] The open source Libxml2[5] library was used to parse and regenerate the XML via its SAX interface. A selection of the more notable tools:

dumptitles	Converts a MediaWiki XML dump (henceforth, "MWXML") into a plain, newline-separated, list of page titles. Useful for diagnostics, e.g. confirming that the random subset contains an appropriate range of pages.
discardnonart	Reads in MWXML, and outputs MWXML, sans any pages which are in a namespace; pedantically, due to the poor semantics of MWXML, those with colons in the title. This implements the "trim to articles" step of figure 1.
randomsubset	Reads and writes MWXML, preserving a random subset of the input pages. In order for this to be O(1) in memory consumption, this does not strictly provide a given proportion of the input; instead, the control is the probability of including a given page in the output. As a result, asking for 50% of the input *may* actually yield anywhere between none and all of the pages: it is just far more likely that the output will be around 50% of the input.[6]
categorise	Reads MWXML and categorises the revisions, outputting results to a simple XML format.
cataggr	A Perl script which processes the categorisation XML to produce final statistical results and graph data. By this point, the data are small enough that a SAX parser is used to build a custom in-memory document tree, such that manipulation is easier.

The tools are available under the open source MIT license, and can be retrieved from `http://users.ecs.soton.ac.uk/prb/phd/wikipedia/` to recreate the experiment.

---

[4] Specifically, GNU Zip for intermediate; bzip2, as originally used by Wikipedia, made processing heavily CPU-bound.

[5] `http://xmlsoft.org/`

[6] A better algorithm, which is O(1) with regards to total data size, but O(n) with regards to subset size, is to store a buffer of up to n pages, and probabilistically replace them with different pages as they are encountered. However, even this would be prohibitively memory intensive on statistically significant subset sizes, as each page may have thousands of revisions, each with thousands of bytes of text, all of which must be copied into the buffer.

2.4 Results

Because of the known error margin of the approximation of Levenshtein distance, we computed results from both genuine and approximated distances on the 0.01% subset, so as to discover and illustrate the effects of approximation; the computational cost difference between the algorithms was significant: two-and-a-half hours for genuine, eight minutes for approximated. Results were then generated from the more statistically significant 0.04% subset (27 hours). This set contained some pages on contentious topics, which had seen large numbers of revisions as a result.

Table 1 Proportions of edits related to index management

(a) 0.01% subset		(b) 0.04% subset	
Edit type	Proportion	Edit type	Proportion
Categories	8.71%	Categories	8.75%
Lists	1.97%	Lists	3.72%
Overhead	10.56%	Overhead	12.34%

2.4.1 Index management

Table 1 shows the proportions of edits in categories pertaining to index management. "Categories" are changes to the categories in which a page was placed. "Lists" are any change to any 'List of' page; these pages serve as manually-maintained indices to other pages. "Overhead" are changes which fall into either of these categories: because they are not mutually exclusive (lists may be categorised), it is not a sum of the other two values. Because these metrics do not consider the change in 'content' magnitude of a change, they are unaffected by the choice of distance algorithm.

The ten percent overhead shows a strong case for the need for stronger semantics and querying on Wikipedia; this is one of the key goals, and expected benefits, of the Semantic MediaWiki project. While virtually every 'list of' node could be replaced with a query on appropriate attributes, the gain in category efficiency is harder to measure. Any semantic wiki must still be provided with categorisation metadata such that the type of pages can be used to answer such queries. However, some improvement is to be expected, as there are current Wikipedia categories which could be inferred: either because they are a union of other categories (e.g. 'Free software' and 'Operating systems' cover the existing category 'Free software operating systems') or because they are implied by a more specialised category, and no longer need to be explicitly applied to a page.

The increase in list overhead seen in the larger subset is likely a result of having a more representative proportion of 'List of' pages. Otherwise, the results are largely consistent across sample sizes.

Table 2 Proportions of edits related to link management

(a) 0.01% subset, Levenshtein

Edit type	Proportion
Links	49.60%
Links only	35.53%
Hyperstructure	61.65%
Content	17.81%
Edit type	**Ratio/content**
Links	2.79
Links only	2.00
Hyperstructure	3.46

(b) 0.01% subset, Approximated

Edit type	Proportion
Links	49.60%
Links only	23.36%
Hyperstructure	61.65%
Content	35.60%
Edit type	**Ratio**
Links	1.39
Links only	0.71
Hyperstructure	1.73

(c) 0.04% subset, Approximated

Edit type	Proportion
Links	49.56%
Links only	25.24%
Hyperstructure	61.90%
Content	35.99%
Edit type	**Ratio**
Links	1.38
Links only	0.70
Hyperstructure	1.72

2.4.2 Link management

Table 2 shows categories related to the management of links. "Links" refers to edits which changed either page-to-page or page-to-URL links. "Links only" refers to such edits *excluding* those edits which also constituted a 'major' content change: they are edits concerned only with links and other structure. "Hyperstructure" is the category of edits which changed any of the navigational capabilities of the wiki: either categories, 'List of' pages, links, or templates. "Content" is simply the category of 'major' edits.

The overestimating effect of the approximate string distance algorithm can be seen as a greater proportion of edits being considered 'major', with a knock-on effect on reducing the ratios of edits over content edits. However, the results are consistent between the 0.01% subset with the approximated string distance, and the sample set four times the size. As a result, it would appear that the smaller size of the sample set has not introduced significant error in this case, and it is reasonable to assume that a Levenshtein distance comparison of the larger dataset would yield similar results to the 0.01% subset. Therefore, further discussion will focus on the 0.01% subset with Levenshtein distance results.

These figures show the significance of hyperstructure to Wikipedia, to a surprising degree. While we expected that link editing would prove a substantial proportion of edits compared to content, we did not anticipate that *twice as many edits change links alone than those that change content*. Most link changes were page links—those to other pages on the wiki, or metawiki—as opposed to URL links to arbitrary webpages (in some cases, pages on the wiki with special arguments). 36,076 edits modified the former, but only 8,525 the latter.

With such a proportion of editing effort being expended on modifying links on Wikipedia, there is a clear need to improve this process. Introducing richer hypermedia features to wikis, such as generic links, should prove one possible improvement. Generic links are links whose endpoints are defined by matching on criteria of the document content: a basic example being matching on a particular substring.

A generic link can specify that a page's title should link to that page, rather than requiring users to manually annotate it: some early wiki systems offered this capability, but only for page titles which were written in the unnatural 'CamelCase' capitalisation. Advanced examples such as local links, present in Microcosm [4, 5], can specify scope limits on the matching. This would help with ambiguous terms on Wikipedia, such as 'Interval', which should be linked to a specific meaning, such as 'Interval (music)'.

Table 3 Categorisation of edits for 0.01% subset, Levenshtein

Category	Registered	Unregistered	Total
List of	1,146	453	1,599
Revert	4,069	679	4,748
Category	6,121	954	7,075
URL link	5,548	2,977	8,525
Indexing	7,174	1,397	8,571
Template	7,992	1,330	9,322
Content	10,275	4,182	14,457
Minor	13,776	9,961	23,737
Link only	20,969	7,877	28,846
Page link	27,205	8,871	36,076
Links	29,671	10,606	40,277
Hyperstructure	38,358	11,701	50,059
Total	57,463	23,733	81,196

2.4.3 Overall editing distribution

Table 3 shows the categorisation of all edits in the 0.01% dataset, using Levenshtein for string distance, for registered and unregistered users. Note that the edit categories are not mutually exclusive, thus will not sum to the total number of edits by that class of user. "Minor" is the category of edits which did not appear to change anything substantial: either the information extracted from the markup remains the same, and the plaintext very similar; or a registered user annotated the edit as minor. Notably, over 5% of edits are reverts: edits completely rolling back the previous edit; this implies that a further 5% of edits are being reverted (presumably as they are deemed unsuitable).[7] A substantial amount of effort is being expended merely keeping Wikipedia 'stationary'.

Figure 2 demonstrates the distribution of users over the total number of edits they have made, in the vein of the Swartz study [11]. There is a sharp falloff of number of users as the number of edits increases (note the logarithmic scale on both

[7] Actual figures may vary in either direction: this does not detect rollbacks to versions earlier than the immediately preceding version, and 'edit wars' of consecutive rollbacks *will* be entirely included in the first 5%, not belonging in the latter.

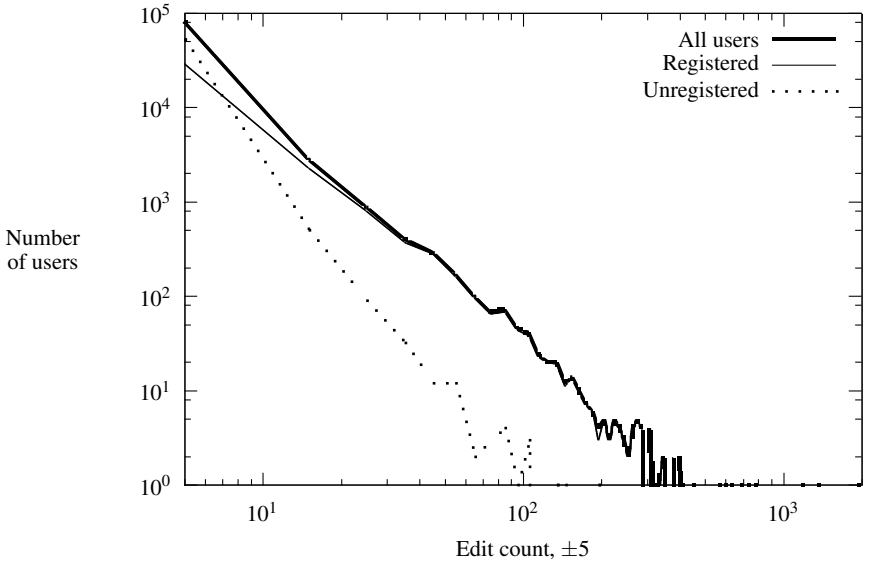

Fig. 2 User distribution over total number of edits made; 0.04% subset

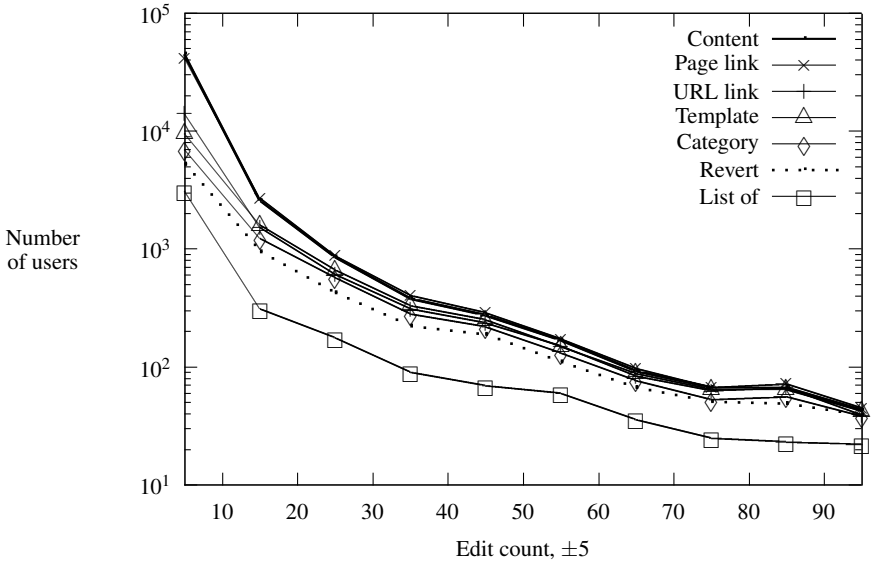

Fig. 3 User distribution over total number of edits made, by category; 0.04% subset

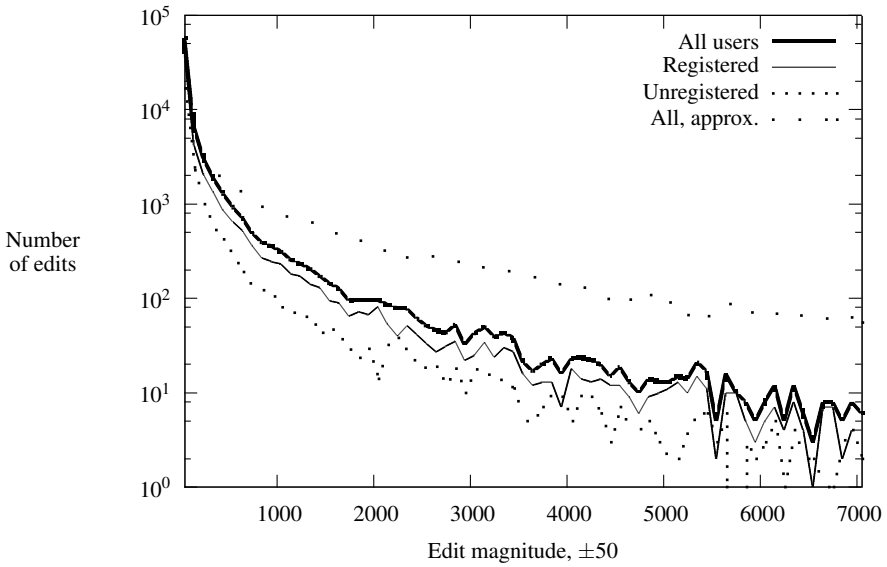

Fig. 4 Edit distribution over magnitude of edit; 0.01% subset

axes): by far, most users only ever make very few edits, whether registered or not. Unsurprisingly, registered users tend to make more edits overall, and unregistered users are dominant at the scale of fewer than ten edits.

Figure 3 breaks the low-edit end of this distribution down by basic categories. It is interesting to note that, other than being in close proximity (e.g. "content" and "page link"), the lines do not have any definitive overlaps: the breakdown of edits is consistent regardless of the number of edits the user has made. Users who have made 70 edits have made edits in the same relative proportions (i.e., more "revert" than "list of") as those who have only made five.

Figure 4 shows how the magnitude of edits breaks down by the number of edits of that magnitude, again in the vein of Swartz [11]. Because this is clearly sensitive to the string distancing algorithm, the 0.01% subset was used, with a focus on Levenshtein: the approximate distance for all users is shown as a sparsely dotted line with a consistent overestimate. These results are largely unsurprising: registered users make larger edits, and most edits are small, with the count rapidly falling off as magnitude increases.

2.4.4 Limitations of detection

There are, unfortunately, several kinds of 'overhead' costs which simply cannot be detected in a computationally feasible manner by this approach. For example, MediaWiki supports a feature called template 'substitution', which actually imports

the template, with parameter substitution performed (with some caveats), into the source text of the including node. It is important to note that the relationship between the including and included nodes is lost, and that the benefits of re-use (such as storage efficiency and later corrections) are not available. The information regarding the origin of the text is also lost without manual documentation effort, including any parameters required for the more complicated templates. Because use of this feature is not semantically recorded by MediaWiki, it is largely indistinguishable from the addition of a paragraph of wikitext. As a result, it is not then possible to evaluate the cost of maintaining or documenting these substitutions once the link to the original template has been lost.

It is also not computationally feasible to detect the pattern of a user performing the same fix on multiple pages, which would identify the cost of inadequate, or underused, transclusion. Transclusion is an inclusion-by-reference mechanism, where a selected (fragment of a) document is included 'live' into another, greatly facilitating re-use.

In Wikipedia, it is often desirable to accompany a link to a page with a short summary of that page's topic. In particular, Wikipedia has many cases where articles include a summary of another article, along with a "main article" link. The 'London' page[8], for example, has many sections which consist primarily of summaries of more detailed pages, such as 'Education in London'. However, without some form of transclusion or composition to share text, if the main article's summary changes—possibly because its subject changes—this change must be replicated manually out to any page which also summarises it. A transclusion mechanism would allow a single summary of the subject to be shared by all pages which reference it, including the main article on the subject, if desired.

For example, the 'Education in London' page may begin with a summary of its topic, highlighting the most notable institutions and successful research areas. The article on 'London' may then, within its 'Education' section, transclude this summary from the 'Education in London' page. Should the summary be updated, perhaps because a University gains significant notability in a new research area, this change would be automatically reflected in the 'London' page, as it is using the same text.

While MediaWiki's templates do function as transclusion, they are not employed for this role: common usage and development effort focus on their use as preprocessing macros.

2.5 Summary

The experiment consisted of the non-exclusive classification of edits made throughout the history of Wikipedia, a large and public wiki system. Classifications included both the areas of "text editing" (assumed to primarily be maintaining the *information*

[8] `http://en.wikipedia.org/w/index.php?title=London&oldid=155695080`

content of Wikipedia: its encyclopædic articles), and "link editing" (maintaining the *navigational structure* of the content). The hypothesis, that link editing formed a substantial proportion of total editing effort, which may potentially be automated, was supported by the results. Twice as many edits changed links alone, not affecting the article text. Edits which maintained manual indexes of pages constituted approximately a tenth of total edits.

3 Micro-scale experiment

The macro-scale experiment shows that scope exists for hypermedia-based improvements to wiki editing. To better understand the relative usefulness of these improvements, formal study must be made of current editing practices on large-scale wiki systems. This is a form of knowledge elicitation task, and thus has no particular hypothesis to test. However, the domain of possible actions, and the steps entailed in performing them, are already known as aspects of the software.

The objective of this experiment is to identify the mental processes behind wiki editing: information on the tasks editors set themselves, and how their actions are used to achieve them. This will then be used to prioritise efforts to develop hypermedia features to assist with these tasks.

3.1 Procedure

The experiment consisted of two main parts: a week of data collection while the participant used Wikipedia, or a functionally similar system, normally, followed by a meeting of less than an hour, covering a pair of protocol analysis sessions [1, 10]. A small questionnaire preceded the week of collection to record the prior experience of the subject and ensure that we covered a wide range, as well as to obtain informed consent of their willingness to participate.

The first protocol analysis was an off-line review using logged editing information from Wikipedia. Off-line study is necessary in order to work with real-world problems in a real-world environment: the reduced accuracy of recall for the reasoning behind decisions and actions is balanced against the validity of those actions. Wikipedia helps provide partial compensation here by encouraging the participant to record a short motivation for any action, which may prompt their memory.

The second protocol analysis supplemented this with an on-line self-report session in a high-fidelity simulated environment (another MediaWiki install with a tiny sample of Wikipedia's content), and a set of synthetic problems, presented in a randomised order. This then trades validity of the actions for the benefits of immediate, more accurate, feedback regarding the participant's thought processes. The investigator is also present at the time of the decision to observe any other details of the process of which the participant does not make explicit note.

Information was retrieved from the Wikipedia database about the participant's editing within the span of the study: which pages were edited, and how the source text changed. This information was publicly available as part of normal Wikipedia activity. It was not, however, directly analysed: instead, it provided material for the off-line review. The data collected were anonymised transcripts from this, and the on-line self-report, for verbal protocol analysis.

Participants were taken from geographically close wiki editors, as a practical limitation of the in-person nature of the data collection. We sought people who already had some experience with wikis, so as to capture the editing process, rather than the initial user interface experiences of a beginner. While this limited the set of potential candidates, the method of analysis is not statistical, and can work at small sample sizes [2].

The tasks for the self-report were designed around the knowledge elicitation goal, to attempt to capture the user's reasoning, and also to solicit their opinions on the perceived effort required for each task.

Edit description of same villain in two movies. Discovers how the user handles having to update a section of text which is used by two articles. The history of the page shows rough synchronisation by manual copying, but is desynchronised at the point captured for the test. Transclusion could be used to share this text.

Add fact to specific article, and to summary in general article. A similar case to the above, but this time with a level-of-detail angle. Because the latter is summary of the former article, there is less scope for text to be shared outright. The domain is the London Underground. Adaptation and transclusion could form a solution.

Refine links pointing to the disambiguation node for 'shelf'. Requires the ability to traverse (or edit, but MediaWiki does not support this) links in the reverse direction. The ability to edit links from any of their endpoints, which is facilitated by first-class links, could help with this.

Create page summarising two other articles about type of train. Requires aggregation of summaries, which are suggested to be taken from the pages' introductory overview. This again touches on the issues of synchronising shared content, and also the task of updating the list as new train articles are written. Transclusion of query-endpoint fat links could achieve this.

Add links where appropriate to plain text of 'cake' article. This task attempts to capture the reasoning behind which words in an article are hyperlinked, which then informs us where use of generic links may, or may not, be appropriate.

Add fact to infobox of 'Belgium' article. Tests resource property editing, which is hidden within template code. Provides information on how users approach what is abstractly a very simple operation. Richer support for semantics, such as forms to edit known class properties, could improve this.

All of the tasks were created from content taken from Wikipedia, for authenticity of the simulated environment. Minimal errors and omissions were introduced where necessary to set up the required problem.

3.2 Results

Six subjects participated, with a range of experience levels from casual editors with only passing knowledge of the wiki system's complicated markup, to experience with administrative tasks. All participants considered themselves to make fewer than ten major edits a week, and only participants five and six considered themselves in the 10–30 band of overall edits per week.

3.2.1 Off-line reviews

All participants had made general text edits to a range of articles: correcting typographical, spelling, and grammatical errors. One even tended to use the 'random article' feature to find articles to correct, and would perform more extensive editing to ensure that articles were suitably encylocopædic in style. Half also attempted to find supporting citations for articles that were lacking them: usually via web search, although in one case the participant found suitable sources at the bottom of the article which had simply not been linked in the appropriate places.

Two participants looked at other articles as examples when fixing markup during general editing; another two had learnt from the documentation. The preview feature was useful to refine from-example markup through trial-and-error. One participant showed signs of also learning Wikipedia policy through example: they improved the indexing of articles by adding redirects from acronyms which they expected people to use to the appropriate article. They also added an article to a disambigation page after being unable to find it via that page, and having to resort to a search; added an article to a category it had been omitted from, having seen the category used elsewhere for the same class of items; and added a "recently died" template to a person's article after hearing about the death on the news, having seen it done elsewhere. Another converted an external link to the website for a piece of software to a link to that software's Wikipedia article, as such a link should offer information which is more encyclopædic.

Some edits were removals, rather than additions. One participant reverted vandalism, using the 'undo' feature of the page history, because it was quicker than editing out the maliciously added word, and added a warning template to the page of the user responsible. They also removed a dangling link from navigational templates, reasoning that the page used to exist but has been deleted, and this template overlooked. Another participant found a section marked with a long-standing citation problem template, and removed the marker and still-unsupported text.

A participant had taken a photograph for the purpose of illustrating a notable person's Wikipedia article, which they added by using modified markup from another included image. Conversely, one of the participants decided that an article needed illustration, and sought suitable images via a web search.

One participant worked on articles for things they were particularly related to outside of the wiki: their home town and place of study. They made structural changes and, in one case, deliberately left empty subsections for some of the transport facil-

ities in their town which they intended to fill in later, or to prompt some other editor to provide more information.

Another two participants corrected errors they discovered in the process of looking to see what information Wikipedia had on a subject they were familiar with. One commented that they didn't "particularly [go] out to go and edit something; it's just that I happened to come across it". One of these edits was to remove a recurrent piece of incorrect information, so the participant left a note about it on the article's discussion page to justify its removal and try to dissuade its re-addition.

Two participants transferred information from Wikipedia variants in their native language to the English Wikipedia. Neither performed a straightforward translation: one modified it as they went, and the other largely rewrote the articles.

Large-scale edits

One of the experienced participants performed two large sets of edits: a continuation of some previously-started work on an article about a section of railway which they were preparing in a personal sandbox under their user account's namespace, and a cleanup of a series of computer games. We now cover these edits in detail.

The railway article contains a diagram of the rail network for this section of track, which is constructed out of a grid of icons using templates. With this, the participant adds some text, and references taken from web searches. The text contains a 'citation needed' claim, as the participant knew the fact but couldn't find a satisfactory citation for it, so they decided to "get the gist in" and let someone else fix it.

They found out how to construct the rail network diagrams by spotting other diagrams in rail articles and looking at how they were constructed. They then searched for the templates used to find the documentation, which includes a catalogue of icons. The participant constructs the diagrams within a sandbox because they do not want to leave 'public' nodes in a half-finished or broken state. The preview feature is unsuitable because the diagram templates are very complex and time-consuming to build up; if they make a mistake, they want to have an edit history available to revert to, else they have effectively lost their work as an incomprehensible mess.

Once it is in a "reasonable" state, the participant moves the sandbox article to its intended page about the railway station. First they prepare the links to the target page. They find articles which should be linking to it by physical properties—for a rail network, this is station adjacency—and make all the link names consistent, as dangling links can often suffer from co-reference problems.They then view the backlinks of the still-non-existent target page to ensure that all pages they expect to be linking inwards are now doing so. Finally, they copy-and-paste from the sandbox to the target, and set the sandbox pages to redirect to the new, 'public' page.

For another railway section, the participant had some information left over in their prepared notes while writing about the station which they could not work into the text of the article. Rather than leave these notes unused as a local file on their computer, they put them on the discussion page for the article, with sources, so that other editors may use them if they see fit.

The participant also found some historical pictures of a railway while browsing, and wondered if Wikipedia would have any. They discovered that Wikipedia did not, so added the pictures, and a reference.

The other major set of edits, about twenty in number, affected a series of computer games. There was one article for the series overall, and one article for the second game in the series; the participant felt that the first game should also be split out, the third was not yet up to quality—it should be copy-edited before it is moved—and that the fourth game's section was too small. First, the participant added a template proposing to split the series article apart. They found the template via cheat-sheet documentation, which they access via a shortcut term in the search box. They added the reasoning for the proposed split to the talk page, and received positive feedback from a previous editor, identified from the page history.

The participant added an 'in use' template, seen in the documentation, to the series, which acts as an advisory lock to warn other editors that their changes may be lost in conflicts, and to avoid editing now. They cut down the second game's section within the series article because it already had an article. They then factored out the first game to a separate article, created by following a dangling link, although they changed plan slightly during the process: rather than avoiding an existing disambiguation page, they replaced it with this article and added the template for 'for less common meaning X, see article Y' to the top. This did lose the very short history of the new first game article up to that point, because only administrators can move pages while preserving or merging histories. They also moved out categories and external links to the game-specific articles. After a lot of adjustments, they removed the 'in use' and 'split apart' templates.

3.2.2 On-line self-reports

Edit duplicated villain

The participants had to add a fact about a villain to his section in the articles for two films he appeared in.

Two participants simply edited both appearances of the villain; another contemplated splitting the villain out into a separate article, but didn't know how, so resorted to editing both. Two would split the article, although one suggested first looking for other examples of characters with multiple appearances to see if this case has been tackled elsewhere.

The participant most experienced with Wikipedia would make the edit in both places due to the relatively small change against the complexity of refactoring the articles, but would add templates and discussion page comment to propose a split. They consider it important to seek consensus before making a large change, so as to avoid "upsetting" other editors.

Edit summary and main article

The participants had to add an important fact to the ‘London Underground’ article, then to the ‘London Underground’ section of the ‘Transport in London’ article.

All but one participant completed the task successfully. None of the participants made any consideration of sharing text. One noted that the edits were ‘major’, as they had “added a new fact, which might be under debate”.

Disambiguate Shelf links

The participants had to find links to the disambiguation page for ‘Shelf’, and correct them to point at the specific meaning of ‘shelf’ intended.

All but one participant were able to traverse links in reverse without problems, and all participants were able to disambiguate links successfully once they arrived at the place where the link was embedded.

Summarise trains

The participants had to create a page summarising several trains from different countries.

All participants did, or declared that they would, use different text for the summary article. Only two directly derived the summary text from the text for a specific country’s train, and one of these said that they would want to work on improving the quality and consistency of the per-country articles first to avoid duplicated work. The other four wanted different summary text due to the different context: one stressed the need for simpler terminology.

One participant considered sharing some text via templates, but said that such would be complex and confusing, as the text would not then actually be in either page when editing.

With regards to keeping this summary page updated as new regional variations of the train were added, most participants simply stated that the page would need another subheading or list item. One suggested categories as a possible approach, although noted the caveat that category pages on Wikipedia can only have an overall summary of the category, not a small summary for each item listed within it. Another suggested ‘see also’ links, to keep people aware of the interdependencies of these pages, or a navigation box template if there were more than four or five types of train, which would be more visible.

Link terms in Cake

The participants had to add what they felt were appropriate links to the introductory, plain text paragraph of the ‘cake’ article.

All participants focused on nouns, and limited the amount of links they created; one commented that "all the things can be really linked to", and another wanted to avoid "overloading the user". They differed in which words they selected: two selected toward simpler terms and more fundamental ingredients (e.g. "flour", "sugar"), while the others chose those they considered to be uncommon or ambiguous terms (e.g. "buttercream", "sweetening agent").

One participant ensured that link targets existed and were suitable (e.g. not disambiguation pages); another did this for most links, but deliberately left simpler targets, such as "marzipan", unchecked as they should be created if they did not yet exist. No participant linked a term more than once, and two explicitly stated that this was a deliberate effort.

Add fact to infobox

The participants had to add the date of EU membership to the infobox in the 'Belgium' article.

Two participants were unable to complete the task, with one expressing surprise that adding information to an infobox was harder than adding to a table. Two improvised a solution which did not use the specific template key for the date of EU membership, but instead a general-purpose one for chronological events. The other two found the template documentation, and added the information with the correct key, although one had problems due the number of adjacent templates in the source, the syntax of which they found "nasty".

3.3 Summary

We set out to determine the goals editors set themselves, and how they act to achieve them.

Several of the participants edited articles to correct errors they encountered while following a primary goal of looking up some information. While this is not particularly surprising for cases of simple, non-content corrections, such as markup or typographical errors, it is counter-intuitive that people who are looking up information, and thus are presumably not experts in that field, will make more significant edits, such as finding and providing references. However, some participants looked up articles on subjects about which they are knowledgeable, either as a reference, or out of curiosity as to what information Wikipedia would have.

There are three ways shown that editors will select images to add to an article. They may deliberately seek to create them with the intent to then add them to Wikipedia; they may discover them while browsing on unrelated tasks, then decide to add them to Wikipedia; or they may be editing an article, decide that it needs illustration, and search for suitable images. There is therefore a range of premeditation to major edits such as this; an extreme case for textual editing is the major

railway work, with preparation of an entire node in a semi-private area of the wiki, and a local collection of resources.

Learning by example is a common practice to all of the participants, even those who are also adept at using the documentation. Editors often tried to keep their articles, and meta-activities such as edit comments, consistent with those of other editors. They were actively aware that other editors were at work, and in cases implicitly delegated tasks to whichever editor is inclined to address any outstanding issues, by leaving incomplete sections, dangling links, or marker templates (such as 'citation needed'). Dangling links also provided a common mechanism to create new pages, as the wiki has no explicit UI feature to do this.

Even relatively advanced features which do not offer additional capabilities over simpler ones, such as the 'undo' links in the history, may be used if they save the editor time.

In the tasks where participants were asked to share text between articles, most of them decided that they would use different text on the different articles, because of the different contexts. For the specific article/general overview case, they edited the information into the existing contexts with no outward consideration of synchronising the summaries. For the trains, several explicitly stated that different text was needed. Hence, there are cases where what is abstractly the same semantic information in multiple places may still not be sharable, because of differing presentation needs. Conversely, the 'villain' task shows that sharing is suitable in some situations, where there is a larger, mostly self-contained section of content. This task also highlights the need for better knowledge modelling on Wikipedia, as the current articles do not clearly divide the concepts of actors, characters, and films.

Templates were generally troublesome, even to the more experienced editors. While they would technically permit content sharing, as one participant observed, this has the detrimental property of "hiding" away the text while editing, requiring the editor to follow a possible chain of templates to find where the text they wish to change actually exists. Infobox templates made what should be a simple task of adding a statement about a property of a resource a complicated procedure which some participants could not complete without prompting. While Wikipedia, and hence the synthetic environment, currently runs on a non-semantic wiki, we must stress the risks in not breaking away from this templating-for-properties paradigm as one moves on to systems such as Semantic MediaWiki.

The general problem is that templates on Wikipedia, due to their macro limitations, are presentational, not declarative. This is problematic with regard to their usage for straightforward semantic properties, but the rail network diagram activities highlight this as a more general problem. The complexity of these templates stems from their need to specify exact layout and rendering of arbitrarily complex graphs, here composited from tiled images, when the actual semantic content is a relatively straightforward set of connections. In this case, simple text display of the relation data is insufficient: there is a more complex transform required to generate appropriate presentation. Other example problem domains are molecular diagrams and family trees. Wikipedia currently primarily uses manually-created images for the former, and the community are investigating approaches to entering and dis-

playing the latter,[9] but all are presentational. Solving this in the general case may be impractical without providing the facility to define Turing-complete transforms, which then introduces security and performance problems.

Both in the 'cake' task, and in general editing behaviour, all participants felt that things that exist should generally be linked, but that there is a optimal link density to maintain. The threshold to which they would link terms varies significantly between participants, from most of the nouns, to just a few phrases (nouns and noun phrases being those most likely to be article titles). They also prioritised links differently: some chose the simpler terms as their few; others the more obscure terms. All participants only linked a single instance of each term, and several commented explicitly on this decision.

4 Conclusion

We now consider how these observations apply to a hypothetical, richer hypertext system, to determine the desired ordering of feature importance.

4.1 Current strengths

We should note the importance of keeping two common wiki features, despite our push towards stronger hypertext. First, the editors made use of 'broken' hyperstructure, such as empty sections and dangling links, so we should *not* attempt to prevent this, as many classic hypertext systems did. This is somewhat of a unique point of wikis, in that their mutable nature means that navigating to a non-existent target can have useful behaviour: creating the node. Second, the editors often learn by example, so must be able to view the source of pages to see how some construct is achieved, even if they are not permitted to modify the source. Some wikis entangle the source view with the editing operation such that this is not possible, which then deprives the editors of this valuable source of real-world example usage.

These incomplete states are being used as a form of passive communication between editors. The message is implicit and, interestingly, the recipient is often simply the next editor to encounter the page who has the motivation and experience to act upon it. Because these incomplete states are co-ordination between editing users, they are potentially of no interest to reading users. However, the complexity with hiding them from non-editors is that, on a normal wiki, every user is potentially an editor, even if not logged in. While users which have not created accounts are potentially less likely to undertake major editing tasks (see "Overall editing distribution" for the macro-scale experiment), this is heuristic at best, and may discourage editors from getting involved if only because they are not aware of the incomplete changes.

[9] `http://en.wikipedia.org/w/index.php?title=Wikipedia:Family_trees&oldid=212894318`

The current approach of using different styles of link—by colour, in MediaWiki—has the advantage of leaving the decision, if also workload, of ignoring dangling links to the user.

4.2 Current weaknesses

The ability to edit links from any of their anchors is a relatively simple step from first-class linking, but we did not reveal any compelling evidence that there is a pressing need for this. All the participants, once they had found the functionality in the user interface, were able to use the wiki backlinks tool to find the endpoint at which the link was embedded, and correct it there. This capability may yet prove useful as wikis transition towards semantic links, as many semantic relations are meaningful in either direction (i.e. many properties have inverses), but is not currently a priority.

Level of detail, part of adaptation, may not be as useful as one may theoretically suppose. Abstractly, it would seem sensible that a low-detail version of a page could be used as a summary about that page when linking to it from elsewhere, as with the specific/general task. However, we have found that the surrounding context affects, if not the semantics of the content, the appropriate wording, such that these summaries are not particularly re-usable.

This also affects the use of transclusion with fat and computed links for aggregation. The most obvious application of this functionality in our synthetic tests would be the types of train, aggregating the low-detail summaries of each train type into a general page on the subject. However, this is also the case where we have identified that context affects the re-usability of the content.

Transclusive re-use of content in general, however, has useful cases. Content which is sufficiently self-contained, not a summary in the context of another page, is a potential candidate for sharing.

Edit-time transclusion solves one of the problems identified by a participant: templates hide the text away. This opacity then greatly limits the usefulness of templates for re-use. As such, we consider the transparency that would be afforded by edit-time transclusion worth prototyping.

The template mechanism also greatly overcomplicates property editing. Instance property editing based on class descriptions, where an HTML form is generated based on RDFS or OWL knowledge of property ranges and domains, would provide a much cleaner interface to this. We consider this feature highly important due to the significant problems with the current approach, but note that similar, non-research implementation is already underway in the Semantic Forms extension[10].

The linking of terms, as stressed in the ‘cake’ task, is effectively a manual form of generic linking. Outside of the synthetic tasks, this was a common ‘minor edit’ behaviour, and as such there should be enough benefit from automating it that we

[10] `http://www.mediawiki.org/wiki/Extension:Semantic_Forms`

consider this a strong priority to develop. However, we must be aware that the task is not as trivial as pattern matching. Editors have varying heuristics to determine if a 'manual' generic link should be applied to a given instance of a term, and while the lack of such variation in a deterministic algorithm may improve consistency, we must ensure that the link density is kept manageable by some comparable means. At least one restriction is reasonably clear: only one instance of a term per document should be linked.

4.3 Toward solutions

We have continued this work with the development of a model for a system which overlaps the open hypermedia and semantic web areas, with focus informed by these experiments. Our long-term goal is to continue this research by means of implementation and evaluation of a prototype system, which can be used to test the hypothesis that increased hypermedia features actually result in benefits such as a decrease of editing overhead.

References

1. Bainbridge, L.: Verbal protocol analysis. In: J.R. Wilson, E.N. Corlett (eds.) Evaluation of human work, chap. 7. Taylor & Francis Ltd (1990)
2. Borenstein, N.S.: Programming as if People Mattered, chap. 19. Princeton University Press (1991)
3. Bush, V.: As We May Think. The Atlantic Monthly **176**, 101–108 (1945). URL `http://www.theatlantic.com/doc/194507/bush`
4. Davis, H., Hall, W., Heath, I., Hill, G., Wilkins, R.: Towards an integrated information environment with open hypermedia systems. In: ECHT '92: Proceedings of the ACM conference on Hypertext, pp. 181–190. ACM Press, New York, NY, USA (1992). DOI http://doi.acm.org/10.1145/168466.168522
5. Fountain, A.M., Hall, W., Heath, I., Davis, H.: MICROCOSM: An open model for hypermedia with dynamic linking. In: European Conference on Hypertext, pp. 298–311 (1990). URL `http://citeseer.ist.psu.edu/fountain90microcosm.html`
6. Halasz, F., Schwartz, M.: The Dexter hypertext reference model. Communications of the ACM **37**(2), 30–39 (1994). DOI http://doi.acm.org/10.1145/175235.175237
7. Krötzsch, M., Vrandečić, D., Völkel, M.: Wikipedia and the semantic web - the missing links. In: Proceedings of the WikiMania2005 (2005). URL `http://www.aifb.uni-karlsruhe.de/WBS/mak/pub/wikimania.pdf`
8. Levenshtein, V.: Binary Codes Capable of Correcting Deletions, Insertions and Reversals. Soviet Physics Doklady **10**, 707 (1966)
9. Nelson, T.: Literary Machines, 93.1 edn. Mindful Press, Sausalito, California (1993)
10. Shadbolt, N., Burton, M.: Knowledge elicitation. In: J.R. Wilson, E.N. Corlett (eds.) Evaluation of human work, chap. 13. Taylor & Francis Ltd (1990)
11. Swartz, A.: Who Writes Wikipedia? (2006). URL `http://www.aaronsw.com/weblog/whowriteswikipedia`. Online only.