Contextual Bearing on Linguistic Variation in Social Media

Stephan Gouws[*], **Donald Metzler, Congxing Cai** and **Eduard Hovy**
{gouws, metzler, ccai, hovy}@isi.edu
USC Information Sciences Institute
Marina del Rey, CA
90292, USA

Abstract

Microtexts, like SMS messages, Twitter posts, and Facebook status updates, are a popular medium for real-time communication. In this paper, we investigate the writing conventions that different groups of users use to express themselves in microtexts. Our empirical study investigates properties of lexical transformations as observed within Twitter microtexts. The study reveals that different populations of users exhibit different amounts of shortened English terms and different shortening styles. The results reveal valuable insights into how human language technologies can be effectively applied to microtexts.

1 Introduction

Microtexts, like SMS messages, Twitter posts, and Facebook status updates, are becoming a popular medium for real-time communication in the modern digital age. The ubiquitous nature of mobile phones, tablets, and other Internet-enabled consumer devices provide users with the ability to express what is on their mind nearly anywhere and at just about any time. Since such texts have the potential to provide unique perspectives on human experiences, they have recently become the focus of many studies within the natural language processing and information retrieval research communities.

The informal nature of microtexts allows users to invent *ad hoc* writing conventions that suit their particular needs. These needs strongly depend on various user contexts, such as their age, geographic location, how they want to be outwardly perceived, and so on. Hence, social factors influence the way that users express themselves in microtexts and other forms of media.

In addition to social influences, there are also usability and interface issues that may affect the way a user communicates using microtexts. For example, the Twitter microblog service imposes an explicit message length limit of 140 characters. Users of such services also often send messages using mobile devices. There may be high input costs associated with using mobile phone keypads, thus directly impacting the nature of how users express themselves.

In this paper, we look specifically at understanding the writing conventions that different groups of users use to express themselves. This is accomplished by carrying out a novel empirical investigation of the lexical transformation characteristics observed within Twitter microtexts. Our empirical evaluation includes: (i) an analysis of how frequently different user populations apply lexical transformations, and (ii) a study of the types of transformations commonly employed by different populations of users. We investigate several ways of defining user populations (e.g., based on the Twitter client, time zone, etc.). Our results suggest that not all microtexts are created equal, and that certain populations of users are much more likely to use certain types of lexical transformations than others.

This paper has two primary contributions. First, we present a novel methodology for contextualized analysis of lexical transformations found within mi-

[*] This work was done while the first author was a visiting student at ISI from the MIH Media Lab at Stellenbosch University, South Africa. Correspondence may alternatively be directed to stephan@ml.sun.ac.za.

Proceedings of the Workshop on Language in Social Media (LSM 2011), pages 20–29,
Portland, Oregon, 23 June 2011. ©2011 Association for Computational Linguistics

crotexts. The methodology leverages recent advances in automated techniques for cleaning noisy text. This approach enables us to study the frequency and types of transformations that are common within different user populations and user contexts. Second, we present results from an empirical evaluation over microtexts collected from the Twitter microblog service. Our empirical analysis reveals that within Twitter microtexts, different user populations and user contexts give rise to different forms of expression, by way of different styles of lexical transformations.

The remainder of this paper is laid out as follows. Section 2 describes related work, while Section 3 motivates our investigation. Our multi-pronged methodology for analyzing lexical transformations is described in Section 4. Section 5 describes our experimental results. Finally, Section 6 concludes the paper and describes possible directions for future work.

2 Related Work

Although our work is primarily focused on analyzing the lexical variation in language found in online social media, our analysis methodology makes strong use of techniques for normalizing 'noisy text' such as SMS-messages and Twitter messages into standard English.

Normalizing text can traditionally be approached using three well-known NLP metaphors, namely that of spell-checking, machine translation (MT) and automatic speech recognition (ASR) (Kobus et al., 2008).

In the spell-checking approach, corrections from 'noisy' words to 'clean' words proceed on a word-by-word basis. Choudhury (2007) implements the noisy channel model (Shannon and Weaver, 1948) using a hidden Markov model to handle both graphemic and phonemic variations, and Cook and Stevenson (2009) improve on this model by adapting the channel noise according to several predefined word formations such as stylistic variation, word clipping, etc. However, spelling correction is traditionally conducted in media with relatively high percentages of well-formed text where one can perform word boundary detection and thus tokenization to a high degree of accuracy. The main drawback is the strong confidence this approach places on word boundaries (Beaufort et al., 2010), since detecting word boundaries in noisy text is not a trivial problem.

In the machine translation approach (Bangalore et al., 2002; Aw et al., 2006), normalizing noisy text is considered as a translation task from a source language (the noisy text) to a target language (the cleansed text). Since noisy- and clean text typically vary wildly, it satisfies the notion of translating between two languages. However, since these transformations can be highly creative, they usually need a wide context (more than one word) to be resolved adequately. Kobus (2008) also points out that despite the fairly good results achieved with this system, such a purely phrase-based translation model cannot adequately handle the wide level of lexical creativity found in these media.

Finally, the ASR approach is based on the observation that many noisy word forms in SMSes or other noisy text are based on phonetic plays of the clean word. This approach starts by converting the input message into a phone lattice, which is converted to a word lattice using a phoneme-grapheme dictionary. Finally the word lattice is decoded by applying a language model to the word lattice and using a best-path algorithm to recover the most likely original word sequence. This approach has the advantage of being able to handle badly segmented word boundaries efficiently, however it prevents the next normalization steps from knowing what graphemes were in the initial sequence (Kobus et al., 2008).

What fundamentally separates the noisy text cleansing task from the spell-checking problem is that most often lexical ill-formedness in these media is *intentional*. Han (2011) proposes that this might be in an attempt to save characters in length-constrained media (such as Twitter or SMS), for social identity (conversing in the dialect of a specific group), or due to convention of the medium. Emotional context is typically expressed with repeat characters such as 'I am sooooooo tired' or excessive punctuation. At times, however, out-of-vocabulary tokens (spelling errors) might result purely as the result of cognitive oversight.

Cook and Stevenson (2009) are one of the first to explicitly analyze the *types* of transformations found in short message domains. They identify: 1) stylistic variation (better→betta), 2) subsequence abbreviation (doing→dng), 3) clipping of the letter 'g' (talking→talkin), 4) clipping of 'h' (hello→ello), and 5) general syllable clipping (anyway→neway), to be the most frequent transformations. Cook and Stevenson then incorporate these transformations into their model. The idea is that such an unsupervised approach based on the linguistic properties of creative word forms has the potential to be adapted for normalization in other similar genres without the cost of developing a large training corpus. Most importantly, they find that many creative texting forms are the result of a *small* number of specific word formation processes.

Han (2011) performs a simple analysis on the out-of-vocabulary words found in Twitter, and find that the majority of ill-formed words in Twitter can be attributed to instances where letters are missing or where there are extraneous letters, but the lexical correspondence to the target word is trivially accessible. They find that most ill-formed words are based on morphophonemic variations.

3 Motivation

All of the previous work described in Section 2 either

i) only focus on recovering the most likely 'standard English' form of a message, disregarding the stylistic structure of the original noisy text, or

ii) considers the structure of the noisy text found in a medium as a whole, only as a first step (the means) to identify common types of noisy transformations which can subsequently be accounted for (or 'corrected') to produce normalized messages (the desired end result).

However, based on the fact that language is highly contextual, we ask the question: What influence does the *context* in which a message is produced have on the resulting observed surface structure and style of the message?

In general, since some topics are for instance more formal or informal than others, vocabulary and linguistic style often changes based on the topic that is being discussed. Moreover, in social media one can identify several other types of context. Specifically in Twitter, one might consider a user's geographical location, the client from which a user is broadcasting her message, how long she has been using the Twitter service, and so forth.

The intuition is that the unconstrained nature of these media afford users the ability to invent writing conventions to suit their needs. Since users' needs depend on their circumstances, and hence their context, we hypothesize that the observed writing systems might be influenced by some elements of their context. For instance, phonemic writing systems might be related to a user's dialect which is related to a user's geographical location. Furthermore, highly compressed writing conventions (throwing away vowels, using prefixes of words, etc.) might result from the relatively high input cost associated with using unwieldy keypads on some mobile clients, etc.

The present work is focused on looking at these stylistic elements of messages found in social media, by analyzing the types of stylistic variation at the lexical level, across these contextual dimensions.

4 Method

In the following discussion we make a distinction between *within-tweet context* and the general *message-context* in which a message is created. Within-tweet context is the linguistic context (the other terms) that envelopes a term in a Twitter message. The general context of a Twitter message is the observable elements of the environment in which it was conceived. For the current study, we record

1. the user's **location**, and
2. the **client** from which the message was sent,

We follow a two-pronged analytic approach: Firstly, we conduct a naïve, context-free analysis (at the linguistic level) of all words not commonly found in standard, everyday English. This analysis purely looks at the terminology that are found on Twitter, and does not attempt to normalize these messages in any way. Therefore, different surface forms of the same word, such as 'today', '2day', '2d4y', are all considered distinct terms. We then analyse the terminology over different contextual dimensions such as client and location.

Secondly, we perform a more in-depth and *contextual* analysis (at the word level) by first normalizing the potentially noisy message to recover the most likely surface form of the message and recording the types of changes that were made, and then analyzing these types of changes across different general contextual dimensions (client and location).

As noted in Section 2, text message normalization is not a trivial process. As shown by Han (2011), most transformations from in-vocabulary words to out-of-vocabulary words can be attributed to a single letter that is changed, removed, or added. Furthermore, they note that most ill-formed words are related to some morphophonemic variation. We therefore implemented a text cleanser based on the design of Contractor (2010) using pre-processing techniques discussed in (Kaufmann and Kalita, 2010).

It works as follows: For each input message, we replace @-usernames with “*USR*” and urls with “*URL*”. Hash tags can either be part of the sentence (‘just got a #droid today’) or be peripheral to the sentence (‘what a loooong day! #wasted’). Following Kaufmann (2010) we remove hashtags at the end of messages when they are preceded by typical end-of-sentence punctuation marks. Hash tags in the middle of messages are retained, and the hash sign removed.

Next we tokenize this preprocessed message using the NLTK tokenizer (Loper and Bird, 2002). As noted earlier, standard NLP tools do not perform well on noisy text out-of-the-box. Based on inspection of incorrectly tokenized output, we therefore include a post-tokenization phase where we split all tokens that include a punctuation symbol into the individual one or two alphanumeric tokens (on either side of the punctuation symbol), and the punctuation symbol[1]. This heuristic catches most cases of run-on sentences.

Given a set of input tokens, we process these one by one, by comparing each token to the words in the lexicon L and constructing a confusion network CN. Each in-vocabulary term, punctuation token or other valid-but-not-in-vocabulary term is added to CN with probability 1.0 as shown in Algorithm 1. valid_tok(w_i) checks for “*USR*”, “*URL*”, or any token longer than 1 character with no alphabetical characters. This heuristic retains tokens such as ‘9-11’, ‘12:44’, etc.

Character	Transliteration candidates
1	‘1’, ‘l’, ‘one’
2	‘2’, ‘to’, ‘too’, ‘two’
3	‘3’, ‘e’, ‘three’
4	‘4’, ‘a’, ‘for’, ‘four’
5	‘5’, ‘s’, ‘five’
6	‘6’, ‘b’, ‘six’
7	‘7’, ‘t’, ‘seven’
8	‘8’, ‘ate’, ‘eight’
9	‘9’, ‘g’, ‘nine’
0	‘0’, ‘o’, ‘zero’
‘@’	‘@’, ‘at’
‘&’	’&’, ‘and’

Table 1: Transliteration lookup table.

At this stage, all out-of-vocabulary (OOV) terms represent the terms that we are uncertain about, and hence candidate terms for cleansing. First, for each OOV term, we enumerate each possibly ambiguous character into all its possible interpretations with the transliteration table shown in Table 1. This expands, for example, ‘t0day’ → [‘t0day’, ‘today’], and also ‘2day’ → [‘2day’, ‘twoday’, ‘today’], etc.

Each transliterated candidate word in each confusion set produced this way is then scored with the original word and ranked using the heuristic function (`sim()`) described in (Contractor et al., 2010)[2]. We also evaluated a purely phonetic edit-distance similarity function, based on the Double Metaphone algorithm (Philips, 2000), but found the string-similarity-based function to give more reliable results.

Each confusion set produced this way (see Algorithm 2) is joined to its previous set to form a growing confusion lattice. Finally this lattice is decoded by converting it into the probabilistic finite-state grammar format, and by using the SRI-LM toolkit’s (Stolcke, 2002) `lattice-tool` command to find the best path through the lattice by

[1] This is easily accomplished using a regular expression group-substitution of the form `(\w*)([P])(\w*)`→`[\1, \2, \3]`, where `\w` represents the set of alphanumeric characters, and `P` is the set of all punctuation marks `[.,;'"...]`

[2] The longest common subsequence between the two words, normalized by the edit distances between their consonant skeletons.

Transformation Type	Rel %
single_char (“see” → “c”)	29.1%
suffix (“why” → “y”)	18.8%
drop_vowels (“be” → “b”)	16.4%
prefix (“tomorrow” → “tom”)	9.0%
you_to_u (“you” → “u”)	8.3%
drop_last_char (“running” → “runnin”)	7.0%
repeat_letter (“so” → “soooo”)	5.5%
contraction (“you will” → “you’ll”)	5.0%
th_to_d (“this” → “dis”)	1.0%

Table 2: Most frequently observed types of transformations with an example in parentheses. *Rel %* shows the relative percentage of the top-10 transformations which were identified (excluding unidentified transformations) to belong to a specific class.

making use of a language model to promote fluidity in the text, and trained as follows:

We generated a corpus containing roughly 10M tokens of clean English tweets. We used a simple heuristic for selecting clean tweets: For each tweet we computed if $\frac{\#(OOV)}{\#(IV)+1} < \rho$, where $\rho = 0.5$ was found to give good results. On this corpus we trained a trigram language model, using Good-Turing smoothing. Next, a subset of the LA Times containing 30M words was used to train a ‘general English’ language model in the same way. These two models were combined[3] in the ratio 0.7 to 0.3.

The result of the decoding process is the hypothesized clean tokens of the original sentence. Whenever the cleanser makes a substitution, it is recorded for further analysis. Upon closer inspection, it was found that most transformation types can be recognized by using a fairly simple post-processing step. Table 2 lists the most frequent types of transformations. While these transformations do not have perfect coverage, they account for over 90% of the (correct) transformations produced by the cleanser. The rules fail to cover relatively infrequent edge cases, such as “l8r → later”, “cuz → because”, “dha → the”, and “yep → yes” [4].

Original	Cleansed
Swet baby jeebus, someone PLEASE WINE ME!	sweet baby jesus , someone please wine me !
2 years with Katie today!	two years with katie today!
k,hope nobody was hurt.gud mornin jare	okay , hope nobody was hurt . good morning jamie
When u a bum but think u da best person on da court you doodooforthebooboo	when you a bum but think you the best person on the court you dorothy
NYC premiere 2morrow.	nice premiere tomorrow .

Table 3: Examples of original and automatically cleansed versions of Twitter messages.

Algorithm 1 Main cleanser algorithm pseudo code. The `decode()` command converts the confusion network (CN) into PFSG format and decodes it using the `lattice-tool` of the SRI-LM toolkit.

```
Require: Lexicon $L$, Punctuation set $P$
  function CLEANSE_MAIN($M_{\text{in}}$)
      for $w_i \in M_{\text{in}}$ do
          if $w_i \in L \cup P$ or valid_tok($w_i$) then
              Add $(1.0, w_i)$ to CN$_{\text{out}}$      ▷ Probability 1.0
          else
              Add conf_set($w_i$) to CN$_{\text{out}}$
          end if
      end for
      return decode(CN$_{\text{out}}$)
  end function
```

Table 3 illustrates some example corrections made by the cleanser. As the results show, the cleanser is able to correct many of the more common types of transformations, but can fail when it encounters infrequent or out-of-vocabulary terms.

5 Evaluation

This section describes our empirical evaluation and analysis of how users in different contexts express themselves differently using microtexts. We focus specifically on the types of lexical transformations that are commonly applied globally, within populations of users, and in a contextualized manner.

[3] Using the `-mix-lm` and `-lambda` and `-mix-lambda2` options to the SRI-LM toolkit’s `ngram` module.

[4] To our surprise these ‘typical texting forms’ disappeared into the long tail in our data set.

Algorithm 2 Algorithm pseudo code for generating confusion set CS. $L[w_i]$ is the lexicon partitioning function for word w_i.

```
Require: Lexicon $L$, confusion set $CS$, implemented as
    top-$K$ heap containing $(s_i, w_i)$, indexed on $s_i$
  function CONF_SET($w_i$)
      W ← translits($w_i$)
      for $w_j \in$ W do
          for $w_k \in L[w_j]$ do
              $s_k \leftarrow$ sim($w_j, w_k$)
              if $s_k >$ min(CS) then
                  Add $(s_k, w_k)$ to CS
              end if
          end for
      end for
      return CS
  end function
```

5.1 Out-of-Vocabulary Analysis

We begin by analyzing the types of terms that are common in microtexts but not typically used in proper, everyday English texts (such as newspapers). We refer to such terms as being *out-of-vocabulary*, since they are not part of the common written English lexicon. The goal of this analysis is to understand how different contexts affect the number of out-of-vocabulary terms found in microtexts. We hypothesize that certain contextual factors may influence a user's ability (or interest) to formulate clean microtexts that only contain common English terms.

We ran our analysis over a collection of one million Twitter messages collected using the Twitter streaming API during 2010. Tweets gathered from the Twitter API are tagged with a language identifier that indicates the language a user has chosen for his or her account. However, we found that many tweets purported to be English were in fact not. Hence, we ran all of the tweets gathered through a simple English language classifier that was trained using a small set of manually labeled tweets, uses character trigrams and average word length as features, and achieves an accuracy of around 93%. The everyday written English lexicon, which we treat as the "gold standard" lexicon, was distilled from the same collection of LA Times news articles described in Section 4. This yielded a comprehensive lexicon of approximately half a million terms.

Timezone	% In-Vocabulary
Australia	86%
UK	85%
US (Atlantic)	84%
Hong Kong	83%
US (Pacific)	81%
Hawaii	81%
Overall	81%

Table 4: Percentage of in-vocabulary found in large English lexicon for different geographic locations.

For each tweet, the tokenized terms were looked up in the LA Times lexicon to determine if the term was out-of-vocabulary or not. Not surprisingly, the most frequent out-of-vocabulary terms identified are Twitter usernames, URLs, hasthags, and RT (the terminology for a re-broadcast, or re-tweeted, message). These tokens alone account for approximately half of all out-of-vocabulary tokens. The most frequent out-of-vocabulary terms include "lol", "haha", "gonna", "lmao", "wanna", "omg", "gotta". Numerous expletives also appear amongst the most common out-of-vocabulary terms, since such terms never appear in the LA Times. Out of vocabulary terms make up 19% of all terms in our data set.

In the remainder of this section, we examine the out-of-vocabulary properties of different populations of users based on their geographic location and their client (e.g., Web-based or mobile phone-based).

5.1.1 Geographic Locations

To analyze the out-of-vocabulary properties of users in different geographic locations, we extracted the time zone information from each Tweet in our data set. Although Twitter allows users to specify their location, many users leave this field blank, use informal terminology ("lower east side"), or fabricate non-existent locations (e.g., "wherever i want to be"). Therefore, we use the user's time zone as a proxy for their actual location, in hopes that users have less incentive to provide incorrect information.

For the Twitter messages associated with a given time zone, we computed the percentage of tokens found within our LA Times-based lexicon. The results from this analysis are provided in Table 4. It is

Client	% In-Vocabulary
Facebook	88%
Twitter for iPhone	84%
Twitter for Blackberry	83%
Web	82%
UberTwitter	78%
Snaptu	73%
Overall	81%

Table 5: Percentage of in-vocabulary found in large English lexicon for different Twitter clients.

important to note that these results were computed over hundreds of thousands of tokens, and hence the variance of our estimates is very small. This means that the differences observed here are statistically meaningful, even though the absolute differences tend to be somewhat small.

These results indicate that microtexts composed by users in different geographic locations exhibit different amounts of out-of-vocabulary terms. Users in Australia, the United Kingdom, Hong Kong, and the East Coast of the United States (e.g., New York City) include fewer out-of-vocabulary terms in their Tweets than average. However, users from the West Coast of the United States (e.g., Los Angeles, CA) and Hawaii are on-par with the overall average, but include 5% more out-of-vocabulary terms than the Australian users.

As expected, the locations with fewer-than-average in-vocabulary tokens are associated with non-English speaking countries, despite the output from the classifier.

5.1.2 Twitter Clients

In a similar experiment, we also investigated the frequency of out-of-vocabulary terms conditioned on the Twitter client (or "source") used to compose the message. Example Twitter clients include the Web-based client at `www.twitter.com`, official Twitter clients for specific mobile platforms (e.g., iPhone, Android, etc.), and third-party clients. Each client has its own characteristics, target user base, and features.

In Table 5, we show the percentage of in-vocabulary terms for a sample of the most widely used Twitter clients. Unlike the geographic location-based analysis, which showed only minor differences amongst the user populations, we see much more dramatic differences here. Some clients, such as Facebook, which provides a way of cross-posting status updates between the two services, has the largest percentage of in-vocabulary terms of the major clients in our data.

One interesting, but unexpected, finding is that the mobile phone (i.e., iPhone and Blackberry) clients have *fewer* out-of-vocabulary terms, on average, than the Web-based client. This suggests that either the users of the clients are less likely to misspell words or use slang terminology or that the clients may have better or more intuitive spell checking capabilities. A more thorough analysis is necessary to better understand the root cause of this phenomenon.

At the other end of the spectrum are the UberTwitter and Snaptu clients, which exhibit a substantially larger number of out-of-vocabulary terms. These clients are also typically used on mobile devices. As with our previous analysis, it is difficult to pinpoint the exact cause of such behavior, but we hypothesize that it is a function of user demographics and difficulties associated with inputting text on mobile devices.

5.2 Contextual Analysis

In this section, we test the hypothesis that different user populations make use of different *types* of lexical transformations. To achieve this goal, we make use of our noisy text cleanser. For each Twitter message run through the cleanser, we record the original and cleaned version of each term. For all of the terms that the cleanser corrects, we automatically identify which (if any) of the transformation rules listed in Table 2 explain the transformation between the original and clean version of the term. We use this output to analyze the distribution of transformations observed across different user populations.

We begin by analyzing the types of transformations observed across Twitter clients. Figure 1 plots the (normalized) distribution of lexical transformations observed for the Web, Twitter for Blackberry, Twitter for iPhone, and UberTwitter clients, grouped by the transformations. We also group the transformations by the individual clients in Figure 2 for more direct comparison.

The results show that Web users tend to use more

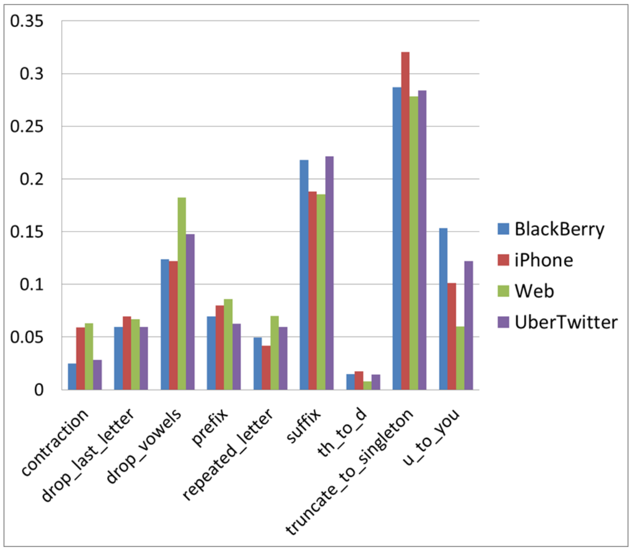

Figure 1: Proportion of transformations observed across Twitter clients, grouped by transformation type.

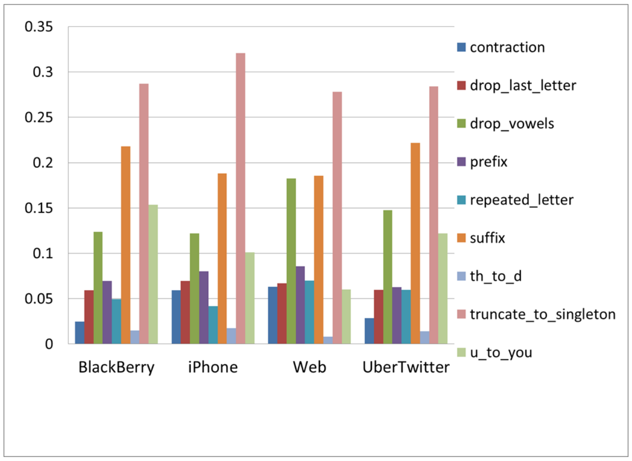

Figure 2: Proportion of transformations observed across Twitter clients, grouped by client.

contractions than Blackberry and UberTwitter users. We relate this result to the differences in typing on a virtual compared to a multi-touch keypad. It was surprising to see that iPhone users tended to use considerably more contractions than the other mobile device clients, which we relate to its word-prediction functionality. Another interesting result is the fact that Web users often drop vowels to shorten terms more than their mobile client counterparts. Instead, mobile users often use suffix-style transformations more, which is often more aggressive than the dropping vowels transformation, and possibly a result of the pervasiveness of mobile phones: Large populations of people's first interaction with technology these days are through a mobile phone, a device where strict length limits are imposed on texting, and which hence enforce habits of aggressive lexical compression, which might transfer directly to their use of PCs. Finally, we observe that mobile device users replace "you" with "u" substantially more than users of the Web client.

We also performed the same analysis across time zones/locations. The results are presented in Figure 3 by transformation-type, and again grouped by location for direct comparison in Figure 4. We observe, perhaps not surprisingly, that the East Coast US, West Coast US, and Hawaii are the most similar with respect to the types of transformations that they commonly use. However, the most interesting finding here is that British users tend to utilize a noticeably different set of transformations than American users in the Pacific time zones. For example, British users are much more likely to use contractions and suffixes, but far less likely to drop the last letter of a word, drop all of the vowels in a word, use prefix-style transformations, or to repeat a given letter multiple times. In a certain sense, this suggests that British users tend to write more proper, less informal English and make use of strikingly different styles for shortening words compared to American users. This might be related to the differences in dialects between the two regions manifesting itself during a process of phonetic transliteration when composing the messages: Inhabitants of the south-west regions in the US are known for pronouncing for instance *running* as *runnin'*, which manifests as dropping the last letter, and so forth.

Therefore, when taken with our out-of-vocabulary analysis, our experimental evaluation shows clear evidence that different populations of users express themselves differently online and use different types of lexical transformations depending on their context. It is our hope that the outcome of this study will spark further investigation into these types of issues and ultimately lead to effective contextually-aware natural language processing and information retrieval approaches that can adapt to a wide range of user contexts.

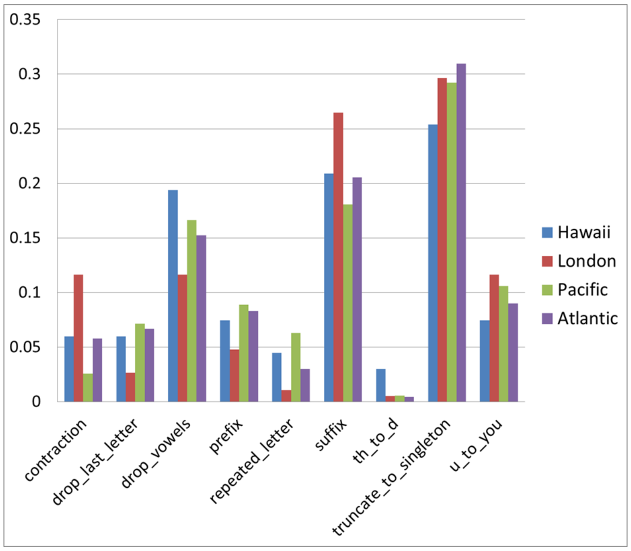

Figure 3: Proportion of transformations observed across geographic locations, grouped by transformation type.

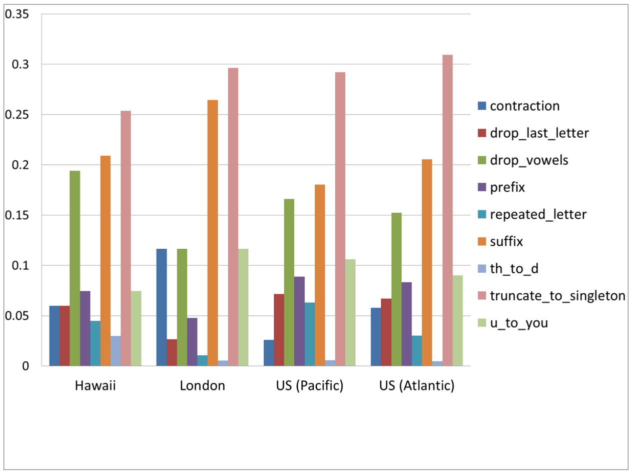

Figure 4: Proportion of transformations observed across geographic locations, grouped by location.

6 Conclusions and Future Work

This paper investigated the writing conventions that different groups of users use to express themselves in microtexts. We analyzed characteristics of terms that are commonly found in English Twitter messages but are never seen within a large collection of LA Times news articles. The results showed that a very small number of terms account for a large proportion of the out-of-vocabulary terms. The same analysis revealed that different populations of users exhibit different propensities to use out-of-vocabulary terms. For example, it was found that British users tend to use fewer out-of-vocabulary terms compared to users within the United States.

We also carried out a contextualized analysis that leveraged a state-of-the-art noisy text cleanser. By analyzing the most common types of lexical transformations, it was observed that the types of transformations used varied across Twitter clients (e.g., Web-based clients vs. mobile phone-based clients) and geographic location. This evidence supported our hypothesis that the measurable contextual indicators surrounding messages in social media play an important role in determining how messages in these media vary at the surface (lexical) level from what might be considered standard English.

The outcome of our empirical evaluation and subsequent analysis suggests that human language technologies (especially natural language processing techniques that rely on well-formed inputs) are likely to be highly susceptible to failure as the result of lexical transformations across nearly all populations and contexts. However, certain simple rules can be used to clean up a large number of out-of-vocabulary tokens. Unfortunately, such rules would not be able to properly correct the long tail of the out-of-vocabulary distribution. In such cases, more sophisticated approaches, such as the noisy text cleanser used in this work, are necessary to combat the noise. Interestingly, most of the lexical transformations observed affect non-content words, which means that most information retrieval techniques will be unaffected by such transformations.

As part of future work, we are generally interested in developing population and/or context-aware language processing and understanding techniques on top of microtexts. We are also interested in analyzing different user contexts, such as those based on age and gender and to empirically quantify the effect of noise on actual natural language processing and information retrieval tasks, such as part of speech tagging, parsing, summarization, etc.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful comments. Stephan Gouws would like to thank MIH Holdings Ltd. for financial support during the course of this work.

References

A.T. Aw, M. Zhang, J. Xiao, and J. Su. 2006. A Phrase-based Statistical Model for SMS Text Normalization. In *Proceedings of the COLING/ACL Main Conference Poster Sessions*, pages 33–40. Association for Computational Linguistics.

S. Bangalore, V. Murdock, and G. Riccardi. 2002. Bootstrapping Bilingual Data Using Consensus Translation for a Multilingual Instant Messaging System. In *Proceedings of the 19th International Conference on Computational Linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

R. Beaufort, S. Roekhaut, L.A. Cougnon, and C. Fairon. 2010. A Hybrid Rule/Model-based Finite-State Framework for Normalizing SMS Messages. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 770–779. Association for Computational Linguistics.

M. Choudhury, R. Saraf, V. Jain, A. Mukherjee, S. Sarkar, and A. Basu. 2007. Investigation and Modeling of the Structure of Texting Language. *International Journal on Document Analysis and Recognition*, 10(3):157–174.

D. Contractor, T.A. Faruquie, and L.V. Subramaniam. 2010. Unsupervised Cleansing of Noisy Text. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 189–196. Association for Computational Linguistics.

P. Cook and S. Stevenson. 2009. An Unsupervised Model for Text Message Normalization. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity*, pages 71–78. Association for Computational Linguistics.

Bo Han and Timothy Baldwin. 2011. Lexical Normalisation of Short Text Messages: Makn Sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

M. Kaufmann and J. Kalita. 2010. Syntactic Normalization of Twitter Messages.

C. Kobus, F. Yvon, and G. Damnati. 2008. Normalizing SMS: Are Two Metaphors Better Than One? In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 441–448. Association for Computational Linguistics.

E. Loper and S. Bird. 2002. NLTK: The Natural Language Toolkit. In *Proceedings of the ACL-02 Workshop on Effective tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics-Volume 1*, pages 63–70. Association for Computational Linguistics.

L. Philips. 2000. The Double Metaphone Search Algorithm. *CC Plus Plus Users Journal*, 18(6):38–43.

C.E. Shannon and W. Weaver. 1948. The Mathematical Theory of Communication. *Bell System Technical Journal*, 27:623–656.

A. Stolcke. 2002. SRILM - An Extensible Language Modeling Toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, volume 2, pages 901–904.