Streamlining text messages in a disaster management system: A Text Mining Approach

Maria Regina Estuar*
Ateneo de Manila University
Loyola Heights
Quezon City, Philippines
restuar@ateneo.edu

Hadrian Ang†
Institute for Clarity in Documentation
P.O. Box 1212
Dublin, Ohio 43017-6221
webmaster@marysville-ohio.com

Miguel Palma‡
The Thørväld Group
1 Thørväld Circle
Hekla, Iceland
larst@affiliation.org

ABSTRACT

In developing countries, effectiveness of disaster management systems can be measured through adoption. To ensure that the general public embraces the technology, the design of the system should be technology inclusive. eBayanihan is a nationwide web - mobile participatory disaster management system which captures the human dimension of disaster by allowing ordinary citizens to post incidents as they experience it. This paper discusses possible solutions to problems encountered in the SMS based platform in eBayanihan. Specifically, we address the problem of correcting incorrect syntax. (we will place our solution here).

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Theory

Keywords

SMS, text mining, disaster systems

1. DISASTERS IN THE PHILIPPINES

More recently, an average of 19 typhoons enter the area of responsibility with around 6 to 9 making a landfall on Philippine soil [**?**]. In 2014, there was a total of X tropical rainstorms that hit the Philippines causing Y pesos of damage as well as Z loss of lives. In year, the Philippine government through Republic Act No. established the National Disaster Risk Reduction Management Council whose primary role is to enter role here. There are N clusters in this council ranging from enter clusters here. The cluster in Information Communications Technology (ICT) requires managing communication infrastructure as will as delivering information from top down (official news to the public) as well as bottom up (public information to the national level).

*Project Leader of eBayanihan system

†description

‡description.

2. REVIEW OF EXISTING DISASTER INFORMATION SYSTEMS

In the Philippines, there have been efforts in contributing to the development of disaster management systems and applications to collect and report disaster related information. Discuss systems here.

As of enter 2013, the Philippines received a worldwide rank of 12 in the mobile phone ownership with a ratio of enter ratios here. Since only 30 percent of the mobile users own a smart phone, there is still a need to provide an SMS based application for disaster reporting.

2.1 Types

2.2 Measuring Effectiveness

3. SMS BASED PLATFORMS

3.1 Design

The SMS based application was designed to accept reports based on the following syntax: enter syntax here.

enter technical description of algorithm for receiving, processing and plotting

enter figure of SMS feature phone and SMS app.

enter figure of SMS to eBayanihan

3.2 Problems

To send an SMS report to eBayanihan, a specific format has to be followed so that it can be parsed properly by the system. The SMS is then parsed for a keyword, urgency level, barangay, city or municipality and then the actual report. Location information is then extracted from the SMS and used to map the report.

With this in mind, there are three problems that may occur with an SMS report. First, the SMS sent may not follow the proper format (there are extra or missing commas). Place example.

Second, keywords and urgency level may be misspelled, thus confusing the system. Place example.

Third, the location extracted may not be geo-locatable, which may occur because a) the names of places are misspelled, b) the place has not yet been mapped by the services used (Google Maps, Nominatim).

4. PROPOSED SOLUTION

There are two possible approaches in solving the problem, namely by correction and approximation. I will explain futher here.

4.1 Correction

One possible approach to the problem of erroneous SMS reports is to attempt to correct them. A program first queries the database fro new SMS that have been deemed to be wrong. The format is checked and fixed if there are mistakes. Afterward, spelling correctors may be run based on a dictionary of keywords (for misspelled keywords) and a dictionary of all the barangays and cities in the Philippines (for misspelled places). After fixing these problems, a gazetteer may then be used to get latitude and longitude for the specified place.

4.1.1 Fixing the Format

4.1.2 Approximate String Matching

One approach to the misspelled words problem is to find the closest string in the given dictionary and then replacing the string with that one. Afterward, this may then be passed on to the gazetteer query stage. Numerous string distance metrics allow this kind of approximate matching. Hamming distance, Levenshtein distance, Damerau-Levenshtein distance, Longest Common Subsequence and Longest Common Substring are just some examples of these distance metrics.

Levenshtein distance, or edit distance, is a commonly used metric in approximate string matching and spelling correction. In Peter Norvig's trials, he claims that around 76% of errors were within an edit distance of one and 98.9% were within a distance of two [6]. It is defined as the minimum number of operations required to change one string into another. The operations defined are insertion (addition of an extra letter), deletion (removal of a character from the string), and substitution (replacing one character with another). Damerau-Levenshtein distance is an extension of this that keeps the three operations, but adds a fourth one, namely transposition or the flipping of adjacent characters.

Since there are only 15 keywords and 2 urgency levels, correcting errors given the dictionary is a simple task of selecting the word with the smallest edit distance when compared to a certain query. Since only a few comparisons are needed, a straightforward dynamic programming approach was taken using the given recurrence relation.

<INSERT RECURRENCE RELATION HERE>

While the same approach may be used for the names of places, the dictionary here is a lot larger. With over a thousand cities, over a thousand municipalities and thousands of barangays, a straightforward comparison approach will be too slow (given the O(mn) run time of the dynamic programming approach). To solve this problem of efficiency, the group implemented two different tree-based solutions. The first is a hybrid of a trie and the dynamic programming approach, while the second is an implementation of a BK-Tree.

4.1.3 Trie with Dynamic Programming

A trie is a tree data structure where nodes can contain different keys pertaining to parts of a word. In this algorithm's case, each node contains a character. By using a trie, redundant computations are prevented, as prefixes to the wide range of words in the dictionary will only have to be involved in the computations once. In other words, the memoization array used in the dynamic programming algorithm is recycled, and carried over to other words with the same prefix.

<ADD PICTURE OF SAMPLE TRIE HERE>

Instead of simply using Levenshtein distance, the group instead opted to use the string optimal alignment distance function, which is a slightly restricted version of the Damerau-Levenshtein distance mentioned earlier. It also counts four operations instead of the three by the Levenshtein distance function, however, a single substring cannot be edited more than once. For example, the Damerau-Levenshtein distance of "‘there"' and "‘etre"' is two, as one can flip the first and second letters or "‘etre"' and then insert an "‘h"' in between. Its optimal string alignment distance, however, is three because after the initial transposition of the first two letters, one cannot insert another letter (the substring has already been edited). While more restrictive than Damerau-Levenshtein distance, this metric is sufficient for the purposes of the algorithm (approximate matching of wrongly spelled words).

4.1.4 BK-Trees

4.2 Approximations

Text mining is proposed approximation approach in solving the problem. In this approach, the syntax is almost disregarded as input string is tokenized and compared to a lookup table for matching. The difference in this approach is that there is a validation process that happens in every correct or incorrect match so there is a percentage increase or decrease in the matching or relevance score.

We will use a machine learning classification algo here to tag correct and incorrect match.

5. EXPERIMENTATION

5.1 Problem1: Incorrect syntax

5.2 Problem2: Incorrect keyword or location

5.3 Problem3: Location not found

6. RESULTS

7. DISCUSSION

7.0.1 Inline (In-text) Equations

A formula that appears in the running text is called an inline or in-text formula. It is produced by the **math** environment, which can be invoked with the usual `\begin. . .\end` construction or with the short form `$. . .$`. You can use any of the symbols and structures, from α to ω, available in LaTeX[5]; this section will simply show a few examples of in-text equations in context. Notice how this equation: $\lim_{n\to\infty} x = 0$, set here in in-line math style, looks slightly different when set in display style. (See next section).

7.0.2 *Display Equations*

A numbered display equation – one set off by vertical space from the text and centered horizontally – is produced by the **equation** environment. An unnumbered display equation is produced by the **displaymath** environment.

Again, in either environment, you can use any of the symbols and structures available in LaTeX; this section will just give a couple of examples of display equations in context. First, consider the equation, shown as an inline equation above:

$$\lim_{n\to\infty} x = 0 \tag{1}$$

Notice how it is formatted somewhat differently in the **displaymath** environment. Now, we'll enter an unnumbered equation:

$$\sum_{i=0}^{\infty} x + 1$$

and follow it with another numbered equation:

$$\sum_{i=0}^{\infty} x_i = \int_0^{\pi+2} f \tag{2}$$

just to demonstrate LaTeX's able handling of numbering.

7.1 Citations

Citations to articles [1, 3, 2, 4], conference proceedings [3] or books [7, 5] listed in the Bibliography section of your article will occur throughout the text of your article. You should use BibTeX to automatically produce this bibliography; you simply need to insert one of several citation commands with a key of the item cited in the proper location in the `.tex` file [5]. The key is a short reference you invent to uniquely identify each work; in this sample document, the key is the first author's surname and a word from the title. This identifying key is included with each item in the `.bib` file for your article.

The details of the construction of the `.bib` file are beyond the scope of this sample document, but more information can be found in the *Author's Guide*, and exhaustive details in the *LaTeX User's Guide*[5].

This article shows only the plainest form of the citation command, using `\cite`. This is what is stipulated in the SIGS style specifications. No other citation format is endorsed.

7.2 Tables

Because tables cannot be split across pages, the best placement for them is typically the top of the page nearest their initial cite. To ensure this proper "floating" placement of tables, use the environment **table** to enclose the table's contents and the table caption. The contents of the table itself must go in the **tabular** environment, to be aligned properly in rows and columns, with the desired horizontal and vertical rules. Again, detailed instructions on **tabular** material is found in the *LaTeX User's Guide*.

Table 1: Frequency of Special Characters

Non-English or Math	Frequency	Comments
Ø	1 in 1,000	For Swedish names
π	1 in 5	Common in math
\$	4 in 5	Used in business
Ψ_1^2	1 in 40,000	Unexplained usage

Immediately following this sentence is the point at which Table 1 is included in the input file; compare the placement of the table here with the table in the printed dvi output of this document.

To set a wider table, which takes up the whole width of the page's live area, use the environment **table*** to enclose the table's contents and the table caption. As with a single-column table, this wide table will "float" to a location deemed more desirable. Immediately following this sentence is the point at which Table 2 is included in the input file; again, it is instructive to compare the placement of the table here with the table in the printed dvi output of this document.

7.3 Figures

Like tables, figures cannot be split across pages; the best placement for them is typically the top or the bottom of the page nearest their initial cite. To ensure this proper "floating" placement of figures, use the environment **figure** to enclose the figure and its caption.

This sample document contains examples of **.eps** and **.ps** files to be displayable with LaTeX. More details on each of these is found in the *Author's Guide*.

As was the case with tables, you may want a figure that spans two columns. To do this, and still to ensure proper "floating" placement of tables, use the environment **figure*** to enclose the figure and its caption.

Note that either **.ps** or **.eps** formats are used; use the `\epsfig` or `\psfig` commands as appropriate for the different file types.

7.4 Theorem-like Constructs

Other common constructs that may occur in your article are the forms for logical constructs like theorems, axioms, corollaries and proofs. There are two forms, one produced by the command `\newtheorem` and the other by the command `\newdef`; perhaps the clearest and easiest way to distinguish them is to compare the two in the output of this sample document:

This uses the **theorem** environment, created by the `\newtheorem` command:

Table 2: Some Typical Commands

Command	A Number	Comments
`\alignauthor`	100	Author alignment
`\numberofauthors`	200	Author enumeration
`\table`	300	For tables
`\table*`	400	For wider tables

THEOREM 1. *Let f be continuous on $[a, b]$. If G is an antiderivative for f on $[a, b]$, then*

$$\int_a^b f(t)dt = G(b) - G(a).$$

The other uses the **definition** environment, created by the `\newdef` command:

Definition 1. If z is irrational, then by e^z we mean the unique number which has logarithm z:

$$\log e^z = z$$

Two lists of constructs that use one of these forms is given in the *Author's Guidelines*.

and don't forget to end the environment with figure*, not figure!

There is one other similar construct environment, which is already set up for you; i.e. you must *not* use a `\newdef` command to create it: the **proof** environment. Here is a example of its use:

PROOF. Suppose on the contrary there exists a real number L such that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = L.$$

Then

$$l = \lim_{x\to c} f(x) = \lim_{x\to c} \left[gx \cdot \frac{f(x)}{g(x)}\right] = \lim_{x\to c} g(x) \cdot \lim_{x\to c} \frac{f(x)}{g(x)} = 0 \cdot L = 0,$$

which contradicts our assumption that $l \neq 0$. □

Complete rules about using these environments and using the two different creation commands are in the *Author's Guide*; please consult it for more detailed instructions. If you need to use another construct, not listed therein, which you want to have the same formatting as the Theorem or the Definition[7] shown above, use the `\newtheorem` or the `\newdef` command, respectively, to create it.

A *Caveat* for the TEX Expert

Because you have just been given permission to use the `\newdef` command to create a new form, you might think you can use TEX's `\def` to create a new command: *Please refrain from doing this!* Remember that your LATEX source code is primarily intended to create camera-ready copy, but may be converted to other forms – e.g. HTML. If you inadvertently omit some or all of the `\def`s recompilation will be, to say the least, problematic.

8. CONCLUSIONS

This paragraph will end the body of this sample document. Remember that you might still have Acknowledgments or Appendices; brief samples of these follow. There is still the Bibliography to deal with; and we will make a disclaimer about that here: with the exception of the reference to the LATEX book, the citations in this paper are to articles which have nothing to do with the present subject and are used as examples only.

9. ACKNOWLEDGMENTS

This section is optional; it is a location for you to acknowledge grants, funding, editing assistance and what have you. In the present case, for example, the authors would like to thank Gerald Murray of ACM for his help in codifying this *Author's Guide* and the **.cls** and **.tex** files that it describes.

10. ADDITIONAL AUTHORS

Additional authors: John Smith (The Thørväld Group, email: `jsmith@affiliation.org`) and Julius P. Kumquat (The Kumquat Consortium, email: `jpkumquat@consortium.net`).

11. REFERENCES

[1] M. Bowman, S. K. Debray, and L. L. Peterson. Reasoning about naming systems. *ACM Trans. Program. Lang. Syst.*, 15(5):795–825, November 1993.

[2] J. Braams. Babel, a multilingual style-option system for use with latex's standard document styles. *TUGboat*, 12(2):291–301, June 1991.

[3] M. Clark. Post congress tristesse. In *TeX90 Conference Proceedings*, pages 84–89. TeX Users Group, March 1991.

[4] M. Herlihy. A methodology for implementing highly concurrent data objects. *ACM Trans. Program. Lang. Syst.*, 15(5):745–770, November 1993.

[5] L. Lamport. *LaTeX User's Guide and Document Reference Manual*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1986.

[6] P. Norvig. How to write a spelling corrector. http://norvig.com/spell-correct.html, 2010.

[7] S. Salas and E. Hille. *Calculus: One and Several Variable*. John Wiley and Sons, New York, 1978.

APPENDIX

A. HEADINGS IN APPENDICES

The rules about hierarchical headings discussed above for the body of the article are different in the appendices. In the **appendix** environment, the command **section** is used to indicate the start of each Appendix, with alphabetic order designation (i.e. the first is A, the second B, etc.) and a title (if you include one). So, if you need hierarchical structure *within* an Appendix, start with **subsection** as the highest

Figure 1: A sample black and white graphic (.eps format) that needs to span two columns of text.

level. Here is an outline of the body of this document in Appendix-appropriate form:

A.1 Introduction

A.2 The Body of the Paper

A.2.1 Type Changes and Special Characters

A.2.2 Math Equations

Inline (In-text) Equations

Display Equations

A.2.3 Citations

A.2.4 Tables

A.2.5 Figures

A.2.6 Theorem-like Constructs

A Caveat for the TEX Expert

A.3 Conclusions

A.4 Acknowledgments

A.5 Additional Authors

This section is inserted by LaTeX; you do not insert it. You just add the names and information in the `\additionalauthors` command at the start of the document.

A.6 References

Generated by bibtex from your .bib file. Run latex, then bibtex, then latex twice (to resolve references) to create the .bbl file. Insert that .bbl file into the .tex source file and comment out the command `\thebibliography`.

B. MORE HELP FOR THE HARDY

The acm_proc_article-sp document class file itself is chockfull of succinct and helpful comments. If you consider yourself a moderately experienced to expert user of LaTeX, you may find reading it useful but please remember not to change it.