A Theoretical Review on SMS Normalization using Hidden Markov Models (HMMs)

Ratika Bali

Department of Computer Science and Engineering
Guru Tegh Bahadur Institute of Technology
New Delhi, India

***Abstract*— SMS language or textese is a term for the abbreviations and slang most commonly used due to the necessary brevity of mobile phone text messaging, in particular the widespread SMS (Short Message Service) communication protocol. [1] Recent times have seen a magnificent augmentation in mobile based data services that facilitate people to use SMS to access these data services. With the dynamically escalating diffusion of mobile phones, social networking and micro blogging, textese-pigeonholed by atypical acronyms, shortening and omissions, has rapidly emerged as the language of the youth. It throws up a challenge to conventional electronic processing of text and thus calls for SMS Normalization. In this research paper, the usage of Hidden Markov Models (HMMs) has been illustrated to perform SMS normalization by filtering the textese and generate noise-free conventional form of original text.**

***Keywords*— SMS, textese, noise, normalization, HMMs, training set.**

I. INTRODUCTION

. There has been a tremendous growth in the number of new mobile subscribers in the recent past. Most of these new subscribers are from developing countries where mobile is the primary information device. Even for users familiar with computers and the internet, the mobile provides unmatched portability. This has encouraged the proliferation of information services built around SMS technology. Several applications, traditionally available on Internet, are now being made available on mobile devices using SMS.

Recent analyses have revealed that short message service (SMS) and instant messaging (IM) services have emerged as the most preferred modes of communication after speech. The reason for the gargantuan popularity of IM or SMS has been the fact that this mechanism of sending and receiving messages not only saves time but costs less as well.

Millions of users all over the globe generate, transfer and store electronic content in a dialect that does not adhere to conventional grammar, syntactical punctuation and spelling standards. Words are intentionally compressed by non-standard spellings, abbreviations and phonetic transliterations.

The rest of the paper is organized as follows. Section 2 discusses the concept of SMS normalization. Section 3 and section 4 describe Markov models and Hidden Markov Models (HMMs) respectively. Section 5 focuses on the working principle of HMMs and its approach to normalize text followed by section 6 which exhibits its evaluation by throwing light on the advantages and disadvantages of the above mentioned technique. Section 7 discusses short comings of the results and section 8 concludes.

II. SMS NORMALIZATION

Noise in a text is defined as any genre of phonetic substitution and/or character deletion. The same word may be malformed by different users to different kinds of noisy variants.

SMS normalization refers to the course of converting noisy SMS text into their intended non-noisy form. Thus, SMS normalization is an indispensable prerequisite to facilitate electronic processing of such text.

Conversion of SMSes to non-noisy versions would succour enhanced speech synthesis to help visually impaired mobile phone users. Clean SMSes can be precisely translated automatically, thus enabling seamless and continuous SMS communiqué between users of different natural languages.

There are many websites [2, 3, 4] on the internet that proffer miscellaneous services to translate SMS language or textese to Standard English. But these website employ straight dictionary substitution which is not a very efficient approach as it fails to disambiguate between possible words substitutions.

This deduction led to invention of various SMS normalization techniques. A set of noisy SMSes along with their clean versions are referred to as the training set. The usage of this training set is quintessence of most of the SMS normalization techniques.

An algorithm then works on such pairs of noisy as well as clean words (derived from training set) and facilitates the translation from SMS English to English.

Type of Noise Detected in Textese	Illustration	
	Original Word/Phrase	Noisy Word/Phrase
Character elimination	Would	wud
Phonetic Replacement	Forever	4ever
Abbreviation/ Acronym	Be right back	brb
Casual/Informal Lingo	Let me	lemme

Table 1 A table illustrating the type of noise detected in textese and corresponding examples

Clean Word	Noisy version
Thanks	Thnx
Laughing Out Loud	LOL
Great	Gr8
Before	B4
Oh My God	OMG
Tomorrow	Tmrw
At	@

Table 2 A table illustrating the most common noisy versions of clean words used world-wide

III. Markov Model

A. Markov Property and Markov Random Processes

A random sequence has the Markov property if its distribution is determined solely by its current state. Any random process having this property is called a Markov random process. For observable state sequences (state is known from data), this leads to a Markov chain model. For non-observable states, this leads to a Hidden Markov Model (HMM). [5]

B. Defining Markov Model

In probability theory, a Markov model is a stochastic model that assumes the Markov property. Generally, this assumption enables reasoning and computation with the model that would otherwise be intractable. [6]

C. Markov Model Specification

1) Set of states: Defined as follows $\{s_1, s_2, \ldots, s_N\}$

2) Sequence of states: Process moves from one state to another generating a sequence of states: $s_{i1}, s_{i2}, \ldots, s_{ik}, \ldots$

3) Markov chain property: probability of each subsequent state depends only on what was the previous state:

$$P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) = P(s_{ik} \mid s_{ik-1})$$

4) *Defining Markov Model:* To define Markov model, the following probabilities have to be specified-transition probabilities $a_{ij} = P(s_i \mid s_j)$ and initial probabilities $\pi_i = P(s_i)$

5) *Calculation of Sequence Probability:* By Markov chain property, probability of state sequence can be found by the formula:

$$P(s_{i1}, s_{i2}, \ldots, s_{ik}) = P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) P(s_{i1}, s_{i2}, \ldots, s_{ik-1})$$
$$= P(s_{ik} \mid s_{ik-1}) P(s_{i1}, s_{i2}, \ldots, s_{ik-1}) = \ldots$$
$$= P(s_{ik} \mid s_{ik-1}) P(s_{ik-1} \mid s_{ik-2}) \ldots P(s_{i2} \mid s_{i1}) P(s_{i1})$$

6) *Output of the Process:* The output of the process is the set of states at each instant of time.

IV. Hidden Markov Model

A. HMM Overview

The Hidden Markov Model (HMM) is a powerful statistical tool. It is a machine learning method which makes use of state machines. It's based on probabilistic models and is useful in problems having sequential steps. It can only observe output from states, not the states themselves. For example: speech recognition. The observable states are the acoustic signals and hidden states are the phonemes (distinctive sounds of a language). [7]

B. History of HMM Theory

Basic HMM theory was developed and published in 1960s and 70s. But no widespread understanding and application cropped up until late 80s. Because the theory was published in mathematic journals which were not widely read by practicing engineers and partially due to the insufficient tutorial material facilitated for readers to understand and apply concepts.

C. HMM Elements

HMM elements can be described as follows:

HMM Element	Representation
Set of states	$\{s_1, s_2, \ldots, s_N\}$
Set of Possible Outputs	$\{v_1, v_2, \ldots, v_N\}$
Matrix of Transition Probabilities	$A=(a_{ij})$, $a_{ij}= P(s_i \mid s_j)$
Matrix of Observation Probabilities	$B=(b_i(v_m))$, $b_i(v_m) = P(v_m \mid s_i)$
Vector of Initial Probabilities	$\pi=(\pi_i)$, $\pi_i = P(s_i)$
Model	M= (A, B, π).

Table 3 A table illustrating HMM elements with respective representations.

D. HMM Assumptions

1) Markov assumption: The state transition depends only on the origin and destination.

2) *Output-independent assumption:* All observation frames are dependent on the state that generated them, not on neighbouring observation frames.

E. HMM Core Problems

There are three basic HMM problems tabulated as follows:

Problem	Given HMM	Given Observation	Compute Probability of the Observation
Evaluation	λ	$o_1, o_2, \ldots o_T$	$p\{o_1, o_2, \ldots o_T \mid \lambda\}$
Decoding	λ	$o_1, o_2, \ldots o_T$	$s_{q1}, s_{q2}, \ldots, s_{qT}$
Learning	λ	$o_1, o_2, \ldots o_T$	$p\{o_1, o_2, \ldots o_T \mid \lambda_1\} > p\{o_1, o_2, \ldots o_T \mid \lambda\}$

Table 4 A table illustrating the three basic HMM problems

F. *Applications of HMM*

1) *Speech recognition:* Recognizing spoken words and phrases.
2) *Text processing:* Parsing raw records into structured records.
3) *Text Normalization:* Transforming noisy text to noise-less text.
4) *Bioinformatics*: Protein sequence prediction
5) *Financial*: Stock market forecasts (price pattern prediction) and comparing shopping services.

V. SMS Normalization By Application Of HMMs

The HMM hypothesis for SMS normalization builds a model. This model is fabricated for apiece word in a training set of words. It uses omissions and noisy variations overtly.

A Hidden Markov model may be considered as a set of interconnected states, each of which may radiate specific values based on their output probabilities which are then observed in the output. The noisy variant of a word is considered to be radiated from a word's HMM.

Cogitate the word *today*; the ordered set of graphemes within it is {`*t*`, `*o*`, `*d*`, `*a*`, `*y*`} whereas the corresponding set of phonemes is {'T', 'AH', 'D', 'AY'}. Fig. 1(a) represents a HMM constructed out of the graphemes (characters, in our context). This is represented as a linear sequence of hidden states, each state corresponding to a token in the grapheme set.

In a non-noisy version, each HMM state would radiate the equivalent token; thus, a left-to-right HMM would always emit the exact word. However, because noise is what is to be modelled, each state is devised to be able to radiate either the corresponding grapheme, any other token (@) or nothing at all (ε). An analogous phonemic HMM is represented in Fig. 1(b).

The transformation of a phoneme to a grapheme is itself noisy, and consequently, the radiation set only includes the graphemes that could possibly map to the phoneme associated with the state. The "to" part in "today" may be transformed to the numeral "2" due to phonemic similarity, and Fig. 1(b) shows how that is accounted for in the phonemic HMM.

The graph-emic and phonemic HMMs are cross-linked intuitively to produce a single HMM as shown in Fig. 1(c) (radiation graphemes are omitted in the figure to reduce clutter). Each clean word, along with its noisy variants, is used as a training corpus to *learn* the transition probabilities and radiation probabilities.

For example, at the end of the training, state G1 may have a radiation probability distribution ['T':0.8, ε : 0.1, @:0.1] and an onward state transition distribution as [G2: 0.6, P2: 0.4].

Such erudite HMMs are thenceforth post-processed and channelled using standard techniques to decrypt the "clean" version from a noisy word. Such word-level cleansing is amassed to reach normalization of SMS texts. [5][10]

VI. Evaluation of HMM Approach

A. *HMM Approach Advantages*

1) The Hidden Markov Model (HMM) is a powerful statistical tool. It is a machine learning method which makes use of state machines. It's based on probabilistic models and is useful in problems having sequential steps.

2) Can handle variations in record structure, optional fields and varying field ordering.

B. *HMM Approach Disadvantages*

1) *Requires training using annotated data*
2) *Not completely automatic*
3) *May require manual markup*
4) *Size of training data may be an issue*

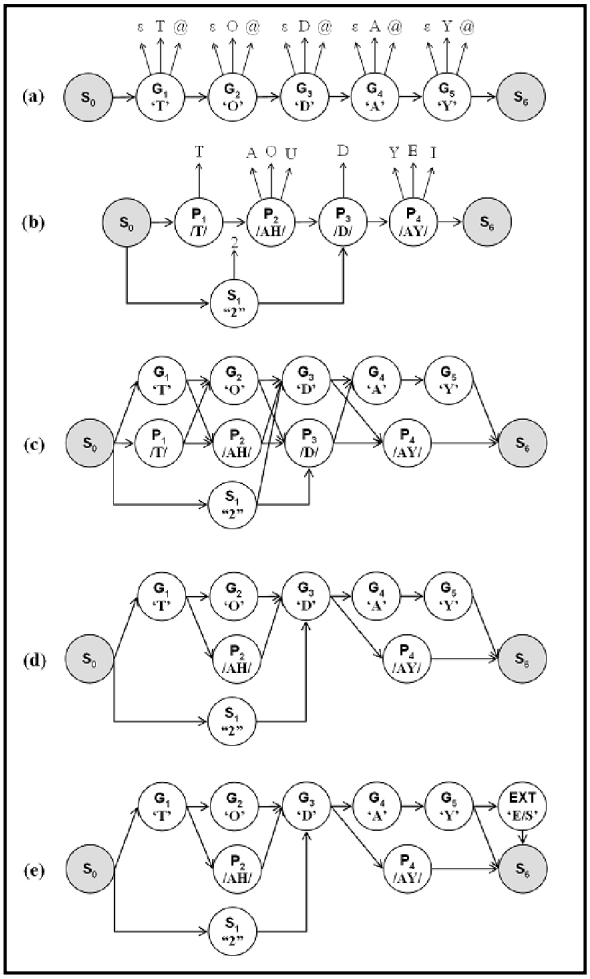

Fig 1: Construction of the word HMM illustrated for the word "today". The shaded nodes S0 and S6 represent the start and the end states respectively. (a) graph-emic path, (b) phonemic path, (c) cross linkages, (d) after state minimization, (e) after inclusion of the extended state EXT.

VII. CONCLUSIONS

With the ever escalating popularity and attractiveness of the SMS language through SMSes and micro blogging websites, cleansing SMS text has become a prerequisite for operative expansion and implementation of services such as text-to-speech and automatic translation. There has been a lot of interest in developing techniques to cleanse SMS text of late. Though, in this paper, only HMM technique of SMS normalization is elucidated..

REFERENCES

[1] http://en.wikipedia.org/wiki/SMS_language
[2] http://www.dtxtrapp.com/
[3] http://transl8it.com/
[4] http://www.lingo2word.com/translate.php
[5] http://classes.soe.ucsc.edu/cmpe264/Fall06/LecHMM.pdf
[6] http://en.wikipedia.org/wiki/Markov_model
[7] http://digital.cs.usu.edu/~cyan/CS7960/hmm-tutorial.pdf
[5] P. Deepak, V Subramaniam, et al. (2012) 'Correcting SMS Text Automatically.' CSI Communications.
[8] AiTi Aw, et al. (2006). "A Phrase-Based Statistical Model for SMS Text Normalization", *Proceedings of COLING/ ACL Conference*, Sydney, Australia.
[9] Brown, P, et al. (1993). "The mathematics of statistical machine translation: parameter estimation", *Computational Linguistics*, 19(2), 263-311.
[10] Choudhury, M, et al. (2007). "Investigation and modeling of the structure of texting language", *1st Intl. Workshop on Analytics for Noisy Unstructured Text Data*, Hyderabad, India.
[11] Contractor, D, et al. (2010). "Unsupervised cleansing of noisy text",
[12] *Proceedings of the COLING Conference*, Beijing, China.
[13] Kobus, C, et al. (2008). "Normalizing SMS: are two metaphors better than one?" *Proceedings of the COLING Conference*, Manchester.
[14] Venkata Subramaniam, L, et al. (2009). "A survey of types of text noise and techniques to handle noisy text",
[15] *Proceedings of the Third Workshop on Analytics for Noisy Unstructured Text Data*, Barcelona, Spain
[16] Venkata Subramaniam, L (2010). "Noisy Text Analytics", *Tutorial at the NAACL HLT Conference*, Los Angeles, USA. **N**