SYSTEM AND METHOD FOR OPTIMAL ENCODING OF SMS MESSAGES IN A MOBILE DEVICE

Field of the Invention

[0001] The present invention generally relates to mobile devices and Short Messaging Service (SMS) technologies, and more particularly to a system and method for optimal encoding SMS messages so as to enable the mobile devices to send longer text messages.

Background of the Invention

[0002] Mobile devices provide their users with convenience; they can be mobile phones, PDAs or other wireless devices. Currently, most, if not all, of the mobile devices have the function of Short Messaging Service (SMS). SMS is a text message service that enables short messages of generally no more than 160 characters in length to be transmitted from, and received by a mobile device. In an SMS system, messages are stored in and forwarded to SMS centers from which the SMS messages are forwarded to their designated recipients.

[0003] The limitation of characters that can be sent in one SMS message challenges mobile device users when they need to send an SMS message that is a few words more than the limitation. Common ways used to solve this problem include abbreviating commonly used words (e.g., "are" to 'r', and "the" to 'e'), and sending multiple messages to cover up the one longer than limitation. However, the abbreviation method has a number of drawbacks including: encouragement of the poor usage of proper English by the mobile device users, and limitation of the use to the mobile device users who are used to sending and receiving abbreviated messages; otherwise, the mobile device users who are not used to abbreviation would have to guess the meaning of the abbreviated letters. In addition, the drawbacks for sending multiple messages include extra cost and waste of space.

[0004] Another way to tackle the length limitation is by taking out blank spaces from the messages. For example, when the original text message of "how are you" is

compacted into "howareyou", there will be no problem for a recipient to understand the meaning of the compacted message. However, there are too many situations where recipients may have to guess the meaning of a compacted message. For instance, when an original message of "be this good" is compacted into 'bethisgood', the recipient may read it as "bet his good".

[0005] US 2003/0167306 A1 discloses a system and method for optimal Short Messaging Service (SMS) encoding in a wireless communications device having SMS capabilities. It provides a wireless communications device with three encoding formats (seven-bit ASCII, eight-bit ISO Latin 1, and two eight-bit Unicode), where the three formats have different capacities for different sets of characters. When a text message is input into the wireless device, the device will check all the characters within the text message to determine which format will accommodate all the characters; then the text message will be encoded using the format that can accommodate all the characters with minimum memory. This prior art has the intention to reduce the memory space for encoding a text message, however, it fails to do so because of its requisite of accommodating all characters within the text message. For example, when only one special character within a text message requires Unicode format to support it, the whole text message will be encoded in Unicode format, resulting in uNearest Neighborecessarily increase of the memory for the text message even if this special character bears no significance to the meaning of the text message.

Summary of the Invention

[0006] Therefore, there is an imperative need of a method and system that can encode an SMS text message in such a way that it will enable a mobile device to send and receive a longer text message simply and effectively.

[0007] In one aspect of the present invention, there is provided a method for encoding/sending an SMS text message via a mobile device for sending/receiving SMS text messages. The method comprises the steps of inputting a text message into the mobile device that provides at least two encoding formats, wherein a first of the at least two encoding formats supports less characters so that it uses less bits for encoding each character (compact format), and wherein a second of the at least two encoding formats

supports more characters so that it uses more bits for encoding each character (standard format); thereby, when the mobile device defines a limit of how many bytes of each text message can be sent/received under the standard format, the compact format can send/receive more characters within one text message for its use of less bits for encoding each character; deciding the encoding format for the input text message; wherein, when the input text message can be encoded with the standard format without exceeding the limit, it will be encoded with the standard format; wherein, when the input text message is encoded with the compact format and it still exceeds the limit, it will be encoded with the standard format; and wherein, when the input text message does not exceed the limit if encoded by the compact format but will exceed the limit if encoded by the standard format, it will be encoded with the compact format; encoding the input text message with either the compact or standard format as decided; and sending the encoded text message. In one embodiment, in step of deciding the encoding format, after the input text message is found to be suitable for the compact format but before the encoding step, the method optionally further comprising a step of checking whether the text message contains any character that is not supported the compact format, and if the text message contains any such character, it will be encoded with the standard format. In another embodiment, the mobile device is mobile phone or PDA. In yet another embodiment, the limit is defined by the SMS service system. In still another embodiment, the compact format is a 6-bit format, and the standard format an 8-bit format.

[0008] In another aspect of the present invention, there is provided a method for receiving/decoding an SMS text message via a mobile device embedded with all necessary electronic readable programs for receiving/decoding SMS text messages in at least two formats. The method comprises the steps of detecting an incoming text message; translating the incoming text message in the at least two format provided by the mobile device, wherein a first of the at least two formats supports less characters so that it uses less bits for encoding each character (compact format), and wherein a second of the at least two encoding formats supports more characters so that it uses more bits for encoding each character (standard format); thereby, when the mobile device defines a limit of how many bytes of each text message can be sent/received under the standard format, the compact format can send/receive more characters within one text message for its use of less bits for encoding each character; deciding the decoding format for the incoming text message by

manipulating the length of the translated text message within the limit; wherein, if the incoming text message is less than that of a text message with maximum characters allowed by the limit, it will be translated with the standard format; otherwise, the incoming text message will be translated with the compact and standard formats to further decide which format should be used; decoding the incoming text message with either the compact or standard format decided in the prior step; and displaying the decoded text message on the mobile device.

[0009] In one embodiment, in the deciding step, if the incoming text message is translated with the compact format and the length is more than that of a text message with maximum characters allowed by the limit that is encoded with the standard format but translated by the compact format, the decoding format is decided by the following steps of:calculating ratio patterns of the incoming text message in both formats, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers; checking the Inbox folder to see whether there are other text messages from the same sender, wherein if no, further check the Inbox folder to see if there are other previous text messages from other senders, wherein if no, then select the format by choosing the format that has the most characters; if not successful, choosing the format that has more characters than numbers; if not successful, choose the standard format; calculating the standard deviation in both the compact and standard formats of the length of the previous text messages from either the same sender or other senders; wherein, if there are previous messages which lengths are within the standard deviation range of the incoming text message length, these previous messages will be checked in the next step; and wherein, if there is no previous message within the standard deviation range, all messages will be checked in the next step; checking the ratio patterns of the selected messages from the prior step, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers; comparing the patterns of the length and two ratios in previous messages with both compact and standard formats by using CBR's NEAREST NEIGHBOR technique, then the one with the highest similarity scoring will be selected; and recommending either compact or standard format for decoding. In another embodiment, the mobile device is mobile phone or PDA. In yet another embodiment, the limit is defined by the SMS service system. In still another embodiment, the compact format is a 6-bit format, and the standard format an 8-bit format.

[0010] In yet another aspect of the present invention, there is provided a mobile device for sending/receiving an SMS text message that has more characters allowed by the limit of the mobile device under its standard encoding format. The mobile device comprises an electronic readable medium; an SMS module embedded within the electronic readable medium, wherein the SMS module comprises an SMS encoding/sending circuit for encoding an input SMS text message in at least two encoding formats depending upon the length of the text message and an SMS receiving/decoding circuit for decoding an incoming SMS text message in either of the at least two encoding format according to the length of the incoming SMS text message; and a memory embedded within the electronic readable medium for storing the SMS text messages. In one embodiment, the memory includes ROM and RAM.

[0011] In another embodiment, the at least two encoding formats of the SMS encoding/sending circuit comprise a standard encoding format that supports more characters so that its uses more bits for encoding each character (standard format), and at least one compact encoding format that supports less characters so that it uses less bits for encoding each character (compact format); thereby the mobile device can encode/send more characters in one SMS text message using the compact format; wherein the SMS encoding/sending circuit encodes/sends the input text message by a method comprising the steps of deciding the encoding format for the input text message; wherein, if the input text message can be encoded with the standard format without exceeding the limit, it will be encoded with the standard format; wherein, if the input text message is encoded with the compact format and it still exceeds the limit, it will be encoded with the standard format; and wherein, when the input text message does not exceed the limit if encoded by the compact format but will exceed the limit if encoded by the standard format, it will be encoded with the compact format; encoding the input text message with either the compact format or the standard format as decided; and sending the encoded text message. In another embodiment, in step of deciding the encoding format, after the input text message is found to be suitable for the compact format but before the encoding step, the method optionally further comprising a step of checking whether the text message contains any character that is not supported by the compact format, and if the text message contains any such character, it will be encoded with the standard format.

[0012] In yet another embodiment, the at least two encoding formats of the SMS receiving/decoding circuit comprise a standard encoding format that supports more characters so that its uses more bits for encoding each character (standard format), and at least one compact encoding format that supports less characters so that it uses less bits for encoding each character (compact format); thereby the mobile device can receive/decode more characters in one SMS text message using the compact format; wherein the SMS receiving/decoding circuit receives/sends the incoming text message by a method comprising the steps of detecting the incoming text message; translating the incoming text message in both formats; deciding the decoding format for the incoming text message by manipulating the length of the translated text message within the limit; wherein, if the incoming text message is translated with the compact format and the length is still less than that of a text message with maximum characters allowed by the limit that is encoded with the standard format but translated by the compact format, it will be translated with the standard format; otherwise, the incoming text message will be translated with both formats to further decide which format should be used for decoding; decoding the incoming text message with either the compact or standard format decided in the prior step; and displaying the decoded text message on the mobile device. In a further embodiment, in the deciding step, if the incoming text message is translated with the compact format and the length is more than that of a text message with maximum characters allowed by the limit that is encoded with the standard format but translated by the compact format, the decoding format is decided by the following steps of calculating ratio patterns of the incoming text message in both formats, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers; checking the Inbox folder to see whether there are other text messages from the same sender, wherein if no, further check the Inbox folder to see if there are other previous text messages from other senders, wherein if no, then select the format by choosing the format that has the most characters; if not successful, choosing the format that has more characters than numbers; if not successful, choose the standard format; calculating the standard deviation in both the compact and standard formats of the length of the previous text messages from either the same sender or other senders; wherein, if there are previous messages which lengths are within the standard deviation range of the incoming text message length, these previous messages will be checked in the next step; and wherein, if there is no previous message within the standard

deviation range, all messages will be checked in the next step; checking the ratio patterns of the selected messages from the prior step, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers; comparing the patterns of the length and two ratios in previous messages with both the compact format and the standard format using CBR's NEAREST NEIGHBOR technique, then the one with the highest similarity scoring will be selected; and recommending either the compact or standard format for decoding.

[0013] One advantage of the present invention is that the mobile device will be able to send more characters in one text message. For example, when the standard length for a text message is about 160 characters using an 8-bit format, a 6-bit format will enable a mobile device to send and receive a text message with a maximum of 213 characters. This results in savings in cost and transmitting time. While compaction can be done with lesser number of bits, the 6-bit compaction is preferred because it gives a good balance between optimization and commonly-used character representations.

[0014] The objectives and advantages of the invention will become apparent from the following detailed description of preferred embodiments thereof in coNearest Neighborection with the accompanying drawings.

Brief Description of the Drawings

[0015] Preferred embodiments according to the present invention will now be described with reference to the Figures, in which like reference numerals denote like elements.

[0016] FIG 1 is a block diagram of a mobile device with an SMS module for optimal encoding text messages in accordance with one embodiment of the present invention.

[0017] FIG 2 is a flowchart for sending an SMS text message via the SMS encoding/sending circuit in accordance with one embodiment of the present invention.

[0018] FIG 3 is a flowchart of the working process of the receiving/decoding circuit of the mobile device in accordance with one embodiment of the present invention.

[0019] FIG 4 is a flowchart for 6-bit and 8-bit translation examination during receiving/decoding in accordance with one embodiment of the present invention.

Detailed Description of the Invention

[0020] The present invention may be understood more readily by reference to the following detailed description of certain embodiments of the invention.

[0021] Throughout this application, where publications are referenced, the disclosures of these publications are hereby incorporated by reference, in their entireties, into this application in order to more fully describe the state of art to which this invention pertains.

[0022] In the following detailed description, specific details are set forth in order to provide a thorough understanding of the invention. However, it will be understood by those skilled in the relevant art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, and materials have not been described in detail so as not to obscure the present invention.

[0023] Referring now to FIG 1, there is provided an SMS module with an SMS encoding/sending circuit and an SMS receiving/decoding circuit embedded in a mobile device in accordance with one embodiment of the present invention. The mobile device is able to send and receive text messages with a length longer than the standard one. For example, with a current standard limit of 160 characters for one text message, the mobile device of the present invention can send a text message with a maximum of 213 characters. As shown schematically in FIG 1, the mobile device **1** comprises programs **2**, an SMS module **3**, a memory **7**, and an input/output module **6**. The programs **2** comprise all necessary functional programs for the mobile device such as camera, phone, and alarm. The SMS module **3** comprises an SMS encoding/sending circuit **4** and an SMS receiving/decoding circuit **5**, where their functions will be described in detail hereinafter. The input/output module **6** may be any available type. The memory **7** includes ROM and RAM for storing any necessary information for the functionalities of the mobile device. While the most common mobile device is the mobile phone, the present invention could be applicable to any mobile device including PDAs as long as it has the SMS capacity.

[0024] In one aspect of the present invention, an SMS text message is encoded with reduced memory using a 6-bit encoding format so that more characters can be sent in one SMS text message. Table 1 shows an example of a 6-bit encoding format.

[0025] Table 1. An example of a 6-bit encoding format

Bit	Alphabet	Bit	Alphabet
000000	A	100000	g
000001	B	100001	h
000010	C	100010	i
000011	D	100011	j
000100	E	100100	k
000101	E	100101	l
000110	G	100110	m
000111	H	100111	n
001000	I	101000	o
001001	J	101001	p
001010	K	101010	q
001011	L	101011	r
001100	M	101100	s
001101	N	101101	t
001110	O	101110	u
001111	P	101111	v
010000	Q	110000	w
010001	R	110001	x
010010	S	110010	y
010011	T	110011	z
010100	U	110100	0
010101	V	110101	1
010110	W	110110	2
010111	X	110111	3
011000	Y	111000	4
011001	Z	111001	5
011010	a	111010	6
011011	b	111011	7
011100	c	111100	8

011101	d	111101	9
011110	e	111110	<blank space>
011111	f	111111	. (full stop)

[0026] When one mobile device has the capacity of sending one standard 8-bit SMS text message with maximum of 160 characters, the 6-bit format would allow this mobile device to send an SMS with maximum of 213 characters (floor(160x8/6)). In one embodiment, the SMS encoding/sending circuit may be configured to first check whether characters in a text message can be represented by the 6-bit format. If the text message contains any characters not represented by the 6-bit format, the text message will be encoded using the normal 8-bit format. In another embodiment, a text message will be encoded in 6-bit format when the text message falls in the 6-bit format encoding range as discussed hereinafter, even if the text message contains one or more characters that are not represented by the 6-bit format. The special characters may be deleted from the encoded message or represented by any bytes that are not used in the message. It is easy to remind mobile device users to avoid any special characters when they are typing in long text messages, so that the special character problem will not be a serious issue in the employment of the 6-bit format for encoding text messages for mobile devices.

[0027] Now more details are provided for the determination of the 6-bit format encoding range. As discussed above, one objective of the present invention is to enable a mobile device to send more characters in one SMS text message by allowing the mobile device to encode the text message in a 6-bit format. It has to bear in mind that the 6-bit format should not be used if the text message has less characters than the maximum limit allowed by the standard 8-bit format. In addition, the 6-bit format has a limited compaction capacity. As mentioned above, the maximum characters in a 6-bit format allowed in a standard SMS text message is 213. Thus, a text message with 214 characters will need two SMS messages even if the 6-bit format is employed. Obviously, there is no need to use 6-bit format in this situation.

[0028] The 6-bit format encoding range can be generally represented by the following formula:

[0029]

$$\mathbf{160n < x \leq \{FLOOR(160x8/6)\}n}$$

[0030] where x = The length of the text message; and n = any integral number ($\geq$ 1). It is be appreciated that the current limitation for an SMS text message is 160 characters. In the future, when the limitation is extended, the principles embedded in this application will be still applicable when more characters are needed to be sent in a 6-bit format.

[0031] Now one example will be used to illustrate the reduction of SMS messages from two to one using 6-bit format encoding. Assume a message typed as such: "The building is standing tall at the heart of the city. You can see it even from the neighbouring islands. The height of the building is humongous, some even say that it is the best they have seen ever." The message has 202 characters in total. Referring to the formula as discussed above, this message falls into the 6-bit format range: $160<202<213$, where the n=1. More specifically, using the normal 8-bit encoding, the message will be translated to 202 bytes; the calculation is as such: 202 characters x 8 bits each = 202 bytes (2 SMS). Using the 6-bit compaction, the message will be translated to 152 bytes; the calculation is as such: 202 characters (all representable by the 6-bit compaction table) x 6 bits/8 bits = 151.5 bytes (1 SMS).

[0032] Now referring to FIG 2, there is provided a flowchart for sending an SMS text message via the SMS encoding/sending circuit **4** in accordance with one embodiment of the present invention. The sending process will start **201** when a mobile device user selects the SMS module of his mobile phone. Then in step **202**, the user keys in the texts via the input/output module **6**. As soon as the user attempts to send a text message, in step **203**, the SMS encoding/sending circuit with a built-in optimization agent will check whether the length of the text message is between the 6-bit format encoding range. If the text message does not fall within the range, it will be encoded by the 8-bit format **206** before it is sent **207** and the program ends **208**.

[0033] Still referring to FIG 2, if the text message falls within the range, then the optimization agent will examine whether the text message contains any special characters that caNearest Neighborot be represented by the 6-bit format **204**. If no special characters exist in the text message, it will be encoded in the 6-bit format **205** before it is sent **207** and the program ends **208**. If one or more special characters exist in the text message, in one embodiment, the text message will be encoded by the 8-bit format **206** before it is sent **207**

and the program ends **208**. As mentioned above, in another embodiment, the text message with special characters will still be encoded in the 6-bit format **205** before it is sent **207** and the program ends **208**, where the special characters can be either deleted from the text message or substituted with one byte that is not used in the encoding process. It is to be noted that the text message encoded in 6-bit format will be sent in an 8-bit format; thus if the total bit length of the message is not divisible by 8, padding will be performed to the message. The padding is well known to those skilled in the art.

[0034] The pseudocodes for sending process in accordance with one embodiment of the present invention is as follows:

```
check if length of message is between (160 x n) and (160 x 8/6 x n) where n is a number
larger or equal to 1
if yes then
        check if message can be represented by the 6-bit table
        if yes then
                use 6-bit compaction
        else
                use 8-bit compaction
        end if
else
        use 8-bit compaction
end if
```

[0035] An interesting observation shows that the longer the length of the message (the multiplication (n) is getting bigger) is, the more opportunity for the compaction. In fact, when $n \geq 3$, the compaction will definitely occur as the range following $n \geq 3$ will be overlapped against each other. This is shown Table 2.

[0036] Table 2. Length of text messages with numbers of SMS in different format

Length of text message	Number of SMS for 8-bit	Number of SMS for 6-bit
n = 1, range is 161-213	2	1
n = 2, range is 321-426	3	2
n = 3, range is 481-640	4	3
n = 4, range is 641-853	5	4
n = 5, range is 801-1066 ...	6 ...	5 ...

[0037] When a mobile device receives an SMS text message, its SMS receiving/decoding circuit will decode the received SMS text message according to its length. Briefly, in one embodiment, the receiving/decoding circuit will determine the number of bytes in an 8-bit format. If the message is larger than 120 bytes where there is a chance that there is an occurrence of compaction (121 bytes➔ 161 characters encoded in 6-bit or 121 characters encoded in 8-bit), the message will be translated into both 6-bit and 8-bit encoding messages; otherwise, the message will only be translated into 8-bit. If two translation occurs, the two translated messaged will then be examined on the message feasibility based on certain factors such as comparison of the length of the incoming message, its ratio of characters/numbers vs symbols and its ratio of characters vs numbers against the previous messages in the SMS inbox folder using the Case Based Reasoning's Nearest Neighbour concept calculation. The encoding type with the highest similarity will be selected.

[0038] Now referring to FIG 3, there is provided a flowchart of the working process of the receiving/decoding circuit of the mobile device in accordance with one embodiment of the present invention. The receiving/decoding process will start **301** when an SMS service center transmits a text message to a mobile phone; then the SMS receiving/decoding circuit, upon detecting a new incoming text message, will detect any incoming messages **302**, and check whether the received text message has more than 120 bytes if it is translated in an 8-bit format **303**. The number 120 is chosen because any message having more than 120 bytes may be compacted in the 6-bit format. For example, if a received text message has 121 bytes in the 8-bit format, it possibly has 161 characters encoded in the 6-bit format (121*8/6=161). If the received text message has less than 120 bytes, it will be decoded in the 8-bit format **306**, and displayed **307**, resulting in the end of the decoding program **308**.

[0039] If the received text message has more than 120 bytes, the SMS receiving/decoding circuit will examine the text message in both 6-bit and 8-bit formats **320** so as to instruct whether the received text message should be decoded in the 6-bit format or the 8-bit format. Now referring to FIG 4, a more detailed description of the examination of the received text message will be provided. The examination first calculates **321** the following patterns of the incoming text message: Ratio of characters/numbers vs symbols (8-bit only, if 6-bit predefined table does not contain any

symbol, and Ratio of characters vs numbers (6-bit and 8-bit). Then, check **322** the Inbox folder of the user mobile phone SMS module if there are other text messages from the same sender (e.g., checking by mobile phone number). If yes, the previous messages will be selected. Then, based on the previous messages selected, calculate the standard deviation of the length of the previous text messages.

[0040] Standard Deviation is calculated as follows:

$$s = \sqrt{\text{var}} = \sqrt{\frac{\Sigma(X - \bar{X})^2}{N-1}}$$

[0041] Where X = the length of one of the previous text message;

[0042] $\bar{X}$ = the average length of the previous text messages

[0043] N = the total number of previous text messages.

[0044] For example, assume 5 previous text messages with the following details:

Message No	Length
#1	200
#2	160
#3	120
#4	5
#5	400

[0045] Calculated standard deviation = 144.38.

[0046] Now referring back to step **322**, if the Inbox folder contains no previous text messages by the same sender, the SMS receiving/decoding circuit will check the Inbox folder for other previous text messages from other senders **324**. If any previous text messages are stored, the standard deviation of the length of the previous text messages will be calculated as described in step **323**. If no previous text message is stored, the process will proceed to step **329** described in detail hereinafter.

[0047] In step **325,** check if within the collection of the previous text messages if there are any messages that have lengths, which fall within the standard deviation range of the incoming text message length (both 6-bit and 8-bit). If exists, select those messages and proceed to step **327** else proceed to step **326**. For example, assume the incoming text message has the following details:

[0048]

Message No	Length
6-bit	200
8-bit	150

[0049] Using the previous calculation of standard deviation and message details in 323 example:

[0050]

Message No	Length
#1 ✓	200
#2 ✓	160
#3 ✓	120
#4 ×	5
#5 ×	400

[0051] Calculated standard deviation = 144.38.

[0052] The selected messages are #1, #2 and #3 because their lengths are within the standard deviation range of the incoming text message length (both 6-bit and 8-bit).

[0053] In step **326,** since there does not exist any previous messages' length that fall within the standard deviation range of the incoming text message length, simply select all the previous messages.

[0054] In step **327**, based on the selected messages, calculate the average patterns of the following: Ratio of characters/numbers vs symbols vs symbol, and Ratio of characters vs numbers. It is to be noted that step **321** calculates ratio for incoming messages while step **327** calculates ratio for selected messages from the Inbox folder.

[0055] In step **328**, using the *Nearest-neighbour* concept of Case Based Reasoning (CBR) computation technique, the agent compares the average pattern of the two ratios (as calculated in 327) with the two ratios of the 6-bit and 8-bit incoming text message (as calculated in 321). The *Nearest-neighbour* concept will help in deciding the weightage to calculate the similarity of previous text messages with the incoming text message. The weightages will be applied on the length of the message, the ratio of characters/numbers vs symbols and the ratio of characters vs numbers. For example, the weightages assigned are as follows: Length of the message: 20% (this means the length of the message has 20% important rate in deciding whether a previous text is similar with the incoming one); Ratio of characters/numbers vs symbols: 40%; Ratio of characters vs numbers: 40%.

[0056] Assume we have 3 previous text messages (after the filter of 325 or 326) with the following details:

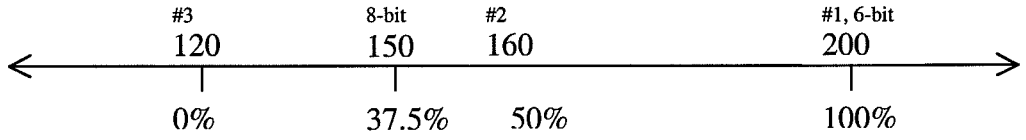
Message No	Length	Ratio of characters/numbers vs symbols	Ratio of characters vs numbers
#1	200	9:1 (0.11)	8:1 (0.125)
#2	160	11:1 (0.09)	10:1 (0.1)
#3	120	11:1 (0.09)	21:1 (0.048)

[0057] And assume our incoming text message is:

Bit Type	Length	Ratio of characters/numbers vs symbols	Ratio of characters vs numbers
6-bit	200	20:0 (0)	19:1 (0.053)
8-bit	150	14:1 (0.07)	140:0 (0)

[0058] Length:Standard deviation = 40

Mean = 160

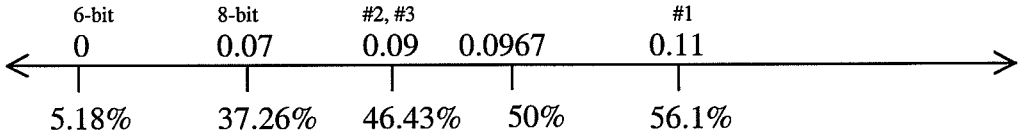


Note: Graph not drawn based on any ratio

[0059] Ratio of characters/numbers vs symbols:

[0060] Standard deviation = 0.012

Mean = 0.0967 (10:1)

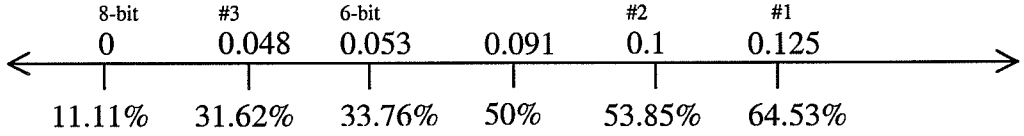


Note: Graph not drawn based on any ratio

[0061] Ratio of characters vs numbers:

[0062] Standard deviation = 0.039

Mean = 0.091 (11:1)



Note: Graph not drawn based on any ratio

[0063] The similarity calculation is as follows:

Message No	Bit Type Comparison	Length similarity (20%)	Ratio of characters/numbers vs symbols similarity (40%)	Ratio of characters vs numbers similarity (40%)	Total similarity (100%)
#1	6-bit	= 100% - |100% - 100%| = 100% x 20% = **20%**	= 100% - |56.1% – 5.18%| = 49.08% x 40% = **19.63%**	= 100% - |64.53% – 33.76%| = 69.23% x 40% = **27.69%**	**67.32%**
#1	8-bit	= 100% - |100% - 37.5%| = 37.5% x 20% = **7.5%**	= 100% - |56.1% – 37.26%| = 81.16% x 40% = **32.46%**	= 100% - |64.53% – 11.11%| = 46.58% x 40% = **18.63%**	**58.59%**
#2	6-bit	= 100% - |50% - 100%| = 50% x 20% = **10%**	= 100% - |46.43% - 5.18%| = 58.75% x 40% = **23.5%**	= 100% - |53.85% - 33.76%| = 79.91% x 40% = **31.96%**	**65.46%**
#2	8-bit	= 100% - |50% - 37.5%| = 87.5% x 20% = **17.5%**	= 100% - |46.43% - 37.26%| = 90.83% x 40% = **36.33%**	= 100% - |53.85% - 11.11%| = 57.26% x 40% = **22.9%**	**76.73%**
#3	6-bit	= 100% - |0% - 100%| = 0% x 20% = **0%**	= 100% - |46.43% - 5.18%| = 58.75% x 40% = **32.46%**	= 100% - |31.62% - 33.76%| = 97.86% x 40% = **39.14%**	**71.6%**
#3	8-bit	= 100% - |0% - 37.5%| = 62.5% x 20% = **12.5%**	= 100% - |46.43% - 37.26%| = 90.83% x 40% = **36.33%**	= 100% - |31.62% - 11.11%| = 79.49% x 40% = **31.8%**	**80.63%**

[0064] The 8-bit encoding type for message #3 is selected as it has the highest total similarity score: 80.63%. However when there exists multiple highest scores, do the following:

[0065] 1. Check if the multiple highest scores have more one encoding type than the others. This means if in the multiple highest scores, there are 2 8-bit and 1 6-bit, the 8-bit will be selected. If the number of 8-bit and 6-bit is even, proceed to #2 else quit.

[0066] 2. Calculate the average of total similarity of 8-bit and 6-bit. Select the encoding type with the highest average. If the average is even, proceed to #3 else quit.

[0067] 3. The encoding selected will be the one that has the most characters (alphabets). If the 8-bit and 6-bit has the same amount, proceed to #4 else quit.

[0068] 4. The encoding selected will be the one that has more characters (alphabets) than numbers. If the 8-bit and 6-bit has the same amount, proceed to #5 else quit

[0069] 5. The 8-bit will be selected.

[0070] In step **329**, since there exists no previous text messages to compare, the following encoding selection will be done:

[0071] 1. The encoding selected will be the one that has the most characters (alphabets);

[0072] 2. If factor #1 caNearest Neighborot resolve the choice, the encoding selected will be the one that has more characters (alphabets) than numbers;

[0073] 3. If factor #2 caNearest Neighborot resolve the choice, the 8-bit will be selected.

[0074] In step **330**, the selected encoding will be recommended by the Optimization Agent.

[0075] Now referring back to FIG 3, based on the recommendation received from the optimization agent, if 6-bit compaction is selected, proceed to step **305** else if 8-bit compaction is used, proceed to step **306**.

[0076] In step **305**, the text message is processed using 6-bit processing. After processing, the text message will be written as per normal 8-bit message and passed back to the SMS module.

[0077] In step **306**, the text message is processed using the normal 8-bit message and passed back to the SMS module. In step **307**, the text message will then be displayed, and the program ends **308**.

[0078] The pseudocodes for receiving/decoding process in accordance with one embodiment of the present invention is as follows:

check if the incoming message length is smaller than 120
if yes then
    use 8-bit type
else
    calculate ratio of characters/numbers vs symbols of incoming message in 8-bit
    calculate ratio of characters vs numbers of incoming message in 6-bit and in 8-bit
    check inbox if there are previous text messages from same sender
    if yes then
        calculate the standard deviation of the length of the previous text messages
        check if there are any messages that are within the standard deviation range of the length of message in 6-bit & 8-bit
        if yes then
            select the messages
        else
            select all messages
        end if
        for the group of messages
            calculate the ratio of characters/numbers vs symbols
            calculate the ratio of characters vs numbers
        end for
        for the group of messages
            compare the similarity score with regards to the length of message with the incoming message in 6-bit
            compare the similarity score with regards to the length of message with the incoming message in 8-bit
            compare the similarity score with regards to the ratio of characters/numbers vs symbols with the incoming message in 6-bit
            compare the similarity score with regards to the ratio of characters/numbers vs symbols with the incoming message in 8-bit
            compare the similarity score with regards to the ratio of characters vs numbers with the incoming message in 6-bit
            compare the similarity score with regards to the ratio of characters vs numbers with the incoming message in 8-bit
        end for
        check if there is only one highest similarity score from the calculation
        if yes then
            select the bit-type corresponding to the score
        else
            check if there are more highest similarity score of one bit-type than the other
            if yes then
                select the bit-type that has more highest similarity scores
            else
                check if there is a higher similarity average of one bit-type's highest similarity scores compared to the other
                if yes then
                    select the bit-type that has the higher similarity average

else
check if there is one bit-type (of the highest similarity scores) that has more characters (alphabets) than the other
if yes then
select the bit-type that has more characters
else
check if there is one bit-type that has more characters (alphabets) than numbers than the other
if yes then
select the bit-type that has more characters (alphabets) than numbers
else
select 8-bit type
end if
end if
end if
end if
else
check if there is one bit-type of incoming message that has more characters (alphabets) than the other bit-type
if yes then
select the bit-type of incoming message that has more characters (alphabets) than the other bit-type
else
check if there is one bit-type that has more characters than numbers than the other bit-type
if yes then
select the bit-type that has more characters than numbers than the other bit-type
else
select 8-bit type
end if
end if
end if
end if

[0079] While the present invention has been described with reference to particular embodiments, it will be understood that the embodiments are illustrative and that the invention scope is not so limited. Alternative embodiments of the present invention will become apparent to those having ordinary skill in the art to which the present invention pertains. Such alternate embodiments are considered to be encompassed within the spirit and scope of the present invention. Accordingly, the scope of the present invention is described by the appended claims and is supported by the foregoing description.

CLAIMS

What is claimed is:

1. A method for encoding/sending an SMS text message via a mobile device embedded with all necessary electronic readable programs for sending/receiving SMS text messages, the method comprising the steps of:

inputting a text message into the mobile device that provides at least two encoding formats, wherein a first of the at least two encoding formats supports less characters so that it uses less bits for encoding each character (compact format), and wherein a second of the at least two encoding formats supports more characters so that it uses more bits for encoding each character (standard format); thereby, when the mobile device defines a limit of how many bytes of each text message can be sent/received under the standard format, the compact format can send/receive more characters within one text message for its use of less bits for encoding each character;

deciding the encoding format for the input text message; wherein, when the input text message can be encoded with the standard format without exceeding the limit, it will be encoded with the standard format; wherein, when the input text message is encoded with the compact format and it still exceeds the limit, it will be encoded with the standard format; and wherein, when the input text message does not exceed the limit if encoded by the compact format but will exceed the limit if encoded by the standard format, it will be encoded with the compact format;

encoding the input text message with either the compact or standard format as decided; and

sending the encoded text message.

2. The method of claim 1, wherein in step of deciding the encoding format, after the input text message is found to be suitable for the compact format but before the encoding step, the method optionally further comprising a step of checking whether the text message contains any character that is not supported the compact format, and if the text message contains any such character, it will be encoded with the standard format.

3. The method of claim 1, wherein the mobile device is mobile phone or PDA.

4. The method of claim 1, wherein the limit is defined by the SMS service system.

5. The method of claim 1, wherein the compact format is a 6-bit format, and the standard format an 8-bit format.

6. The method of claim 1, wherein the method is represented by the following pseudocode:

check if length of message is between (160 x n) and (160 x 8/6 x n) where n is a number larger or equal to 1
if yes then
    check if message can be represented by the 6-bit table
    if yes then
        use 6-bit compaction
    else
        use 8-bit compaction
    end if
else
    use 8-bit compaction
end if

7. A method for receiving/decoding an SMS text message via a mobile device embedded with all necessary electronic readable programs for receiving/decoding SMS text messages in at least two formats, the method comprising the steps of:

detecting an incoming text message;

translating the incoming text message in the at least two format provided by the mobile device, wherein a first of the at least two formats supports less characters so that it uses less bits for encoding each character (compact format), and wherein a second of the at least two encoding formats supports more characters so that it uses more bits for encoding each character (standard format); thereby, when the mobile device defines a limit of how many bytes of each text message can be sent/received under the standard format, the compact format can send/receive more characters within one text message for its use of less bits for encoding each character;

deciding the decoding format for the incoming text message by manipulating the length of the translated text message within the limit; wherein, if the incoming text

message is translated with the compact format and the length is still less than that of a text message with maximum characters allowed by the limit that is encoded with the standard format but translated by the compact format, it will be translated with the standard format; otherwise, the incoming text message will be translated with the compact and standard formats to further decide which format should be used;

decoding the incoming text message with either the compact or standard format decided in the prior step; and

displaying the decoded text message on the mobile device.

8. The method of claim 7, wherein in the deciding step, if the incoming text message is translated with the compact format and the length is more than that of a text message with maximum characters allowed by the limit that is encoded with the standard format but translated by the compact format, the decoding format is decided by the following steps of:

calculating ratio patterns of the incoming text message in both formats, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers;

checking the Inbox folder to see whether there are other text messages from the same sender, wherein if no, further check the Inbox folder to see if there are other previous text messages from other senders, wherein if no, then select the format by choosing the format that has the most characters; if not successful, choosing the format that has more characters than numbers; if not successful, choose the standard format;

calculating the standard deviation in both the compact and standard formats of the length of the previous text messages from either the same sender or other senders; wherein, if there are previous messages which lengths are within the standard deviation range of the incoming text message length, these previous messages will be checked in the next step; and wherein, if there is no previous message within the standard deviation range, all messages will be checked in the next step;

checking the ratio patterns of the selected messages from the prior step, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers;

comparing the patterns of the length and two ratios in previous messages with both compact and standard formats by using CBR's NEAREST NEIGHBOR technique, then the one with the highest similarity scoring will be selected; and

recommending either compact or standard format for decoding.

9. The method of claim 7, wherein the mobile device is mobile phone or PDA.

10. The method of claim 7, wherein the limit is defined by the SMS service system.

11. The method of claim 7, wherein the compact format is a 6-bit format, and the standard format an 8-bit format.

12. The method of claim 7, wherein the method is represented by the following pseudocode:

```
check if the incoming message length is smaller than 120
if yes then
        use 8-bit type
else
        calculate ratio of characters/numbers vs symbols of incoming message in 8-bit
        calculate ratio of characters vs numbers of incoming message in 6-bit and in 8-bit
        check inbox if there are previous text messages from same sender
        if yes then
                calculate the standard deviation of the length of the previous text messages
                check if there are any messages that are within the standard deviation range
of the length of message in 6-bit & 8-bit
                if yes then
                        select the messages
                else
                        select all messages
                end if
                for the group of messages
                        calculate the ratio of characters/numbers vs symbols
                        calculate the ratio of characters vs numbers
                end for
                for the group of messages
                        compare the similarity score with regards to the length of message
with the incoming message in 6-bit
                        compare the similarity score with regards to the length of message
with the incoming message in 8-bit
                        compare the similarity score with regards to the ratio of
characters/numbers vs symbols with the incoming message in 6-bit
```

compare the similarity score with regards to the ratio of characters/numbers vs symbols with the incoming message in 8-bit
compare the similarity score with regards to the ratio of characters vs numbers with the incoming message in 6-bit
compare the similarity score with regards to the ratio of characters vs numbers with the incoming message in 8-bit
end for
check if there is only one highest similarity score from the calculation
if yes then
select the bit-type corresponding to the score
else
check if there are more highest similarity score of one bit-type than the other
if yes then
select the bit-type that has more highest similarity scores
else
check if there is a higher similarity average of one bit-type's highest similarity scores compared to the other
if yes then
select the bit-type that has the higher similarity average
else
check if there is one bit-type (of the highest similarity scores) that has more characters (alphabets) than the other
if yes then
select the bit-type that has more characters
else
check if there is one bit-type that has more characters (alphabets) than numbers than the other
if yes then
select the bit-type that has more characters (alphabets) than numbers
else
select 8-bit type
end if
end if
end if
end if
end if
else
check if there is one bit-type of incoming message that has more characters (alphabets) than the other bit-type
if yes then
select the bit-type of incoming message that has more characters (alphabets) than the other bit-type
else
check if there is one bit-type that has more characters than numbers than the other bit-type

if yes then
select the bit-type that has more characters than numbers than the other bit-type
else
select 8-bit type
end if
end if
end if
end if

13. A mobile device for sending/receiving an SMS text message that has more characters allowed by the limit of the mobile device under its standard encoding format, the mobile device comprising:

an electronic readable medium;

an SMS module embedded within the electronic readable medium, wherein the SMS module comprises an SMS encoding/sending circuit for encoding an input SMS text message in at least two encoding formats depending upon the length of the text message and an SMS receiving/decoding circuit for decoding an incoming SMS text message in either of the at least two encoding format according to the length of the incoming SMS text message; and

a memory embedded within the electronic readable medium for storing the SMS text messages.

14. The mobile device of claim 13, wherein the memory includes ROM and RAM.

15. The mobile device of claim 13, wherein the at least two encoding formats of the SMS encoding/sending circuit comprise a standard encoding format that supports more characters so that its uses more bits for encoding each character (standard format), and at least one compact encoding format that supports less characters so that it uses less bits for encoding each character (compact format); thereby the mobile device can encode/send more characters in one SMS text message using the compact format;

wherein the SMS encoding/sending circuit encodes/sends the input text message by a method comprising the steps of :

deciding the encoding format for the input text message; wherein, if the input text message can be encoded with the standard format without exceeding the limit, it will be

encoded with the standard format; wherein, if the input text message is encoded with the compact format and it still exceeds the limit, it will be encoded with the standard format; and wherein, when the input text message does not exceed the limit if encoded by the compact format but will exceed the limit if encoded by the standard format, it will be encoded with the compact format;

encoding the input text message with either the compact format or the standard format as decided; and

sending the encoded text message.

16. The mobile device of claim 15, wherein in step of deciding the encoding format, after the input text message is found to be suitable for the compact format but before the encoding step, the method optionally further comprising a step of checking whether the text message contains any character that is not supported by the compact format, and if the text message contains any such character, it will be encoded with the standard format.

17. The mobile device of claim 15, wherein the encoding/sending method is represented by the following pseudocode:

```
check if length of message is between (160 x n) and (160 x 8/6 x n) where n is a number
larger or equal to 1
if yes then
        check if message can be represented by the 6-bit table
        if yes then
                use 6-bit compaction
        else
                use 8-bit compaction
        end if
else
        use 8-bit compaction
end if
```

18. The mobile device of claim 13, wherein the at least two encoding formats of the SMS receiving/decoding circuit comprise a standard encoding format that supports more characters so that its uses more bits for encoding each character (standard format), and at least one compact encoding format that supports less characters so that it uses less bits for encoding each character (compact format); thereby the mobile device can receive/decode more characters in one SMS text message using the compact format;

wherein the SMS receiving/decoding circuit receives/sends the incoming text message by a method comprising the steps of :

detecting the incoming text message;

translating the incoming text message in both formats;

deciding the decoding format for the incoming text message by manipulating the length of the translated text message within the limit; wherein, if the incoming text message is less than that of a text message with maximum characters allowed by the limit, it will be translated with the standard format; otherwise, the incoming text message will be translated with both formats to further decide which format should be used for decoding;

decoding the incoming text message with either the compact or standard format decided in the prior step; and

displaying the decoded text message on the mobile device.

19. The mobile device of claim 18, wherein in the deciding step, if the incoming text message is translated with the compact format and the length is more than that of a text message with maximum characters allowed by the limit that is encoded with the standard format but translated by the compact format, the decoding format is decided by the following steps of:

calculating ratio patterns of the incoming text message in both formats, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers;

checking the Inbox folder to see whether there are other text messages from the same sender, wherein if no, further check the Inbox folder to see if there are other previous text messages from other senders, wherein if no, then select the format by choosing the format that has the most characters; if not successful, choosing the format that has more characters than numbers; if not successful, choose the standard format;

calculating the standard deviation in both the compact and standard formats of the length of the previous text messages from either the same sender or other senders; wherein, if there are previous messages which lengths are within the standard deviation range of the incoming text message length, these previous messages will be checked in the next step; and wherein, if there is no previous message within the standard deviation range, all messages will be checked in the next step;

checking the ratio patterns of the selected messages from the prior step, wherein the ratio patterns include ratio of characters/numbers vs symbols and ratio of characters vs numbers;

comparing the patterns of the length and two ratios in previous messages with both the compact format and the standard format by using CBR's NEAREST NEIGHBOR technique, then the one with the highest similarity scoring will be selected; and

recommending either the compact or standard format for decoding.

20. The mobile device of claim 13, wherein the mobile device is mobile phone or PDA.

21. The mobile device of claim 13, wherein the limit is defined by the SMS service system.

22. The mobile device of claim 13, wherein the compact format is a 6-bit format, and the standard format an 8-bit format.