A Lucene stemmer for MXit lingo

LL Butgereit[1,2]

RA Botha[2]

[1]Meraka Institute, CSIR

[2]Nelson Mandela Metropolitan University

lbutgereit@meraka.org.za
reinhardta.botha@nmmu.ac.za

Abstract

MXit lingo is an abbreviated form of written English used by children, teenagers and young adults when communicating using MXit as a medium over cell phones. A stemmer for MXit lingo would enable a search engine such as Lucene to index stored MXit conversations for later searching. A MXit stemmer would have to cater for the new grammatical and linguistic conventions which have developed in MXit lingo. For example, a word which contains a trailing -er may have the -er changed to an -a. Thus the word “ova” can be used in place of “over” and “unda” can be used in place of “under”. This paper describes the creation of a Lucene stemmer for MXit lingo. It also itemizes the conventions which have been noted in MXit lingo.

Keywords: MXit, Lucene, stemmer, Dr Math

Introduction

MXit lingo is one of the latest linguistic developments in South Africa. It is an abbreviated,
crisp, written language used primarily by children, teenagers and young adults when conversing over the MXit chat system on cell phones (Chigona, Chigona, Ngqokelela, & Mpofu, 2009). Although at first glance, MXit lingo seems to have abandoned spelling conventions altogether, that is not true. There are a number of very specific spelling conventions which can be recognized in MXit lingo. For example, a trailing -er or -or can be changed to a trailing -a producing words such as "ova" and "numba" which mean "over" and "number". The hard "c" in the word "factor" can be changed to a "k" in MXit lingo producing "faktor" . These two conventions can be combined creating the word "fakta" in the place of “factor”.

Lucene is an open source indexing system and search engine (Hatcher & Gospodnetic, 2004). It allows users to write their own stemmers for different human languages such as German and French. A stemmer is a utility which takes a word such as "factorization" and finds the root of that word (which is "fact" in this case)

ISBN: 978-0-620-51918-2

taking into account the various suffixes which can safely be removed from the word following the spelling conventions of that language. A MXit stemmer should also be able to take a word spelled "facta" and produce the same root "fact".

The research question for this project was:

> Can a MXit stemmer be developed for Lucene to enable easy indexing and searching of MXit conversations?

Prior to answering this research question, one sub-question needed to be answered:

> What grammatical, syntactical or linguistic conventions can be created to describe the format of MXit lingo?

The research objective was to:

> Create a MXit stemmer for the Lucene open source application.

There was one sub-objective related to the one sub-question:

> Formalise the grammatical, syntactical, or linguistic conventions about MXit lingo.

Numerous grammatical, syntactical or linguistic conventions can be seen with a simple initial glance at a MXit based conversation. For that reason, an interative research methodology was required where each iteration could cater for one of the observed conventions.

A Design and Creation Research Methodology as defined by Oates (2006) was used for the design and creation of the MXit stemmer. According to Oates, the Design and Creation Research Methodology is an iterative process involving five steps:

1. Awareness – the recognition and statement of a problem
2. Suggestions – tentative ideas of how this problems could be solved
3. Development – implementation of those ideas
4. Evaluation – assessment of the developed item
5. Conclusion – consolidation of results.

In this project, steps one through four were iterated over numerous times during the development of the MXit stemmer. This was necessary for each of the various MXit conventions which were itemised. In the example mentioned in the previous section (changing a trailing -er or -a to a traililng -a), the iterations were:

1. Awareneess – recognise that words ending with a trailing -a in MXit lingo often represent the English word which ended in an -er or -or
2. Suggestions – tentatively suggest that a trailing -a (especially when preceeded by a consonant) could be removed by the stemmer
3. Development – implement this in the MXit stemmer
4. Evaluation – test this feature using many example words from various

ISBN: 978-0-620-51918-2

conversations.

MXit

MXit is a communication mechanism for primary use over cell phones or mobile phones. It blends the portability of mobile cell phone communication with the low cost of Internet based communication to create a chat mechanism which has attracted tens of millions of users.

By using Internet based communication over cell phones (instead of SMS or MMS based communication), MXit provides an extremely low cost service. At the time of writing this paper, out-of-bundle data access for the Internet via cell phones or mobile phones cost approximately R2.00 per megabyte (or approximately thirty US cents per megabyte). With SMSs being limited to 160 characters, one megabyte of internet communication via a cell phone can provide the equivalent data as 6250 SMSs – at a much lower cost.

The majority of MXit users are teenagers; however, there is a growing group of young adults who started using MXit as a teenager and have remained with the service into adulthood.

MXit conversations are typically written in MXit lingo which is similar to “SMS lingo” or “Net lingo” or “IM lingo”. Digits and symbols are often used in place of letters which are pronounced similarly. For example, the word “l8r” replaces the word “later”. Some special acronyms have been created for entire phrases. For example, the word “wud” means “what you doing” - with the English verb “are” being unnecessary. Vowels are often omitted completely. For example, the word “fn” replaces the word “fine”.

Much has been written arguing whether or not the use of MXit lingo is harming or helping the literary skills of young people (Considine, 2004; Vosloo, 2009; Wei, 2007). This paper will not enter into that argument. Rather, the authors are attempting to treat MXit lingo as a type of language on its own, to understand its spelling conventions, and to be able to automatically stem (or find the root of) words.

“Dr Math”

“Dr Math” is a service which has been hosted at Meraka Institute since January, 2007. It provides math tutoring using MXit as a mechanism for communication. Pupils who need help with their mathematics homework can use MXit on their cell phones to chat with tutors (primarily university students and industry professionals) about mathematics. The tutors use typical Internet based computer work stations with full sized screen and keyboards. The underlying communication platform, C³TO (Chatter Call Centre/Tutoring Online), links the pupils using MXit on their cell phones to the tutors using a web based application (Butgereit, 2011).

ISBN: 978-0-620-51918-2

The conversations are typically held in this MXit lingo. (In these conversations between tutor and pupil, the tutor's statements will be in ***bold italics***)

hi can i help with matht oday?
what is x if: 2x=3-(-8/2x)
ok so you nned to first get rid of that denominaotr by mulitply 2x times everyth
ing. can you do that?
not reali...
so multiply each term by 2x that means the left is 2x*2x which is 4x^2 right
oh yeah, now i see!
so tell me what you get on the right?
3-(-8)?
no 3 * 2x - (-8/2x)*2x so that is 6x +8
im so lost...
look the right is 3-(-8/2x) so mulitply each term by 2x so the 3 becomes 3*2x (t
hat's 6x). the -8/2x times 2x gives just -8 but then neg minus neg is pos
oh ja, i c!
ok
thanx 4 helpin me..
ok plsr
bye

Although all the of the "Dr Math" tutors use full sized Internet based workstations, some of the tutors spell everything properly and other tutors attempt to "blend in" and use MXit lingo. Often tutors are extremely busy chatting with up to thirty and forty pupils at a time, and their spelling mistakes are due to being rushed and not to attempting to "blend in" with the pupils.

For the basis of this research into MXit lingo, conversations with "Dr Math" from the 2010 academic year were used for development purposes and conversations from the beginning of the 2011 academic year were used for testing purposes. Prior research by the authors included sufficient statistical analysis of these conversations to create a list of "stop words" or words which can be safely removed from conversations with "Dr Math" without harming the mathematical topic of the conversation (Butgereit & Botha, 2011). In English, for example, the phrase "The boy chased the dog in the garden" could be rephrased "boy chased dog garden" and the basic meaning of the two phrases is still the same even though from a linguistic point of view, there is a very slight difference between "the boy chased the dog in the garden" and "the boy chased the dog into the garden". That fine distinction is beyond the scope of this research. The English words "the", "in" and "into" can be considered to be "stop words".

Lucene

Lucene is an open source indexing and retrieval system. It is written in Java and released under an Apache license. It provides a number of Java utilities to easily

ISBN: 978-0-620-51918-2

enable Java developers to implement indexing and searching within their own applications. It is important to note that the indexing and the searching are two completely separate processes. The indexing process takes the input data and extracts key terms and stores them in an index. The search process allows a user to create a query using typical boolean operators, then accesses the intermediate index, and returns addresses of the original input data (Hatcher & Gospodnetic, 2004).

The Lucene indexing process is a multi-step procedure. Not all of the steps will be discussed in this paper. However, one of the steps is to remove the “stop words” of the underlying human language. This step removes the words such as “the” and “and” from the input stream. Another step is to find the stem or the root of a word. Lucene is released with an optional stemmer based on the work of Porter (Porter, 2006) which classified English words into sequences of consonants and vowels. The Java source code for the PorterStemmer is released with the Lucene sources.

Stemmers

There are a number of reasons to write stemmers. There are linguistic reasons and there are information retrieval reasons. From a linguistic point of view, it would be beneficial to develop stemmers for all of South Africa's official languages. This would allow South African citizens to search government websites for documents written in their home language. The author has supervised interns in an attempt to develop a setswana stemmer (Butgereit, 2007).

The PorterStemmer, however, deals primarily with information retrieval reasons and not linguistic reasons. Porter's criterion for removing suffixes from two words W1 and W2 to produce a common stem W requires that the statement “the document is about W1” be equivalent to the statement “the document is about W2”. Thus if it is true that “the document is about running” then it should also be true that “the document is about runners”. This works in the majority of cases. One example given by Porter where this does not work is where words take on a specific meaning in a particular industry. The example that Porter gives is with the words “relatives” and “relativity”. The statement “the document is about relatives” describes the document as being about family, parents, grandparents, and siblings. The statement “the document is about relativity” probably describes a paper about physics (Porter, 2006).

The Porter algorithm describes a word in terms of consonants and vowels. It then describes the suffixes and how they must be handled. In addition, the Porter algorithm specifies minimum lengths of some stems. There was no linguistic basis for the lengths Porter specified. He determined them just by observation. Thus the suffix trailing -ate in the word “pirate” would not be removed but in the word “activate” it would be removed.

Complex suffixes were removed piece by piece in the PorterStemmer. Thus the word “generalizations” was stripped to “generalization” which was then stripped to “generaliz” (with the trailing “e” omitted) which was then stripped to “general” and

ISBN: 978-0-620-51918-2

finally to “gener”. It is important to note that the stemmer does not have to form a perfect English word. The purpose of stemming is to bring variations of a word together. It is not to find the actual root word.

Stemmers do not understand the words they are working with. A stemmer can remove properly formed suffixes to nonsense words. For example, the following words:

tchaghs
tchaghed
tchaghing
tchagher
tchaghest
tchaghly

should all become the root “tchagh” after the stemming process. The word is nonsense but the suffixes follow English rules.

MXit Spelling Conventions

At first glance, MXit lingo seems to have abandoned all spelling conventions; however, this is not true. MXit has just augmented conventional English spelling rules with additional conventions. A number of these conventions have been already mentioned in this paper but will be repeated here. MXit spelling conventions are itemised in Table 1. It is important to note that many of the conventions itemised in Table 1 do not affect stemming. Stemming is merely the removal of suffixes as the end of a word. The conventions, whilst not necessary for this research, will be useful to future researchers.

Table 1: Summary of MXit conventions

#	Description of Convention	Example of Convention	Can Affect Stemming?
1	Any vowels can be removed	“Equation” becomes “eqtn” or “equatn”	Y
2	Trailing -er or -or change to -a	“Number” becomes “numba”, “over” becomes “ova”	Y
3	Trailing -s changed to -z	“Ladies” becomes “ladiez”, “applies” becomes “appliez”, “whats” becomes “whatz”	Y
4	Double letters become single letters	“Borrow” becomes “borow” and “small” becomes “smal”	N
5	Trailing -tion, -sion interchanged or	“Addition” becomes “addision” or “addishun”	Y

ISBN: 978-0-620-51918-2

	changed to -shun		
6	Trailing -ing changed to -n, -in or -ng	“Adding” becomes “addn” or “addng”	Y
7	Trailing -ents, -ants, -ence, -ance intermixed	“experiments” becomes “experimence”	Y
8	Trailing -tive changed to -tiv, -tif, -tv, or -tf	“Negative” becomes “negatif”, “positive” becomes “positif”	Y
9	Hard “c” changed to “k”	“Factor” becomes “faktor”	N
10	Soft “c” changed to “s”	“Circle” becomes “sircle”	N
11	“Th” changed to “d” , “t, or “f”	with” become “wid”, “wif”, or “wit”	N
12	“Ph” changed to “f”	“phone” becomes “fone”	N
13	“Wh” changed to just “w”	“what” changed to “wat”	N
14	Long “a” or long “i” changed to “y”	“night” becomes “nyt”, “late” becomes “lyt”	N
15	“fr” becomes “fw”	“friend” becomes “fwend”	N
16	Letters get swapped around	“help” becomes “hlep”	N
17	Numbers and symbols interchanged for their sounds	“What” becomes “w@”, “late” becomes “l8”	N

(The last column titled “affect stemming” indicates whether or not this spelling convention will affect any stemmer software which is written).

From these conventions, a number of sub-conventions can be derived. For example, the common past tenses formed by -ed and -ied are often spelled with just the trailing -d because some of the vowels have been removed. For example, “asked” becomes “askd” and “worked” becomes “workd”. Another sub-convention is that common suffix -er can be changed to a single -r so “worker” becomes “workr”.

In addition to these MXit lingo spelling conventions, it is important to note that they can happen in combination with each other. For example “smlr” is the word “smaller” following convention 4 (double letters changed to single letters), and 1 (remove vowels).

These conventions were derived from observations in the log files of conversations. For example, the spelling convention of removing any number of vowels can be seen in just about every sentence:

ISBN: 978-0-620-51918-2

hlp wif frctns pls

In this sentence, conventions 1 and 11 are present. The words “help”, “fractions”, and “please” all have vowels removed (convention 1). The word “with” has the trailing -th changed to a trailing -f (convention 11).

Convention 2 can clearly be seen in the following examples. The word spelled “ova” appears only in sentences where the word “over” is clearly indicated:

f of x equal 1 ova 2 x squared and g of x equal 2 -1 ova x plus 1 all plus 1

sinx.cosx is ova 1?

6x ova x-3 - x-3 ova x+3 =9 ova -(x^2-9)

3plus or minus root of 225 ova 2

And the word spelled “numba” only appears where the word “number” would be appropriate:

the numba 1,618

final numba is nt 2 bt is 28

The convention of plurals or singular verbs being written with a trailing -z instead of a trailing -s was derived from sentences such as:

i need help wit my mathz

i can subract fwm 180 dgrez 4 da anonymas anglz

wtz de difarenc b2wn de perimita nd de area?

So it is possible to take a sentence such as:

k nw dats de 1 dat i dnt undrstand plz explyn it in anothr wy plz

and see multiple spelling conventions. Convention 1 (removing vowels) is clearly present in “k”, “nw”, “dnt”, “undrstand”, “plz”, “anothr”, and “wy” where vowels have been removed. Convention 3 (changing a trailing -s to a trailing -z) can be seen in “plz”. Convention 11 (changing “th” to “d”) can be seen in “dats”, “de”, and “dat”. Convention 14 (the long “a” sound being written with a “y”) can be seen in “explyn”.

ISBN: 978-0-620-51918-2

MXit Stemmer

A MXit stemmer must be able to stem normal British and American English in addition to being able to cater for MXit spelling conventions. This is especially true in the case of attempting to stem conversations with “Dr Math” where some tutors are spelling words in full and some tutors are typing MXit lingo. In addition, some tutors use British spelling and some use American spelling.

It is important to note that the stem or root of a word does not have to be a proper English word. It is only important that the stemmer used during the indexing process is identical to the stemmer used during the retrieval purpose. In other words, it is acceptable that a stemmer process the words “happy”, “happier”, and “happiest” and generate the root stem “happ”. It is not necessary that the trailing -y be appended after other suffixes have been removed. It is extremely important, however, that the same rules or conventions apply in both the indexing and retrieval process.

Inside the MXit stemmer, a separate function is written to handle a common group of suffixes. A code outline for such a function to handle plurals and singular verbs might look like:

```
public String singular(String word) {
        String stem = word;
        int length = word.length();

        if (length > 4 && word.endsWith( “ies” ) ) {
               stem = [something]
        }
        else if ( length > 4 && word.endsWIth( “iez” ) ) {
               stem = [something]
        }
        else if ( length > 3 && word.endsWith( “es” ) ) {
               stem = [something]
        }
        else if ( length > 3 && word.endsWith( “ez” ) ) {
               stem = [something]
        }
        else if ( length > 3 && word.endsWith( “s” ) ) {
               stem = [something]
        }
        else if ( length > 3 && word.endsWith( “z” ) ) {
               stem = [something]
        }
        return stem;
}
```

The stemmer also has an internal flag to keep track of whether the original word has been changed yet. If no suffixes have been removed from the word, it is considered to be “clean”. Once a suffix has been removed, the word is considered to be “dirty”. This is to cater for that fact that when suffixes are appended together, they often have slightly different spelling. For example, in the word “calculate” there is a suffix -

ISBN: 978-0-620-51918-2

ate. But when the word is combined with the suffix -or and creates “calculator” the original -ate suffix becomes merely -at and is considered “dirty”.

In the table below, all of the words stem to the root “calcul”.

calculate
calculated
calculates
calculator
calculater
calculata
calculation
calculating
calculatn
calculatng

Stemming vs Equating

It is important to note that stemming is not the same as equating two words to be the same in MXit lingo. Stemming merely removes suffixes from the end of a word. Some of the spelling conventions itemised in this paper deal with stemming and some deal with the internal spelling of the word as indicated in the Table 1. Stemming removes the unnecessary suffixes from the end of a word. In other words, the stemming process would take a word such as “problemz” and indicate that the stem root is “problem” or it would take the word “prblmz” and indicate that the stem root is “prblm”. The stemming process can not equate the word “problem” with the word “prblm”. The stemmer can not remove vowels from the root.

To support this argument, consider five very common words and/or abbreviations in mathematics: “values”, “available”, “evaluate”, “oval”, and “vol” (a common abbreviation for the word “volume”). Removing common suffixes at the end of these words produces “value”, “avail”, “evalu”, “oval”, and “vol”. If all the vowels are now removed, these five words all collapse to “vl” and all meaning has been lost.

Not all of the spelling conventions itemised in Table 1 affect stemming. For example, the fact that “fr” may be written as “fw” does not affect any common suffixes. Some of the spelling conventions affect stemming directly such as trailing -z being used in the place of a trailing -s. Some of the spelling conventions affect stemming indirectly. For example, the fact that vowels are often omitted will change the spelling of these common English suffixes -ed, -ied, -er, -ier, -est, -iest, -ate, -ite, -ise, -fully, -ly -lier, etc.

ISBN: 978-0-620-51918-2

Evaluation

In evaluating this project, it is important to note that the unnecessary words had already been removed. That first step was to remove “stop words” unrelated to mathematics (Butgereit & Botha, 2011). After the “stop words” are removed, the remaining words (which now have a high probability of being words about mathematics) are stemmed for later indexing. It is important to remember that this stemming process does not change the internal spelling of the word (such as removing vowels or changing double letters to single letters). The researchers now understand that a future third step, which was beyond the scope of the original research question, is necessary. This will be to equate roots of different words obtained by the stemming algorithm to be the same or different words. That third step will equate the words “problem”, “prblem” and “prblm”.

Conversations taken from the first two months of the 2011 academic year were used to test the stemmer which was developed. The following steps were taken:

1. Only conversations with at least five lines between tutor and pupil were considered. Shorter conversations normally did not develop into mathematical conversations.
2. The “stop words” were removed from those conversations.
3. The remaining words were indexed using the MXit stemmer integrated into the Lucene indexer and search engine.

Lucene provides a query language which includes normal boolean operators for searching these conversations and provides a score indicating how well the conversation matched the search criterion. This allows the user to search for conversations by typing in search queries such as “parabola and root” or “parabola and quadratic”.

A simple Lucene indexing application and search application were written which implemented the MXit stemmer.

Results

Stemming only involves removing suffixes from the end of a word to provide a common root. It does not involve the internal structure of a word or the prefixes at the beginning of a word. In normal English language where the internal structure of a word (or the normal spelling of a word) does not change, traditional stemming is extremely useful.

In a communication medium such as MXit where the spelling of a word is flexible, stemming is important but can not be used on its own. Some of the critical MXit spelling conventions (such as convention 1, removing vowels, and convention 4, double letters become single letters) can not be implemented in the stemmer. If that were to happen, three words such as “meter”, “matter”, and “motor” all become the same word.

ISBN: 978-0-620-51918-2

This means that the work done on implementing a MXit stemmer for Lucene is not a final solution. Additional work needs to be done to cater for MXit spelling conventions which operate in the middle of a word.

Conclusion

Because of the nature of conversations over MXit, it is not normally useful to index them for later retrieval. The use of an indexer and search engine only becomes useful in situations where a user is looking for a specific word. This could be useful if educators were interested in specific difficulties with the mathematics curriculum. For example, an educator may wish to search the conversations with “Dr Math” for a terms such as “multiply” and be given reference to conversations which contain the terms “multipliers” and “multipliez”. It is clear to the researchers that a future third step is required so that words such as “mltpli” would also be returned by that search.

In conclusion, the research has been valuable in identifying numerous MXit spelling conventions. It has also been useful in being able to stem words properly in MXit lingo. However, additional research needs to be done to be able to fully index and search conversations for a specific word taking into account all the spelling variations of that word in MXit lingo.

As more and more South Africans communicate over MXit, it will be come ever more important to be able to process MXit lingo. Just as South African telephony services are beginning to communicate with their clients in all official languages, MXit based services should be able to communicate with their clients in MXit lingo. This research is one of the important steps in providing that facility.

References

Butgereit, L. (2007). Using open source software contributions in an internship program. *Proeedings of IST-Africa 2007, May 9-11, 2007, Maputo, Mozambique*.

Butgereit, L. (2011). *C³TO: A scalable architecture for mobile chat based tutoring.* Unpublished Masters of Technology, Nelson Mandela Metropolitan University, Port Elizabeth, South Africa.

Butgereit, L., & Botha, R. A. (2011). Stop words for "Dr Math". *Proceedings of IST-Africa, 2011, May 11-13, Gabarone, Botswana.*

Chigona, W., Chigona, A., Ngqokelela, B., & Mpofu, S. (2009). MXIT: Uses, perceptions and self-justifications. *Journal of Information, Information Technology, and Organizations, 4*, 1-16.

Considine, D. M. (2004). LINKING THE LITERACIES: Teaching & learning in a media landscape. *Wisconsin State Reading Association Journal, 44*(5), 49-53.

Hatcher, E., & Gospodnetic, O. (2004). *Lucene in action* Manning Publications.

Oates, B. J. (2006). *Researching information systems and computing* Sage Publications Ltd.

Porter, M. F. (2006). An algorithm for suffix stripping. *Program: Electronic Library and Information Systems, 40*(3), 211-218.

Vosloo, S. (2009). The effects of texting on literacy: Modern scourge or opportunity? *Shuttleworth Foundation,* 2-6.

ISBN: 978-0-620-51918-2

Wei, K. C. (2007). The impact of using net lingo in computer mediated communication on off-line writing tasks.

ISBN: 978-0-620-51918-2